

Juniper Networks® Steel-Belted Radius® Carrier

Reference Guide

Published
2021-11-29

Release
8.6.0

Juniper Networks, Inc.
 1133 Innovation Way
 Sunnyvale, California 94089
 USA
 408-745-2000
www.juniper.net

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. and/or its affiliates in the United States and other countries. All other trademarks may be property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Ulticom, Signalware, Programmable Network, Ultimate Call Control, and Nexworx are registered trademarks of Ulticom, Inc. Kineto and the Kineto Logo are registered trademarks of Kineto Wireless, Inc. Software Advancing Communications and SignalCare are trademarks and service marks of Ulticom, Inc. CORBA (Common Object Request Broker Architecture) is a registered trademark of the Object Management Group (OMG). Raima, Raima Database Manager, and Raima Object Manager are trademarks of Raima, Inc. Sun, Sun Microsystems, the Sun logo, Java, Solaris, MySQL, and all trademarks and logos that contain Sun, Solaris, MySQL, or Java are trademarks or registered trademarks of Oracle America, Inc. in the United States and other countries. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners. All specifications are subject to change without notice.

Contains software copyright 2000–2014 by Oracle America, Inc., distributed under license.

Steel-Belted Radius uses Thrift, licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.

You may obtain a copy of the license at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Steel-Belted Radius uses Xerces XML DOM, from the Apache Group. It has the following terms.

The Apache Software license, Version 1.1

Copyright © 1999-2003 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).” Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names “Xerces” and “Apache Software Foundation” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called “Apache”, nor may “Apache” appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright © 1999, International Business Machines, Inc., <http://www.ibm.com>. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Steel-Belted Radius uses the LDAP v2 Server from the University of Michigan. It has the following terms.

Copyright © 1991 Regents of the University of Michigan. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that this notice is preserved and that due credit is given to the University of Michigan at Ann Arbor. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. This software is provided “as is” without express or implied warranty.

Portions of this software copyright 2003-2009 Lev Walkin <vlm@lionet.info> All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions of this software copyright 1989, 1991, 1992 by Carnegie Mellon University

SBR includes NetSNMP under the following licenses: Derivative Work-1996, 1998-2009 Copyright 1996, 1998-2009. The Regents of the University of California All Rights Reserved. Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Portions of this software copyright © 2001-2009, Networks Associates Technology, Inc. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the Networks Associates Technology, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR

OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions of this software are copyright © 2001–2009, Cambridge Broadband Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of Cambridge Broadband Ltd. may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Steel-Belted Radius uses Jaxen, a “Java XPath Engine” from The Werken Company under the following license:

Copyright 2003 © The Werken Company. All Rights Reserved.

Redistribution and use of this software and associated documentation (“Software”), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name “jaxen” must not be used to endorse or promote products derived from this Software without prior written permission of The Werken Company. For written permission, please contact bob@werken.com.
4. Products derived from this Software may not be called “jaxen” nor may “jaxen” appear in their names without prior written permission of The Werken Company. “jaxen” is a registered trademark of The Werken Company.
5. Due credit should be given to The Werken Company. (<http://jaxen.werken.com/>).

THIS SOFTWARE IS PROVIDED BY THE WERKEN COMPANY AND CONTRIBUTORS “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE WERKEN COMPANY OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR

CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

HTTPClient package Copyright © 1996–2009 Ronald Tschalär (ronald@innovation.ch)

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. For a copy of the GNU Lesser General Public License, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Steel-Belted Radius uses OpenSSL version 1.1.1, which have the following terms:

Copyright ©1998-2018 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit.
(<http://www.openssl.org/>)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit
(<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

OpenSSL is also subject to the following terms.

Copyright ©1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).

The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used.

This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

The word 'cryptographic' can be left out if the rouines from the library being used are not cryptographic related :

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

"This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,

INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

SBR contains software copyright © 2000–2009 by The Legion Of The Bouncy Castle (<http://www.bouncycastle.org>)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Steel-Belted Radius uses modified source from OpenSolaris (now Oracle) under the CDDL, which can be found at http://hub.opensolaris.org/bin/view/Main/opensolaris_license. Modified source is available. Please refer to the SBR Carrier release notes.

SBR includes Spider Monkey libraries under Mozilla Public License Version 2.0

1. Definitions

1.1. “Contributor” means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. “Contributor Version” means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s Contribution.

1.3. “Contribution” means Covered Software of a particular Contributor.

1.4. “Covered Software” means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. “Incompatible With Secondary Licenses” means that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. “Executable Form” means any form of the work other than Source Code Form.

1.7. “Larger Work” means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. “License” means this document.

1.9. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. “Modifications” means any of the following: any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. “Secondary License” means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form” means the form of the work preferred for making modifications.

1.14. “You” (or “Your”) means an individual or a legal entity exercising rights under this License. For legal entities, “You” includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license: under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor: for any code that a Contributor has removed from Covered Software; or for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination

of its Contributions with other software (except as part of its Contributor Version); or under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License v.2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

The "inih" library is distributed under the New BSD license:

Copyright © 2009, Brush Technology
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Brush Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY BRUSH TECHNOLOGY "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL BRUSH TECHNOLOGY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Contains software copyright 2007-2014, by Sencha, Inc., distributed under license.

Steel-Belted Radius uses Jetty 9 under the Apache License 2.0, You may obtain a copy of the license at <http://www.apache.org/licenses/LICENSE-2.0>

Steel-Belted Radius uses Google Web Toolkit (GWT) under the Apache License 2.0, You may obtain a copy of the license at <http://www.apache.org/licenses/LICENSE-2.0>

Steel-Belted Radius uses Apache HTTP components under the Apache License 2.0, You may obtain a copy of the license at <http://www.apache.org/licenses/LICENSE-2.0>

Steel-Belted Radius uses OpenJDK

GNU General Public License, version 2, with the Classpath Exception

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program

itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this

License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright © <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright © year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.
signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

SBR uses Gecko SDK 1.4b

Mozilla Public License Version 2.0

1. Definitions

1.1. "Contributor" means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version" means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution" means Covered Software of a particular Contributor.

1.4. "Covered Software" means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses" means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form" means any form of the work other than Source Code Form.

1.7. "Larger Work" means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License" means this document.

1.9. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications" means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License" means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form" means the form of the work preferred for making modifications.

1.14. "You" (or "Your") means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the

maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

SBR uses Mozilla LDAP C SDK 5.17

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code

of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party

modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims. If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs. If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations. Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions. You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions. Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions. Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works. If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the MPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

"The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____. Portions created by _____ are Copyright (C) _____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[_____] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [_____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

Steel-Belted Radius Carrier 8.6.0 Reference Guide
Release 8.6.0

Revision History
August 2019—Revision 1
April 2020—Revision 2

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <https://support.juniper.net/support/eula/>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Abbreviated Table of Contents

About This Guide | liii

1

Introduction

Chapter 1 Introduction to Configuration Files | 2

2

Steel-Belted Radius Carrier Core and Radius Front-End Files

Chapter 2 Operations Files | 9

Chapter 3 Authentication Configuration Files | 133

Chapter 4 Attribute Processing Files | 187

Chapter 5 Address Assignment Files | 239

Chapter 6 Accounting Configuration Files | 247

Chapter 7 Realm Configuration Files | 262

Chapter 8 EAP Configuration Files | 308

Chapter 9 Session State Register (SSR) Configuration Files | 361

3

Back-End Authentication and Accounting

Chapter 10 SQL Plug-Ins | 386

Chapter 11 LDAP Plug-Ins | 429

Chapter 12 CDR Accounting Plug-Ins | 463

4

SIM Authentication Module

Chapter 13 Common Configurations | 492

Chapter 14 SIM/AKA Authentication | 506

5

Optional Mobility Module Configuration Files

Chapter 15 WiMAX Mobility Module Configuration File | 534

6

SNMP Configuration Files

Chapter 16 SNMP Configuration Overview | 552

Chapter 17 SNMP Traps and Statistics Overview | 565

Part 7 Appendixes

Appendix A Authentication Protocols | 599

Appendix B Vendor-Specific Attributes | 601

Appendix C Configuration Examples | 604

Appendix D Detailed Use Cases | 615

Appendix E SIR.conf File | 655

Table of Contents

About This Guide | liii

Objective | liii

Audience | liii

Documentation Conventions | liv

Related Documentation | lvi

Obtaining Documentation | lxii

Documentation Feedback | lxii

Requesting Technical Support | lxiii

1

Introduction

Introduction to Configuration Files | 2

Configuration Files | 2

Tips for Editing Configuration Files | 6

2

Steel-Belted Radius Carrier Core and Radius Front-End Files

Operations Files | 9

access.ini File | 9

 [Settings] Section | 9

 [Users] and [Groups] Sections | 10

admin.ini File | 11

 [AccessLevel] Section | 12

 [SNMPAgent] Section | 17

events.ini File | 18

 [EventDilutions] Section | 18

 Example | 20

 [Suppress] Section | 20

 Example | 20

 [Thresholds] Section | 21

 Example | 22

events.xml File | 22

radius.ini File | 24**[Addresses] Section | 24****Example 1 | 25****Example 2 | 25****[AuditLog] Section | 26****[AuthRejectLog] Section | 27****[Configuration] Section | 29****[CurrentSessions] Section | 67****[DynAuthProxy] | 68****[LatencyLog] | 69****[EmbedInClass] Section | 72****[HiddenEAPIdentity] Section | 73****[IPPoolSuffixes] Section | 73****[IPv6] Section | 74****[JavaScript] Section | 76****[LDAP] Section | 77****[LDAPAddresses] Section | 78****[Logging] Section | 78****Log File Naming Conventions and Log Rollover | 78****Thread Identifiers | 80****Session Identifiers | 80****Enhanced Proxy Logging | 81****[MsChapNameStripping] Section | 89****[PurgeThreadLogging] Section | 90****[Ports] Section | 91****[Self] Section | 95****[StaticAcctProxy] Section | 96****[Status] Section | 96****[Strip] Section | 99****[StripPrefix] Section | 100****[StripSuffix] Section | 101****[UserNameTransform] Section | 101****Example | 102****[ValidateAuth] and [ValidateAcct] Sections | 103**

sbrd.conf File | 104

services File | 116

servtype.ini File | 117

- [Settings] Section | 117

- [NAS] Section | 118

- [MappingName] Section | 119

- Example | 119

update.ini File | 120

- [HUP] and [USR2] Sections | 120

- Example | 120

Auto-Restart Files | 126

- Perl SNMP Support | 126

- Perl System Log Support | 127

- sbrd.conf File | 127

- radiusd.conf File | 128

- radiusd.conf Configuration File | 128

Authentication Configuration Files | 133

authlog.ini File | 133

- [Alias/name] Sections | 134

- [Attributes] Section | 135

- [Configuration] Section | 136

- [Syslog] Section | 136

- [Settings] Section | 139

authReport.ini File | 144

- [AcceptReport] Section | 145

- [BadSharedSecretReport] Section | 145

- [RejectReport] Section | 146

- [UnknownClientReport] Section | 146

authReportAccept.ini File | 147

- [Attributes] Section | 147

- [Settings] Section | 148

authReportBadSharedSecret.ini File | 151**[Attributes] Section | 151****[Settings] Section | 152****authReportReject.ini File | 155****[Attributes] Section | 155****[Settings] Section | 159****authReportUnknownClient.ini File | 162****[Attributes] Section | 162****[Settings] Section | 163****blacklist.ini File | 166****lockout.ini File | 167****[ClientExclusionList] Section | 168****[UserExclusionList] Section | 168****redirect.ini File | 169****[Settings] Section | 169****[ClientExclusionList] Section | 170****statlog.ini File | 171****[Settings] Section | 172****[Statistics] Section | 173****Attribute Processing Files | 187****Dictionary Files | 187****.dct Files | 187****.dct File Location | 188****.dct File Records | 188****Editing .dct Dictionary Files | 189****Include Records | 189****ATTRIBUTE Records | 190****Macro Records | 195****OPTION Records | 195****.dic Files | 196****.dic File Location | 197****<?dict> Element | 197****<vendor> Element | 197**

[<attribute> Element | 198](#)

[Editing .dic Dictionary Files | 199](#)

[Structured Attributes | 199](#)

[Structured Attribute Dictionary Definitions | 200](#)

[XML Format of Dictionary Files | 201](#)

[Functional Areas That Use Subattributes | 203](#)

[Packet Parsing and Formatting | 203](#)

[Features that Support the Use of Structured and Subattributes | 203](#)

[Single Subattribute Insertion | 204](#)

[Attribute Filtering | 205](#)

[Proxy Attribute Mapping | 205](#)

[Structured Attributes in Return Lists and Check Lists | 205](#)

[Javascript | 207](#)

[Converting Previous Attribute Flatteners with Subattributes | 207](#)

[Plug-in Attribute Access | 207](#)

[Example of Configuration and Usage of a Structured Attribute | 207](#)

[3GPP2 Data Definition | 207](#)

[SBR Carrier XML Dictionary Definition | 208](#)

[Example Data | 209](#)

[LCI Encoding | 211](#)

[classmap.ini File | 212](#)

[\[AttributeName\] Section | 212](#)

[filter.ini File | 213](#)

[Filter Rules | 213](#)

[Order of Filter Rules | 216](#)

[Examples | 216](#)

[Values in Filter Rules | 217](#)

[Referencing Attribute Filters | 219](#)

[sample.rr File | 220](#)

[spi.ini File | 222](#)

[\[Keys\] Section | 222](#)

[\[Hosts\] Section | 223](#)

vendor.ini File | 224

- [Vendor-Product Identification] Section | 224

- Product-Scan Settings | 227

Adding NAS Location Information to Access-Request Messages | 230

- Location-Specific Configuration Files | 231

- locspec.ctrl File | 232

- Example | 232

- Example | 233

- Example | 234

- Example | 235

- proxy.ini File | 236

- Example | 236

- realm.pro File | 237

- Example realm.pro file: | 237

- Example Configuration for Adding NAS Location Attributes to Access-Request | 237

- Example Overview | 237

- Example Configuration | 237

Address Assignment Files | 239**dhcp.ini File | 239**

- [Settings] Section | 239

- [Pools] Section | 242

pool.dhc Files | 242

- [Settings] Section | 242

- [Request] Section | 243

- [Reply] Section | 246

- Reconfiguring Pools | 246

Accounting Configuration Files | 247**account.ini File | 247**

- [Alias/name] Sections | 247

- [Attributes] Section | 249

- [Configuration] Section | 250

- [Settings] Section | 251

[TypeNames] Section | 254

acctReport.ini File | 256

[Settings] Section | 256

[UnknownClientReport] Section | 259

[BadSharedSecretReport] Section | 259

[Attributes] Section | 260

sessionTable.ini File | 260

[Settings] Section | 260

Realm Configuration Files | 262

Proxy Realm Configuration Files | 263

Sample proxy.ini Settings | 263

Sample Proxy Realm (.pro) File | 264

Sample filter.ini File | 265

Directed Realm Configuration Files | 266

Sample proxy.ini File | 266

Sample Directed Realm (.dir) File | 267

proxy.ini File | 268

[Configuration] Section | 269

[Realms] Section | 270

[Directed] Section | 271

[Processing] Section | 272

[AttributeMap] Sections | 272

[DirectedAcctMethods] Section | 275

[StaticAcct] Section | 276

[Interfaces] Section | 278

Proxyrl.ini File | 279

Proxy RADIUS Configuration (.pro) File | 280

[Auth] Section | 281

[Acct] Section | 285

[AutoStop] Section | 289

[Called-Station-ID] Section | 290

[DynAuth] Section | 291

Target Selection Rules | 291

Round-Robin Load Balancing | 292

Selecting a Backup Server | 293

Realm Retry Policy | 293

[FastFail] Section | 294**[ModifyUser] Section | 296****[SpooledAccounting] Section | 297**

Retry Sequence | 299

Directed Realm Configuration (.dir) File | 300

[Auth] Section | 301

[AuthMethods] Section | 303

[Acct] Section | 304

[AcctMethods] Section | 305

[Called-Station-ID] Section | 306

[ModifyUser] Section | 306

radius.ini Realm Settings | 307**EAP Configuration Files | 308****eap.ini File | 308****peapauth.aut File | 312**

[Bootstrap] Section | 312

[Server_Settings] Section | 313

Cipher_Suites Parameter | 313

[Inner_Authentication] Section | 317

[Request Filters] Section | 318

[Response Filters] Section | 319

[Session_Resumption] Section | 320

tlsauth.aut File | 322

[Server_Settings] Section | 322

Cipher_Suites Parameter | 322

[CRL_Checking] Section | 326

[Session_Resumption] Section | 329

Sample tlsauth.aut File | 331

tlsauth.eap File | 334**[Server_Settings] Section | 334****Cipher_Suites Parameter | 334****[Secondary_Authorization] Section | 338****[CRL_Checking] Section | 342****[Session_Resumption] Section | 344****Sample tlsauth.eap File | 345****Configuring Secondary Authorization | 347****SQL Authentication | 347****LDAP Authentication | 347****ttlsauth.aut File | 348****[Bootstrap] Section | 348****[Server_Settings] Section | 349****Cipher_Suites Parameter | 349****[Inner_Authentication] Section | 352****[Request_Filters] Section | 352****[Response_Filters] Section | 354****[CRL_Checking] Section | 355****[Session_Resumption] Section | 357****Sample ttlsauth.aut File | 359****Session State Register (SSR) Configuration Files | 361****Configuring the config.ini File | 361****[tcp default] Section | 361****[ndbd default] Section | 362****[ndbd] Section | 367****Configuring the dbclusterndb.gen File | 368****[Bootstrap] Section | 368****[NDB] Section | 369****[Database] Section | 372****[IpAddressPools] Section | 374****[IpAddressPools:PoolName] Section | 377**

Using the georedSess.ses File to Configure the Geo-Redundancy Feature | 379

[Bootstrap] Section | 379

[ClientSettings] Section | 380

[ServerSettings] Section | 383

Back-End Authentication and Accounting

SQL Plug-Ins | 386

Common Configuration Items | 386

[Bootstrap] Section | 386

[Settings] Section | 387

Limitations of Underlying Database APIs | 392

SQL Parameter | 393

ErrorMap | 395

LogLevel | 397

[Server] Section | 397

[Server/name] Sections | 399

Last Resort Server | 400

Load Balancing Example | 400

JDBC Plug-ins | 401

SQL Authentication | 403

[Settings] Section | 404

PasswordFormat | 404

ClearTextBinary | 404

DefaultResults | 404

SuccessResult | 404

UpperCaseName | 404

[Results] Section | 405

Default [Results] Parameters | 408

[FailedSuccessResultAttributes] Section | 408

[Failure] Section | 409

[Strip] Sections | 410

Example: SQL Authentication Configuration File | 412

SQL Accounting | 413**[Settings] Section | 414****[Type] Section | 414****[Type/statement] Sections | 415****[TypeNames] Section | 417****Example: SQL Accounting Configuration File | 418****SQL Accessors | 419****[Settings] Section | 419****[Results] Section | 420****[Failure] Section | 420****Example: SQL Accessor Configuration File | 421****Detailed Use Cases | 422****Working with Stored Procedures | 422****SQL Database Data Retrieval Methods | 423****SQL=SELECT Method for Data Retrieval from SQL Databases | 423****Stored Procedure Method for Data Retrieval from SQL Databases | 426****LDAP Plug-Ins | 429****Overview | 429****Common Configurations | 429****[Bootstrap] Section | 429****[Settings] Section | 430****[Server] Section | 436****[Server/name] Sections | 437****[Search/DoLdapSearch] Sections | 441****LDAP Authentication | 444****LDAP Authentication Variable Names | 445****[Bootstrap] Section | 445****[Settings] Section | 446****[JavaScript] Section | 447****[Attributes/name] Sections | 448****[RejectResponse] Section | 449****[Response] Section | 450****[Request] Section | 452**

[Defaults] Section | 454

[Failure] Section | 455

Grouped Attributes | 457

GlobalProfile Attribute | 457

ProfileData Attribute | 458

Modifying ldapauth.aut | 459

LDAP Accessor Files | 460

[Settings] Section | 460

[Request] Section | 461

[Response] Section | 462

[Attributes/AttrList] Section | 462

CDR Accounting Plug-Ins | 463

CDR Process Overview | 463

Types of Call Detail Records | 464

Configuring Accounting Options with cdracct.acc | 465

[Bootstrap] Section of cdracct.acc | 465

Example | 466

[Settings] Section of cdracct.acc | 466

Example | 470

Displaying CDR Information | 471

CDR Files | 471

Using cldrump to Display CDR File Contents | 472

cldrump Output | 474

Displaying ASN1 CDR Files in Raw Format Using dumpasn1 | 475

CDR Fields | 475

CDR Field Formats for Binary and ASN.1 CDR Files | 488

Field Formats for Binary Version 1 and Binary Version 2 CDR Files | 488

SIM Authentication Module

Common Configurations | 492

ss7db.gen File | 492

 [Bootstrap] Section | 492

 [Settings] Section | 493

gsmmap.gen File | 495

 [Bootstrap] Section | 496

 [Settings] Section | 496

 [Realms] Section | 497

 Configuring Each Realm Section | 497

 Example | 498

 Relationship Between Sections | 498

 Network Equipment and Data Needed for Processing Access-Requests | 499

 Example: Authorization String | 500

 Disabling Authorization from EAP-SIM | 500

 Target Module Section | 501

 Target Module Fields (General Case) | 501

 MAP Gateway Target Module Fields | 502

 Example of MAP Gateway Target Module Fields | 503

 SQL Database Target Module Fields | 503

 Example of SQL Database Target Module | 503

 LDAP Database Target Module Fields | 504

 Example of LDAP Database Target Module | 504

Signalware MML Commands | 504

SIM/AKA Authentication | 506

Overview | 506

Configuring the simauth.aut File | 506

 simauth.aut [Bootstrap] Section | 507

 simauth.aut [Settings] Section | 507

 simauth.aut [ProfileMap] Section | 513

Authentication Gateway | 516

Configuring the ulcmmg.conf File | 516

Configuring the GWrelay.conf File | 517

Starting and Stopping the GWrelay Process | 518

Configuring the authGateway.conf File | 519

[Routing-Configuration] Section | 519

[Supported-MAP-Messages] Section | 523

[Common-AGW-Configurations] Section | 524

[Process<name>] Section | 527

Configuring the authGateway Startup with MML Commands | 528

Example—Creating and Starting the authGateway Process | 531

5

Optional Mobility Module Configuration Files**WiMAX Mobility Module Configuration File | 534**

wimax.ini File | 534

[Settings] Section | 534

[ASNGW-Requests] Section | 539

[ASNGW-Requests/<name>] Section | 542

[Home-Agent-Requests] Section | 543

[DHCP-Server-Requests] Section | 545

[Other-Requests] Section | 546

[HAs] Section | 547

[DHCPsServers] Section | 547

[RADIUS client-Access-Request-Required-Attributes] Sections | 547

Example wimax.ini File | 548

6

SNMP Configuration Files**SNMP Configuration Overview | 552**

SNMP Overview | 552

SNMP Network Management Architecture Overview | 552

SNMP Versions | 553

MIBs Overview | 554

SNMP Messages | 554

Dilution and Threshold | 555

SNMP Community Overview | 555

Rate Statistic Overview | 556

jnpnsnmpd.conf File Overview | 556

Access Control Overview and Syntax | 557

Security Names Overview and Syntax | 558

Access View Overview and Syntax | 559

Group Access Overview and Syntax | 559

System Contact Overview and Syntax | 560

Traps Overview and Syntax | 560

[snmp] Overview and Syntax | 562

init.jnpnsnmpd Overview and Syntax | 562

Subagent Overview and Syntax | 562

testagent.sh Script Overview and Syntax | 563

SNMP Traps and Statistics Overview | 565

Trap Variables Overview | 566

Trap Definitions | 569

Rate Statistics | 587

Part 7

Appendixes

Appendix A

Authentication Protocols | 599

Appendix B

Vendor-Specific Attributes | 601

Appendix C

Configuration Examples | 604

Steel-Belted Radius Carrier: 3G-to-Wi-Fi Offload Solution Using the SBR MAP Gateway with EAP-SIM or EAP-AKA | 604

Appendix D

Detailed Use Cases | 615

Using SQL Accessors | 615

Configuring gsmmap.gen for Key Field Identification | 616

Examples | 616

Configuring sqlaccessor.gen or sqlaccessor_jdbc.gen for Key Field Identification | 616

Examples | 616

Using LDAP Accessors | 617

Configuring ldapaccessor.gen for Key Field Identification | 618

Assigning IP Addresses Based on APN | 618

Overview | 619

Tasks for Assigning IP Address Based on Access Point | 620

Configuring simauth.aut for IP Address Assignment | 620

Create the IP Address Pool | 622

Adding Attributes to an Access-Accept | 622

Overview | 622

Data Flow | 622

Configuration Tasks | 624

Configuring Files for Adding Attributes to Access-Accept | 624

Example Configuration for Adding Attributes to Access-Accept | 627

Example Overview | 627

Example Notes | 628

Activate the Authentication Method | 629

Interfacing with a Kineto IP Network Controller | 630

Attribute Handling Methods | 631

Access-Request Conversion | 632

Access-Accept Conversion | 639

Access-Reject Conversion | 642

Configuring the SIM Authentication Module for Handling Kineto Attributes | 642

Configuring the kinetoUMAAttrHandler.ctrl File | 642

Configuring the controlpoints.ini File | 644

Configuring Kineto Attribute Recognition | 645

Activating the Authentication Method | 649

Developing Applications for the S1 Interface | 651

Using Managed IPv6 Address Pools | 651

Appendix E

SIR.conf File | 655

[SBR_INFO] Section | 655

[Level_One] Section | 656

[SBR_Configuration_Files] Section | 656

[SBR_GUI_Configuration] Section | 657

[SBR_Package_Information] Section | 657

[SBR_Dictionary_Information] Section | 658

[Certificate_JAR_Files] Section | 659

[Radius_Process_Info] Section | 659

[SBR_XML] Section | 660

[System_Info] Section | 661

[Level_Two] Section | 663

[Log_Files] Section | 664

[Core_Files] Section | 665

Glossary | 666

About This Guide

IN THIS SECTION

- Objective | [liii](#)
- Audience | [liii](#)
- Documentation Conventions | [liv](#)
- Related Documentation | [lvi](#)
- Obtaining Documentation | [lxii](#)
- Documentation Feedback | [lxii](#)
- Requesting Technical Support | [lxiii](#)

This preface provides the following guidelines for using the Steel-Belted Radius Carrier Reference Guide:

Objective

This guide describes how to configure and administer Steel-Belted Radius Carrier software running on the Solaris operating system.

Audience

This guide is intended for network administrators working for wireline and wireless carriers that are deploying converged services or emerging wireless technologies such as Worldwide Interoperability Microwave Access (WiMAX). It provides the information that administrators need to implement and maintain authentication, authorization, and accounting (AAA) services.

This guide assumes that you are familiar with general RADIUS and networking concepts, as well as the network environment that includes Steel-Belted Radius Carrier.

If you use Steel-Belted Radius Carrier with third-party products such as Oracle, this guide assumes you are familiar with the installation, configuration, and use of those products.

Documentation Conventions

Table 1 on page liv defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
NOTE:	Informational note	Indicates important features or instructions.
CAUTION:	Caution	Indicates a situation that might result in loss of data or hardware damage.
WARNING:	Warning	Alerts you to the risk of personal injury.

Table 2 on page liv describes the text conventions used throughout this manual.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Text Conventions		
Bold text like this	Represents commands and keywords in text.	<ul style="list-style-type: none"> Issue the clock source command. Specify the keyword exp-msg.
Bold text like this	Represents text that the user must type.	host1(config)#traffic class low-loss1
Fixed-width text like this	Represents information as displayed on your terminal's screen.	<pre>host1#show ip ospf 2 Routing Process OSPF 2 with Router ID 5.5.0.250 Router is an Area Border Router (ABR)</pre>
<i>Italic text like this</i>	<ul style="list-style-type: none"> Emphasizes words. Identifies variables. Identifies chapter, appendix, and book names. 	<ul style="list-style-type: none"> There are two levels of access, <i>user</i> and <i>privileged</i>. <i>clusterId</i>, <i>ipAddress</i>. <i>Appendix A, System Specifications</i>.
Plus sign (+) linking key names	Indicates that you must press two or more keys simultaneously.	Press Ctrl+b.

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
<i>radiusdir</i>	Represents the directory into which Steel-Belted Radius Carrier has been installed. The default location is /opt/JNPRsbr/radius on Solaris systems, but any location may be specified during installation.	Change directories to /radiusdir /radiusdir
Syntax Conventions		
Plain text like this	Represents keywords.	terminal length
<i>Italic text like this</i>	Represents variables.	<i>mask, accessListName</i>
< > (angle brackets)	Enclose a list of possible selections.	<add replace>
(pipe symbol)	Represents a choice to select one keyword or variable in a list of choices that is separated by the pipe symbol.	<p>diagnostic line</p> <p>In this example, you must specify <i>add</i> or <i>replace</i> but cannot specify both:</p> <p><add replace></p> <p>Attribute [,Attribute]</p>
[] (brackets)	Represent optional keywords or variables.	<p>[internal external], or</p> <p><add replace> = Attribute [,Attribute], where the second attribute is identified as optional by the brackets.</p> <p>When they are used in a configuration files brackets identify a section of the file.</p> <p>In scripts or in operating system commands, brackets indicate the default response or entry.</p>
[]* (brackets and asterisk)	Represent optional keywords or variables that can be entered more than once.	[level1 level2 l1]*
{ } (braces)	Represent required keywords or variables.	<p>{ permit deny } { in out }</p> <p>{ clusterId ipAddress }</p>

Related Documentation

Table 3 on page lvi lists and describes the Steel-Belted Radius Carrier documentation set:

Table 3: Steel-Belted Radius Carrier Documentation

Document	Description
<i>Steel-Belted Radius Carrier Installation Guide</i>	Describes how to install the Steel-Belted Radius Carrier software on the server.
<i>Steel-Belted Radius Carrier Administration and Configuration Guide</i>	Describes how to configure and operate the Steel-Belted Radius Carrier and its separately licensed modules.
<i>Steel-Belted Radius Carrier Reference Guide</i>	Describes the settings and valid values of the Steel-Belted Radius Carrier configuration files.
<i>Steel-Belted Radius Carrier Performance, Planning and Tuning Guide</i>	Provides tips, use cases, and tools you need to: <ul style="list-style-type: none"> • Improve SBRC performance through planning, analysis, and configuration • Increase SBRC throughput and reliability • Analyze specific use cases, in the lab or in the production environment, to identify areas of potential performance enhancement and to limit the impact of resource constraints and failure scenarios
<i>Steel-Belted Radius Carrier Release Notes</i>	Contains the latest information about features, changes, known problems, and resolved problems.

NOTE: If the information in the Release Notes differs from the information in any guide, follow the Release Notes.

Requests for Comments (RFCs)

The Internet Engineering Task Force (IETF) maintains an online repository of Request for Comments (RFCs) online at <http://www.ietf.org/rfc.html>.

Table 4 on page lvii lists the RFCs that apply to Steel-Belted Radius Carrier.

Table 4: RFCs Related to Steel-Belted Radius Carrier

RFC Number	Title
RFC 1035	<i>Domain Names - Implementation and Specification</i> . P. Mockapetris. November 1987.
RFC 1155	<i>Structure and Identification of Management Information for TCP/IP-based Internets</i> . M. Rose, K. McCloghrie, May 1990.
RFC 1213	<i>Management Information Base for Network Management of TCP/IP-based internets: MIB-II</i> . K. McCloghrie, M. Rose, March 1991.
RFC 2006	<i>The Definitions of Managed Objects for IP Mobility Support using SMIv2</i> . D. Cong and others. October 1996.
RFC 2104	<i>HMAC: Keyed-Hashing for Message Authentication</i> . H. Krawczyk, M. Bellare, R. Canetti. February 1997.
RFC 2246	<i>The TLS Protocol</i> . T. Dierks, C. Allen. January 1999.
RFC 2271	<i>An Architecture for Describing SNMP Management Frameworks</i> . D. Harrington, R. Presuhn, B. Wijnen, January 1998.
RFC 2284	<i>PPP Extensible Authentication Protocol (EAP)</i> . L. Blunk, J. Vollbrecht, March 1998.
RFC 2433	<i>Microsoft PPP CHAP Extensions</i> . G. Zorn, S. Cobb, October 1998.
RFC 2548	<i>Microsoft Vendor-specific RADIUS Attributes</i> . G. Zorn. March 1999.
RFC 2607	<i>Proxy Chaining and Policy Implementation in Roaming</i> . B. Aboba, J. Vollbrecht, June 1999.
RFC 2618	<i>RADIUS Authentication Client MIB</i> . B. Aboba, G. Zorn. June 1999.
RFC 2619	<i>RADIUS Authentication Server MIB</i> . G. Zorn, B. Aboba. June 1999.
RFC 2620	<i>RADIUS Accounting Client MIB</i> . B. Aboba, G. Zorn. June 1999.
RFC 2621	<i>RADIUS Accounting Server MIB</i> . G. Zorn, B. Aboba. June 1999.
RFC 2622	<i>PPP EAP TLS Authentication Protocol</i> . B. Aboba, D. Simon, October 1999.

Table 4: RFCs Related to Steel-Belted Radius Carrier (*continued*)

RFC Number	Title
RFC 2719	<i>Framework Architecture for Signaling Transport</i> . L. Ong et al., October 1999
RFC 2809	<i>Implementation of L2TP Compulsory Tunneling via RADIUS</i> . B. Aboba, G. Zorn. April 2000.
RFC 2865	<i>Remote Authentication Dial In User Service (RADIUS)</i> . C. Rigney, S. Willens, A. Rubens, W. Simpson. June 2000.
RFC 2866	<i>RADIUS Accounting</i> . C. Rigney. June 2000.
RFC 2867	<i>RADIUS Accounting Modifications for Tunnel Protocol Support</i> . G. Zorn, B. Aboba, D. Mitton. June 2000.
RFC 2868	<i>RADIUS Attributes for Tunnel Protocol Support</i> . G. Zorn, D. Leifer, A. Rubens, J. Shriver, M. Holdrege, I. Goyret. June 2000.
RFC 2869	<i>RADIUS Extensions</i> . C. Rigney, W. Willats, P. Calhoun. June 2000.
RFC 2882	<i>Network Access Servers Requirements: Extended RADIUS Practices</i> . D. Mitton. July 2000.
RFC 2960	<i>Stream Control Transmission Protocol</i> . R. Stewart and others. October 2000.
RFC 3046	<i>DHCP Relay Agent Information Option</i> . M. Patrick. January 2001.
RFC 3118	<i>Authentication for DHCP Messages</i> . R.Droms and others. June 2001.
RFC 3162	<i>RADIUS and IPv6</i> . B. Aboba, G. Zorn, D. Mitton. August 2001.
RFC 3344	<i>IP Mobility Support for IPv4</i> . C. Perkins. August 2002.
RFC 3539	<i>Authentication, Authorization, and Accounting (AAA) Transport Profile</i> . B. Aboba, J. Wood. June 2003.
RFC 3575	<i>IANA Considerations for RADIUS (Remote Authentication Dial-In User Service)</i> . B. Aboba, July 2003.
RFC 3576	<i>RFC3576 - Dynamic Authorization Extensions to Remote to Remote Authentication Dial In User Service</i> . Network Working Group, 2003

Table 4: RFCs Related to Steel-Belted Radius Carrier (*continued*)

RFC Number	Title
RFC 3579	<i>RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)</i> . B. Aboba, P. Calhoun, September 2003.
RFC 3580	<i>IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines</i> . P. Congdon, B. Aboba, A. Smith, G. Zorn, J. Roese, September 2003.
RFC 3588	<i>Diameter Base Protocol</i> . P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko. September 2003.
RFC 3748	<i>Extensible Authentication Protocol</i> . B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, H. Levkowetz. June 2004.
RFC 3957	<i>Authentication, Authorization, and Accounting (AAA) Registration Keys for Mobile IPv4</i> . C. Perkins and P. Calhoun. March 2005.
RFC 4005	<i>Diameter Network Access Server Application</i> . P. Calhoun, G. Zorn, D. Spence, D. Mitton. August 2005.
RFC 4017	<i>Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs</i> . D. Stanley and others. March 2005.
RFC 4072	<i>Diameter Extensible Authentication Protocol (EAP) Application</i> . P. Eronen, G. Zorn, T. Hiller. August 2005.
RFC 4186	<i>Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM)</i> . H. Haverinen, J. Salowey. January 2006.
RFC 4187	<i>Extensible Authentication Protocol Method for Global System for 3rd Generation Authentication and Key Agreement (EAP-AKA)</i> . J. Arkko, H. Haverinen. January 2006.
RFC 4282	<i>The Network Access Identifier</i> . B. Aboba and others. December 2005.
RFC 4284	<i>Identity Selection Hints for the Extensible Authentication Protocol (EAP)</i> . F. Adrangi, V. Lortz, F. Bari, P. Eronen. January 2006.
RFC 4306	<i>Internet Key Exchange (IKEv2) Protocol</i> . C. Kaufman. December 2005.

Table 4: RFCs Related to Steel-Belted Radius Carrier (*continued*)

RFC Number	Title
RFC 4372	<i>Chargeable User Identity</i> . F. Adrangi and others. January 2006.
RFC 4510	<i>Lightweight Directory Access Protocol (LDAP) Technical Specification Road Map</i> . K. Zeilenga, June 2006.
RFC 4666	<i>Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)</i> . K. Morneault, J. Pastor-Balbas. September 2006.
RFC 4668	<i>RADIUS Authentication Client MIB for IPv6</i> . D. Nelson. August 2006.
RFC 4669	<i>RADIUS Authentication Server MIB for IPv6</i> . D. Nelson. August 2006.
RFC 4670	<i>RADIUS Accounting Client MIB for IPv6</i> . D. Nelson. August 2006.
RFC 4671	<i>RADIUS Accounting Server MIB for IPv6</i> . D. Nelson. August 2006.
RFC 5281	<i>Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)</i> . P. Funk, S. Blake-Wilson. August 2008.
RFC 5448	<i>Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')</i> . J. Arkko, V. Lehtovirta, P. Eronen. May 2009.
RFC 5997	<i>Use of Status-Server Packets in the Remote Authentication Dial In User Service (RADIUS) Protocol</i> . A. DeKok. August 2010.
RFC 6733	<i>Diameter Base Protocol</i> . V. Fajardo, J. Arkko, J. Loughney, G. Zorn. October 2012.
RFC 6911	<i>RADIUS Attributes for IPv6 Access Networks</i> . W. Dec, B. Sarikaya, G. Zorn, D. Miles, B. Lourdelet. April 2013.

3GPP Technical Specifications

The Third-Generation Partnership Project (3GPP) and 3GPP2 maintains an online repository of Technical Specifications and Technical Reports at <http://www.3gpp.org> and <http://www.3gpp2.org>, respectively.

Table 5 on page lxi lists the 3GPP Technical Specifications that apply to Steel-Belted Radius Carrier.

Table 5: 3GPP Technical Specifications

3GPP TS Number	Title	Applicable Sections
3GPP TS 22.234 Version 12.0.0	<i>Requirements on 3GPP system to Wireless Local Area Network (WLAN) interworking</i>	<ul style="list-style-type: none"> ● Section 5.1.7: Interworking between PLMN and WLANs
3GPP TS 23.003 Version 12.6.0	<i>Numbering, addressing, and identification</i>	<ul style="list-style-type: none"> ● Section 2.2: Composition of IMSI
3GPP TS 23.008 Version 12.6.0	<i>Organization of subscriber data</i>	<ul style="list-style-type: none"> ● Section 3B: Definition of subscriber data I-WLAN domain
3GPP TS 23.234 Version 12.0.0	<i>3GPP system to Wireless Local Area Network (WLAN) interworking; System description</i>	<ul style="list-style-type: none"> ● Section 6.1: Reference Model ● Section 6.2: Network Elements
3GPP TS 23.402 Version 12.8.0	<i>Architecture enhancements for non-3GPP accesses</i>	<ul style="list-style-type: none"> ● Section 4.1: Concepts ● Section 4.3: Network Elements
3GPP TS 24.302 Version 14.4.0	<i>Access to the 3GPP Evolved Packet Core (EPC) via non-3GPP access networks; Stage 3</i>	<ul style="list-style-type: none"> ● Section 6: UE – EPC Network protocols ● Section 8: PDUs and parameters specific to the present document
3GPP TS 29.002 Version 12.7.0	<i>Mobile Application Part (MAP) specification</i>	<ul style="list-style-type: none"> ● Section 6: Requirements concerning the use of SCCP and TC ● Section 7.1: Terminology and definitions ● Section 7.2: Modelling principles ● Section 7.3: Common MAP service
3GPP TS 29.273 Version 12.7.0	<i>Evolved Packet System (EPS); 3GPP EPS AAA interfaces</i>	<ul style="list-style-type: none"> ● Section 4: SWa Description ● Section 6: SWd Description ● Section 5: STa Description ● Section 7: SWm Description ● Section 8: SWx Description ● Section 9: S6b and H2 Description ● Section 10: Result-Code and Experimental-Result Values

Table 5: 3GPP Technical Specifications (*continued*)

3GPP TS Number	Title	Applicable Sections
3GPP TS 33.402 Version 14.2.0	<i>3GPP System Architecture Evolution (SAE); Security aspects of non-3GPP accesses</i>	<ul style="list-style-type: none"> • <i>Section 6: Authentication and key agreement procedures</i> • <i>Section 7: Establishment of security contexts in the target access system</i> • <i>Section 8: Establishment of security between UE and ePDG</i> • <i>Section 9: Security for IP based mobility signalling</i> • <i>Section 14: Temporary identity management</i>

WiMAX Technical Specifications

The WiMAX Forum Networking Group (NWG) maintains a repository of technical documents and specifications online at <http://www.wimaxforum.org>. You can also view the WiMAX IEEE standards, 802.16e-2005 for mobile WiMAX and 802.16-2004 for fixed WiMAX, online at <http://www.ieee.org>.

Third-Party Products

For information about configuring your Ulticom software and hardware, or your access servers and firewalls, consult the manufacturer's documentation.

Obtaining Documentation

To obtain the most current version of all Juniper Networks technical documentation, see the products documentation page on the Juniper Networks website at <https://www.juniper.net/>.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation to better meet your needs. Send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/documentation/feedback/>. If you are using e-mail, be sure to include the following information with your comments:

- Document name
- Document part number

- Page number
- Software release version

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- **JTAC Policies**—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>
- **Product Warranties**—For product warranty information, visit <https://www.juniper.net/support/warranty/>
- **JTAC Hours of Operation**—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings:
<https://support.juniper.net/support/>
- Find product documentation:
<https://www.juniper.net/documentation/>
- Find solutions and answer questions using our Knowledge Base: <https://kb.juniper.net/>
- Download the latest versions of software and review release notes:
<https://support.juniper.net/support/downloads/>
- Search technical bulletins for relevant hardware and software notifications:
<https://kb.juniper.net/InfoCenter/index?page=subscriptions>, "Manage My Subscriptions"
- Open a case online in the Juniper Networks Customer Service Portal:
<https://my.juniper.net>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool located at

<https://entitlementsearch.juniper.net/entitlementsearch/>

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Juniper Networks Customer Service Portal at <https://my.juniper.net>
- Call 1-888-314-JTAC (1-888-314-5822 – toll free in the USA, Canada, and Mexico)

For international or direct-dial options in countries without toll-free numbers, visit <https://www.juniper.net/support/requesting-support.html>

When you contact technical support, be ready to provide:

- Your Steel-Belted Radius Carrier release number (for example, Steel-Belted Radius Carrier Release 7.x).
- Information about the server configuration and operating system, including any OS patches that have been applied.
- For licensed products under a current maintenance agreement, your license or support contract number.
- A detailed description of the problem.
- Any documentation that may help in resolving the problem, such as error messages, memory dumps, compiler listings, and error logs.

1

PART

Introduction

[Introduction to Configuration Files | 2](#)

Introduction to Configuration Files

IN THIS CHAPTER

- Configuration Files | 2
- Tips for Editing Configuration Files | 6

This chapter introduces configuration files that Steel-Belted Radius Carrier uses to implement its version of the RADIUS (Remote Authentication Dial-In User Service) protocol. Steel-Belted Radius Carrier interfaces with a wide variety of network access devices, and authenticates remote and WLAN users against numerous back-end databases, allowing you to consolidate the administration of all your remote and WLAN users, however they connect to your network. These topics are included in this chapter:

Configuration Files

The configuration and behavior of your Steel-Belted Radius Carrier server is determined by a set of configuration files. In many cases, you must edit these files manually. In a few cases, the contents of a configuration file are updated dynamically when you use Web GUI to change settings.

Configuration files reside in the *radiusdir* directory. [Table 6 on page 2](#) identifies the files that are used by Steel-Belted Radius Carrier.

Table 6: Steel-Belted Radius Carrier Configuration Files

File	Function
*.acc	Configures the SQL accounting method. Also configures Call Detail Record accounting when using the SIM authentication module.
*.att	Configures attribute handling files used in CDMA
*.aut	Configures a Steel-Belted Radius Carrier authentication method.

Table 6: Steel-Belted Radius Carrier Configuration Files (*continued*)

File	Function
*.dcm	Primary list of dictionary files.
*.dct	Vendor-specific dictionary file.
*.dhc	Configures specific DHCP address pools, where * is the name of an address pool listed in dhcp.ini .
*.dic	XML format of *.dct dictionary files.
*.dir	Configures directed authentication and directed accounting realms.
*.gen	Used in configuration of the Steel-Belted Radius Carrier optional modules such as the option SIM authentication module.
*.rr	Configures attribute value pools.
*.pro	Configures proxy realms.
access.ini	Maps user or group account levels to administrative permissions. Used with admin.ini to grant administrators access privileges to administrative objects and actions.
account.ini	Controls how RADIUS accounting attributes are logged.
admin.ini	Maps administrative access levels to sets of access rights. Used with access.ini to grant administrators access privileges to administrative objects and actions.
authlog.ini	Controls how RADIUS authentication requests are logged by Steel-Belted Radius Carrier.
authReport.ini	Controls what authentication logs Steel-Belted Radius Carrier generates.
authReportAccept.ini	Controls options for the acceptance authentication log file.
authReportBadShared Secret.ini	Controls options for the invalid shared secret authentication log file.

Table 6: Steel-Belted Radius Carrier Configuration Files (*continued*)

File	Function
authReportReject.ini	Controls options for the rejection authentication log file.
authReportUnknownClient.ini	Controls options for the unknown client authentication log file.
blacklist.ini	Configures blacklist settings, which are used to block authentication requests that match a blacklist profile.
classmap.ini	Specifies what Steel-Belted Radius Carrier does with RADIUS attributes encoded in one or more Class attributes included in accounting requests.
dhcp.ini	Configures DHCP address pools so that IP addresses can be assigned from a back-end DHCP server.
eap.ini	Configures EAP authentication methods used by Steel-Belted Radius Carrier.
events.ini	Configures dilution, suppression, and threshold settings for SBR Carrier traps (except for Diameter-related traps) used to signal failures and warnings.
events.xml	Configures dilution and suppression settings for Diameter-related traps used to signal failures and warnings.
filter.ini	Sets up rules for filtering attributes into and out of RADIUS packets.
ldapauth.aut	Specifies settings for LDAP authentication in Steel-Belted Radius Carrier.
lockout.ini	Configures settings that lock user accounts after repeated failed login attempts.
peapauth.aut	Configures the EAP-PEAP authentication method.
proxy.ini	Stores information that applies to all realms on the server.
proxyrl.ini	Configures list of realms for forwarding accounting packets.

Table 6: Steel-Belted Radius Carrier Configuration Files (*continued*)

File	Function
radius.ini	Configures a variety of operational settings for Steel-Belted Radius Carrier.
radsql.acc	Configures Oracle SQL accounting for Steel-Belted Radius Carrier.
radsql.aut	Configures Oracle SQL authentication for Steel-Belted Radius Carrier.
radsqjdbc.acc	Configures JDBC SQL accounting for Steel-Belted Radius Carrier.
radsqjdbc.aut	Configures JDBC SQL authentication for Steel-Belted Radius Carrier.
redirect.ini	Configures settings that redirect users after repeated failed login attempts.
servtype.ini	Configures service type mappings, which allow a user to have multiple authorization attribute sets based on the service type the user is requesting.
sessionTable.ini	Stores any attribute in a request in the CST.
spi.ini	Defines encryption keys and identifies the servers from which Steel-Belted Radius Carrier processes encrypted Class attributes in accounting requests.
statlog.ini	Configures the Steel-Belted Radius Carrier statistics log file, which periodically records server statistics to a comma-delimited ASCII file.
tlsauth.aut	Configures the TLS authentication method.
tlsauth.eap	Configures the operation of the TLS helper method.
ttlsauth.aut	Configures the TTLS authentication method.
uniport.aut	Configures the UniPort authentication method.

Table 6: Steel-Belted Radius Carrier Configuration Files (*continued*)

File	Function
update.ini	Controls what information is updated when Steel-Belted Radius Carrier receives a SIGHUP (1) or SIGUSR2 (17) signal.
vendor.ini	Maps vendor-specific dictionary files to identifiers used in the Steel-Belted Radius Carrier administrative database.

Tips for Editing Configuration Files

When editing configuration files, observe the following guidelines:

- Configuration files are text files that you can edit using a standard text editor. If you use a word processing application such as Microsoft Word to edit your configuration files, make sure that you save the modified file in ASCII text format with UNIX line endings.
- Make a backup copy of your configuration files before you make any changes, so that you have a working archive copy in the event that you delete or misconfigure an important setting and want to revert to your previous configuration.
- Configuration files usually contain one or more sections, denoted with a [SectionName] line. Each section contains one or more settings, which typically take the format **SettingName** = **SettingValue**. In general, the **SettingName** text is not case-sensitive (**ENABLE** = **1**, **Enable** = **1**, and **enable** = **1** are all valid), but the **SettingValue** text is case-sensitive. If a setting is commented out or omitted, *Steel-Belted Radius Carrier* uses the default value for the setting.
- You can enter comments in configuration files by starting the line containing the comment with a semicolon (;) as the first character of the line. To disable a setting, consider commenting it out (by putting a semicolon at the start of the line) instead of deleting it.

NOTE: Commenting out a section header [SectionFoo] does not automatically comment out the parameters of that section. In the following example, the [SectionFoo] parameters are interpreted by the software in the same way as the [SectionBar] parameters:

```
[SectionBar]
BarParameter1 = click
BarParameter2 = clack

;[SectionFoo]
FooParameter1 = bit
FooParameter2 = byte
```

- Put comments on a separate line above or below configuration settings. You cannot include comments on the same line as a configuration setting.

Correct:

```
;Set to 0 on 5/30/2006
Session_Timeout = 0
```

Incorrect:

```
Session_Timeout = 0 ; Set to 0 on 5/30/2006
```

- The default configuration files provided with Steel-Belted Radius Carrier typically include section headers and settings that are commented out. In such cases, Steel-Belted Radius Carrier uses the value shown in the commented setting as the default, meaning that you do not need to change the setting if you want to use the default value.

To change the value for a setting to something other than the default value, you must uncomment the setting by removing the semicolon at the start of the line. The section headers (in square brackets) must also be uncommented for settings to be processed correctly.

- Make sure that lines containing settings or section headers have a text character in the first column. If a line has white space in the first column, it might not be processed correctly.
- If you mistype a setting or enter an invalid setting in a configuration file, Steel-Belted Radius Carrier ignores that setting. It is recommended that you avoid entering any invalid text in configuration files because the text may accidentally interfere with settings reserved for developers.
- You can edit configuration files while Steel-Belted Radius Carrier is running. However, changes to some files, such as **radius.ini**, require that you execute a HUP command or restart Steel-Belted Radius Carrier for the changes to take effect.

2

PART

Steel-Belted Radius Carrier Core and Radius Front-End Files

Operations Files | **9**

Authentication Configuration Files | **133**

Attribute Processing Files | **187**

Address Assignment Files | **239**

Accounting Configuration Files | **247**

Realm Configuration Files | **262**

EAP Configuration Files | **308**

Session State Register (SSR) Configuration Files | **361**

Operations Files

IN THIS CHAPTER

- [access.ini File | 9](#)
- [admin.ini File | 11](#)
- [events.ini File | 18](#)
- [events.xml File | 22](#)
- [radius.ini File | 24](#)
- [sbrd.conf File | 104](#)
- [services File | 116](#)
- [servtype.ini File | 117](#)
- [update.ini File | 120](#)
- [Auto-Restart Files | 126](#)

This chapter describes the usage and settings for files used in Steel-Belted Radius Carrier operations and administration. These topics are included in this chapter:

access.ini File

The **access.ini** file maps operating system user or group account names to levels of administrative privilege. The user account name and password used by an administrator when interacting with the Steel-Belted Radius Carrier server is granted access privileges according to the settings in this file.

[Settings] Section

The [Settings] section of **access.ini** contains overall configuration parameters; do not edit this section.

Table 7: access.ini [Settings] Syntax

Parameter	Function
<i>Method</i>	<p>This parameter controls the database against which the user is authenticated for access.</p> <p>If set to OS, authentication is done against the local operating system database such as /etc/passwd.</p> <p>If set to PAM, authentication is done against the PamService such as LDAP database.</p> <p>The default value is OS.</p> <p>The PamService setting is used to specify the service name, which is mapped to an entry in /etc/pam.conf on Solaris or /etc/pam.d/<name> on Linux.</p> <p>NOTE: To perform PAM authentication on a Linux device, you must install 32-bit binaries of pam_ldap—for example, pam_ldap-185-11.el6.i686—on the Steel-Belted Radius Carrier server. Steel-Belted Radius Carrier does not support pam_ldap.x86_64 binaries.</p>

[Users] and [Groups] Sections

The syntax for the [Users] and [Groups] sections ([Table 8 on page 11](#)) of the **access.ini** file is:

```
[Users]
UserName = AccessLevel
_system.localhost = SnmpAgent

[Groups]
GroupName = AccessLevel
GroupName = AccessLevel
```

NOTE: If you use SNMP to monitor your Steel-Belted Radius Carrier server, the [Users] section of your **access.ini** file must contain this entry:

_system.localhost = SnmpAgent

If you are not using SNMP, comment out or delete the **_system.localhost = SnmpAgent** entry as a security precaution.

Table 8: access.ini Syntax

Parameter	Function
<i>UserName</i> <i>GroupName</i>	<p>Each <i>UserName</i> or <i>GroupName</i> is the name of an authorized administrator account on the server. <i>UserName</i> and <i>GroupName</i> refer to Solaris /etc/passwduser/group.</p> <p>You must list user accounts in the [Users] section and group accounts in the [Groups] section. List groups in priority order; rights are granted based on the first group found of which the user is a member.</p>
<i>AccessLevel</i>	<p>The <i>AccessLevel</i> in each access.ini entry is the access level that you want to assign to that account.</p> <p>Each <i>AccessLevel</i> string must match the name of an [AccessLevel] section in admin.ini. You can define as many [AccessLevel] sections as you require. After an [AccessLevel] section is defined in admin.ini, you can use access.ini to assign the access privileges associated with that level to users and group accounts.</p>

NOTE: Adding a user as an administrator using the Web GUI overrides any access settings specified for that user in the **access.ini** configuration file.

A special access level called **SuperAdmin** grants read/write access to all types of administrative data. This access level is always defined, and can be assigned to a user or group account in **access.ini** without appearing in **admin.ini**.

admin.ini File

The **admin.ini** file maps administrative access levels to sets of access rights. These access levels are enforced for administrators connecting to Steel-Belted Radius Carrier by means of the Web GUI or LDAP

configuration interface (LCI). Each [AccessLevel] section in the **admin.ini** file corresponds to an *AccessLevel* name entered in the **access.ini** file. You can create as many [AccessLevel] sections in the **admin.ini** file as you require.

Access rights are defined according to the categories of administrative data that an account is allowed to read and write. These data categories correspond to Web GUI pages and to objects directly under **o=radius** in the LDAP configuration schema.

NOTE: Due to interdependencies in configuration, to enable an administrator to configure users, the following settings are required in the [AccessLevel] section of the **admin.ini** file:

```
[AccessLevel]
Users=rw
IP-Pools=r
Profiles=r
```

NOTE: If you omit a keyword, access to that data category is specifically denied for all information and dialogs that correspond to that keyword. Misspelled keywords are considered omitted.

[AccessLevel] Section

The syntax for each [AccessLevel] section ([Table 9 on page 13](#)) defined in the **admin.ini** file is:

```
[AccessLevel]
Access = value
Certificates = value
CCMPublish = value
CCMServerList = value
Configuration = value
CurrentUsers = value
ImportExport = value
IP-Pools = value
License = value
Profiles = value
Proxy = value
RAS-Clients = value
Report = value
RuleSets = value
```

```

Statistics = value
Tunnels = value
Users = value

```

Table 9: admin.ini Syntax

Parameter	Function
<i>AccessLevel</i>	Specifies the name of the access level. The value used here must be identical to the value used in the access.ini file.
Access	<p>Specifies whether administrators with this access level can read or write (update) administrative access data, which is controlled by the Administrators List page.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access <p>NOTE: When an administrator requests access, Steel-Belted Radius Carrier checks entries in the Administrators List page in Web GUI before checking the access.ini and admin.ini files. If an applicable administrative account exists in the Administrators List page, the user is given full access to the Steel-Belted Radius Carrier database, regardless of the configuration of the access.ini and admin.ini files.</p>
Certificates	<p>Specifies whether administrators with this access level can modify trusted root and server certificate information through Web GUI.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
CCMPublish	<p>Specifies whether administrators with this access level can publish server replication (ccmpkg) information through Web GUI. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access

Table 9: admin.ini Syntax (*continued*)

Parameter	Function
CCMServerList	<p>Specifies whether administrators with this access level can read or write (update) information in the Server List page in Web GUI. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
Configuration	<p>Specifies whether administrators with this access level can read or write (update) information found in the Authentication Methods page in Web GUI. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
CurrentUsers	<p>Specifies whether administrators with this access level can read or write (update) the Current Sessions Table, which can be displayed in the Reports page in Web GUI. Write access allows the administrator to delete entries from the Current Sessions Table. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access

Table 9: admin.ini Syntax (*continued*)

Parameter	Function
ImportExport	<p>Controls whether the Import and Export menu items are enabled in the Web GUI.</p> <ul style="list-style-type: none"> • Read access allows file export. • Write access allows file import. <p>Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access (allows export but not import) • w—Write-only access (allows import but not export) • rw—Read/write access (allows import and export) <p>Data categories without read access are disabled. If a user tries to export categories of data without having sufficient access rights, categories for which the user does not have read access are omitted from the export operation. Similarly, if a user tries to import categories of data without having sufficient access rights, categories for which the user does not have write access are omitted from the import operation.</p> <p>NOTE: Import and Export are subject to the particular rights that the user has to each type of item, such as Users or Tunnels.</p>
IP-Pools	<p>Specifies whether administrators with this access level can read or write (update) IP address pool data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access <p>NOTE: This applies to standalone SBR Carrier servers only. For information about IP pools on Session State Register servers, see the <i>SBR Carrier Installation Guide</i>.</p>
License	<p>Specifies whether administrators with this access level can add a new license. Valid values are:</p> <ul style="list-style-type: none"> • w—Write-only access • rw—Read/write access

Table 9: admin.ini Syntax (*continued*)

Parameter	Function
Profiles	<p>Specifies whether administrators with this access level can read or write (update) profile data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
Proxy	<p>Specifies whether administrators with this access level can read or write (update) proxy target data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
RAS-Clients	<p>Specifies whether administrators with this access level can read or write (update) RADIUS client data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
Report	<p>Specifies whether administrators with this access level can read or write (update) report data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
RuleSets	<p>Specifies whether certificates are replicated within a realm. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
Statistics	<p>Specifies whether administrators can read Authentication, Accounting, and Proxy statistics generated by the server. Write access is not applicable. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access

Table 9: admin.ini Syntax (*continued*)

Parameter	Function
Tunnels	<p>Specifies whether administrators with this access level can read or write (update) RADIUS tunnel data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access
Users	<p>Specifies whether administrators with this access level can read or write (update) user data. Valid values are:</p> <ul style="list-style-type: none"> • r—Read-only access • w—Write-only access • rw—Read/write access <p>NOTE: You must set the Users parameter to rw (read-write) for a user or group if you want the user or group to be able to import user information into Steel-Belted Radius Carrier.</p>

[SNMPAgent] Section

If you use SNMP to monitor your Steel-Belted Radius Carrier server, the [SNMPAgent] section of **admin.ini** file must include this section to give Read access to the SNMP agent.

```
[SNMPAgent]
RAS-Clients=r
Users=r
Profiles=r
Proxy=r
Tunnels=r
IP-Pools=r
Access=r
Configuration=r
Statistics=r
CurrentUsers=r
Report=r
ImportExport=r
License=r
```

events.ini File

The **events.ini** configuration file configures dilution, suppression, and threshold settings for Steel-Belted Radius Carrier SNMP traps (except for Diameter-related traps) in the **fnkradtr.mib** and **fnkradtr-v2.mib** MIBs that communicate failures, warnings, and other information.

“[SNMP Traps and Statistics Overview](#)” on [page 565](#) summarizes common event values. Only some of these events support thresholds or dilution.

[EventDilutions] Section

The [EventDilutions] section ([Table 10 on page 18](#)) of **events.ini** specifies how many events must occur before Steel-Belted Radius Carrier generates an event report. This feature lets you dilute the rate at which frequently occurring events are logged.

The syntax is:

```
[EventDilutions]
EventName=DilutionCount
```

Where *EventName* identifies a Steel-Belted Radius Carrier event and *DilutionCount* specifies how many times this event must occur before it is reported to the SNMP manager program.

Table 10: events.ini [EventDilutions] Syntax

OID	EventName	Function
5002	ConcurrencyServerFailure	Concurrency server returned failure indication. This event represents <i>nnnn</i> failures. The default dilution count is 100.
5003	ConcurrencyServerTimeout	Timed out in proxy attempt to the concurrency server. This event represents <i>nnnn</i> requests timing out. The default dilution count is 100.
5004	ConcurrencyProxyLocalFailure	Local failure encountered in an attempt to proxy to the concurrency server. This event represents <i>nnnn</i> requests timing out. The default dilution count is 100.

Table 10: events.ini [EventDilutions] Syntax (continued)

OID	EventName	Function
5005	StaticAcctProxyTimeout	Timed out in static accounting and smart static proxy attempts. This event represents <i>nnnn</i> failures. The default dilution count is 100.
5006	StaticAcctProxyFailure	Local failure encountered in an attempt to proxy for static accounting and smart static. This event represents <i>nnnn</i> requests timing out. The default dilution count is 100.
5008	SQLConnectFailure	<i>nnnn</i> attempts to connect to the SQL server failed. The default dilution count is 10.
5009	SQLDisconnect	<i>nnnn</i> disconnects from the SQL server because of an error. The default dilution count is 100.
5010	SQLRequestTimeout	<i>nnnn</i> timeouts on SQL requests. The default dilution count is 10.
5011	AcctDatabaseTimeout	Access to the accounting server database has timed out. This event represents <i>nnnn</i> timeouts. The default dilution count is 100.
5012	AcctDatabaseFailure	Access to the accounting server database has failed. This event represents <i>nnnn</i> failures. The default dilution count is 100.
5016	LDAPConnectFailure	<i>nnnn</i> attempts to connect to the LDAP server failed. The default dilution count is 100.
5017	LDAPDisconnects	<i>nnnn</i> disconnects from the LDAP server because of an error. The default dilution count is 100.

Table 10: events.ini [EventDilutions] Syntax (*continued*)

OID	EventName	Function
5018	LDAPRequestTimeout	<i>nnnn</i> timeouts on LDAP requests. The default dilution count is 100.
5022	ProxySpoolerTimeout	<i>nnnn</i> proxy accounting spooler timeouts. The default dilution count is 100.
5026	ACCTWriteFailure	<i>nnnn</i> failures to commit accounting data to a persistent store such as the file system, database, and so on. The default dilution count is 100.
5030	FloodQueueOverflow	<i>nnnn</i> packets have been dropped due to the flood queue limit being exceeded.
10045	MemoryFailure	<i>nnnn</i> memory allocation failures have occurred. The default dilution count is 1000.

Example

This example specifies that a *Steel-Belted Radius Carrier* server configured to authenticate against a SQL database reports every fifth SQLConnectFailure (trap OID 5008) error:

```
[EventDilutions]
; 5008 - nnnn attempts to connect to SQL server failed.
SQLConnectFailure=5
```

If an SQL error condition prevents the server from connecting to the database, Steel-Belted Radius Carrier retries the connection and reports these attempts in the server log file (**yyyymmdd.log**). Steel-Belted Radius Carrier does not trigger warning event 5008 until the fifth connection attempt fails.

[Suppress] Section

The [Suppress] section of **events.ini** lets you suppress Steel-Belted Radius Carrier events. An event whose trap number appears in this section is not reported when the applicable informational, warning, or error condition occurs.

Example

These settings suppress events relating to SQL database disconnections (5009) or LDAP server disconnections (5017).

```
[Suppress]

5009
5017
```

[Thresholds] Section

The [Thresholds] section ([Table 11 on page 21](#)) of the **events.ini** file lets you specify thresholds that trigger an event report. Thresholds often come in pairs, where a warning event is generated when a resource becomes scarce (low threshold is crossed), and an information event is generated when the resource becomes available (high threshold is crossed).

The [Thresholds] section lets you tune Steel-Belted Radius Carrier event generation for items such as system memory, thread count, and file system space, and can differ for each computer depending on resources, configuration, and other applications.

Table 11: events.ini [Thresholds] Syntax

Parameter	Function
ThreadAvailWarningIssue	When the number of available accounting or authentication threads reaches the specified value, issue the warning event RADMSG_THREADS_LOW or funkSbrTrapLowThreads (5001). Default value is 10.
ThreadAvailWarningClear	When the number of available accounting or authentication threads reaches the specified value, issue the informational event RADMSG_THREADS_NORMAL or funkSbrTrapThreadsNormal (102). Default value is 20.
FileSystemFreeKBWarningIssue	When available system disk space falls to the specified value, issue the warning event RADMSG_FILE_SYSTEM_LOW or funkSbrTrapLowFSSpace (5007). Default value is 4096 KB (4 MB).

Table 11: events.ini [Thresholds] Syntax (*continued*)

Parameter	Function
FileSystemFreeKBWarningClear	<p>When the number of kilobytes of available system disk space reaches the specified value, issue the informational event RADMSG_FILE_SYSTEM_NORMAL or funkSbrTrapFSNormal (103).</p> <p>Default value is 8092 KB (8 MB).</p>
ReserveMemoryKB	<p>Reserve this amount of memory (in kilobytes) at system startup for cases of overload. If a memory allocation failure occurs, Steel-Belted Radius Carrier frees the reserved memory and reports the event.</p> <p>Default value is 2048 KB (2 MB).</p>
PoolPctAddressAvailWarningIssue	<p>When the number of available addresses in any IP address pool drops below the specified percentage, issue a funkSbrTrapIPAddrPoolLow warning.</p> <p>Default value is 20 percent.</p>
PoolPctAddressAvailWarningClear	<p>When the number of available addresses in any IP address pool rises above the specified percentage, issue an informational message.</p> <p>Default value is 40 percent.</p>

Example

This example produces a warning event (5001) when the number of available accounting or authentication threads falls below 10 percent, and an informational event (102) when it rises above 20 percent.

```
[Thresholds]
ThreadAvailWarningIssue=10
ThreadAvailWarningClear=20
```

events.xml File

The **events.xml** configuration file is used to dilute and suppress Diameter traps that are defined in the **jnx-diameter-base-protocol.mib** MIB. [Table 12 on page 23](#) describes the mandatory Diameter trap attributes

allowed in the **events.xml** file. If you want to dilute and suppress a Diameter trap, you need to define the corresponding attributes for the Diameter trap.

Table 12: Attributes Allowed in the events.xml File

Attribute Name	Description
dilutionFactor	Specifies how many times a Diameter event must occur before it is reported to the SNMP manager program. The default dilution count is 1.
enable	Specifies whether to suppress a Diameter trap. <ul style="list-style-type: none"> • false—Suppresses the Diameter trap. • true—Does not suppress the Diameter trap. Default value is false.
dilutionFactorChangeable	Specifies whether to enable or disable changing the dilution count. <ul style="list-style-type: none"> • true—Enables changing the dilution count. • false—Disables changing the dilution count and enables SBR carrier to use only the default dilution count. Default value is true.

For more information about common event values, see [“SNMP Traps and Statistics Overview” on page 565](#). [Table 13 on page 23](#) lists the Diameter traps that can be diluted using the **events.xml** file.

Table 13: Diameter Traps That Can Be Diluted

OID	Trap Name	Description
1	jnxDbpProtocolError	Diameter base protocol error has occurred.
2	jnxDbpTransientFailure	Diameter transient failure has occurred.
3	jnxDbpPermanentFailure	Diameter permanent failure has occurred.
4	jnxDbpPeerConnectionDown	Connection between the Diameter peer and SBR Carrier is down.
5	jnxDbpPeerConnectionUp	Connection is established between the Diameter peer and SBR Carrier.
5000	jnxAAATrapUnauthorizedAdminRequest	A request from the administrator interface is denied.

Table 13: Diameter Traps That Can Be Diluted (*continued*)

OID	Trap Name	Description
10000	jnxAAATrapInternalError	The Diameter Result-Code attribute value indicating internal error is set.
10001	jnxAAATrapLicenseCheckFailure	A disposition indicating license check failure is set.
10002	jnxAAATrapResourceFailure	A disposition indicating resource failure is set.
10003	jnxAAATrapLogFileFailure	Attempt to open, create, or write a log file encounters a failure.

In the following example, the jnxDbpPermanentFailure trap (OID 3) is diluted to be sent once for every five occurrences of Diameter permanent failure.

```
<event id=".1.3.6.1.4.1.2636.8.1.2.1.0.0.3" description="diameter permanent
failure">
  <sinks>
    <sink name="trap" enable="true" dilutionFactor="5" dilutionFactorChangeable="true"
    />
  </sinks>
</events>
```

radius.ini File

The **radius.ini** initialization file is the main configuration file that determines the operation of Steel-Belted Radius Carrier. It contains information that controls a variety of Steel-Belted Radius Carrier functions and operations.

[Addresses] Section

By default, the Steel-Belted Radius Carrier server tries to auto configure all IPv4 addresses that are reported by name services for the primary hostname of the server on which Steel-Belted Radius Carrier is running, so that it can listen for incoming RADIUS packets on all available network interfaces. If IPv6 is enabled, Steel-Belted Radius Carrier auto configures its IPv6 addresses and then listens on all interfaces using IPv6 addresses.

Explicitly configure the IP addresses that you want Steel-Belted Radius Carrier to use in the [Addresses] section of **radius.ini** if Steel-Belted Radius Carrier is running on a multi-homed (more than one network interface) server and if any of these statements apply to your network:

- One or more network interfaces on the server are connected to networks that you do not want to carry RADIUS traffic.
- The server has more than one hostname, and IP addresses exist for names other than the primary hostname.
- The server has private IP addresses that are not published by name services.

Specifying IPv4 or IPv6 addresses causes the server to listen on only those addresses and ignore all other addresses.

Specifying **AutoConfigureIPv4** or **AutoConfigureIPv6** causes Steel-Belted Radius Carrier to attempt to discover and configure all IPv4 or IPv6 addresses that belong to the local host automatically.

Example 1

This example configures Steel-Belted Radius Carrier to listen for RADIUS authentication and accounting requests on the IPv4 address 192.168.12.35 and on all local IPv6 interfaces. IPv6 functionality must be enabled (by setting Enable to 1 in the [IPv6] section of **radius.ini**) before IPv6 addresses can be used.

```
[Addresses]
192.168.12.35
AutoConfigureIPv6
```

To route all of your proxy traffic through a single interface, set the value for **ProxySource** in the [Configuration] section of **radius.ini** to the appropriate IP address or addresses, which must be listed in the [Addresses] section.

Example 2

This example routes all proxy traffic through the interface at 192.10.20.30:

```
[Addresses]
192.10.20.30
192.10.20.31

[Configuration]
ProxySource = 192.10.20.30
```

The **ProxySource** setting in the [Configuration] section of **radius.ini** disables per-realm control of proxy outbound interfaces. If **ProxySource** is not set, sockets are opened and bound for each interface on the server. To route different proxy realms through specific interfaces using the **proxy.ini** file, refer to [“\[Interfaces\] Section” on page 278](#).

[AuditLog] Section

The [AuditLog] section ([Table 14 on page 26](#)) specifies whether Steel-Belted Radius Carrier maintains an audit log file (**audityyyymmdd.xml**) to record administrator activities and CCM events. Audit log records are stored in XML format in the **radius/audit** directory.

Administrator activities include:

- Logging in and out by Steel-Belted Radius Carrier administrators
- Creating, modifying, and deleting Steel-Belted Radius Carrier objects (RADIUS clients, users, profiles, proxy targets, proxy realms, tunnels, administrators, authentication policies, or CCM nodes)
- Importing files

CCM events include publication, notification, and download of CCM files.

NOTE: The audit log does not track changes made through the LDAP configuration interface (LCI).

```
[AuditLog]
;Enable = 0
;LogfilePermissions = owner:group mode
;DaysToKeep = 30
```

Table 14: radius.ini [AuditLog] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">• If set to 0, audit logging is disabled.• If set to 1, audit logging is enabled. Default value is 0.

Table 14: radius.ini [AuditLog] Syntax (*continued*)

Parameter	Function
LogFilePermissions	<p>Specifies the owner and access permission setting for the audit log (audityyyymmdd.xml) file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the audit log file, members of group 1007 can read (but not edit) the audit log file, and other users cannot access the audit log file.</p>
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains each authentication acceptance report.</p> <p>Default value is 30 days.</p>

[AuthRejectLog] Section

You configure the [AuthRejectLog] section ([Table 15 on page 28](#)) of **radius.ini** to specify what types of authentication method rejection messages Steel-Belted Radius Carrier records in the server log file (**yyyymmdd.log**). You can specify that you want the server log file to record reject information generated by all authentication methods, reject information of one or more specific types, or the most relevant rejection information.

Processing an authentication request might result in multiple instances of an authentication method being given a chance to authenticate the user. If this occurs and at least one authentication method succeeds in authenticating the user, no messages are recorded to the server log file. If this occurs and all instances fail to authenticate the user, you can specify that only the most relevant reason for the authentication failure is recorded. For example, if one method resulted in an authentication error of type **InvalidCredentials** and another results in an authentication error of type **SystemError**, only the **InvalidCredentials** message is logged.

You can specify that more than one type of log message be recorded by entering more than one filter type value for the Filter parameter.

Table 15: radius.ini [AuthRejectLog] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, authentication reject details are not recorded in the server log file. • If set to 1, authentication reject details of the specified types are recorded in the server log file. <p>Default value is 0.</p>
Filter	<p>Specifies the types of authentication reject messages to be recorded:</p> <ul style="list-style-type: none"> • All—Record authentication rejection details from all authentication methods. • MostRelevant—When multiple authentication methods are tried and all fail, record the most relevant error messages (the messages with the greatest severity). If two messages have the same severity, both are listed. <p>These values are listed in order of greatest to least relevance:</p> <ul style="list-style-type: none"> • PostProcessRejection—User was authenticated successfully but post processing caused rejection. • InvalidCredentialsOrUser—User was not authenticated because user was not found or credentials were invalid. • InvalidCredentials—User was not authenticated because user was known but the password or certificate was not correct. • UnsupportedCredentialType—User was not authenticated because the credentials presented were of the wrong type. • UserNotFound—User was not authenticated because user cannot be found in the authentication database. • AccessError—Authentication failed because a database or remote server was inaccessible. • InvalidRequest—User was not authenticated because the request appeared to be malformed. • BlacklistedUser—User was not authenticated because user is blocklisted. • SystemError—User was not authenticated because of a system error such as a resource allocation error.

This example causes authentication reject details from all authentication methods to be recorded to the server log file:

```
[AuthRejectLog]
Enable = 1
Filter = All
```

This example causes all authentication reject details of type `SystemError` to be recorded:

```
[AuthRejectLog]
Enable = 1
Filter = SystemError
```

This example causes all authentication reject details of type `SystemError`, `BlacklistedUser`, or `UserNotFound` to be recorded:

```
[AuthRejectLog]
Enable = 1
Filter = SystemError, BlacklistedUser, UserNotFound
```

[Configuration] Section

The [Configuration] section (Table 16 on page 29) of `radius.ini` contains parameters that control basic behavior of Steel-Belted Radius Carrier.

Table 16: radius.ini [Configuration] Syntax

Parameter	Function
AcctAutoStopEnable	<p>The Proxy AutoStop feature forwards session termination information to downstream proxy RADIUS servers when a user session is closed, so that the resources associated with the user session can be freed.</p> <p>You can set the AcctAutoStopEnable parameter value as Enabled, Disabled, or NoOnOff.</p> <ul style="list-style-type: none">• Enabled—Proxy AutoStop feature is enabled.• Disabled—Proxy AutoStop feature is disabled.• NoOnOff—Proxy AutoStop feature is enabled, but prevents Accounting-Stop packets from being sent in response to Accounting-On or Accounting-Off received from a NAS. <p>The default value is Disabled.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Acct-Flood-Queue-Shape	<p>Type of queuing used for accounting requests. You can use one of the following values:</p> <ul style="list-style-type: none"> • FIFO • LIFO • RAND <p>Default value is LIFO.</p>
Acct-Receive-Realtime-Thread-Priority	<p>Accounting requests (as well as proxy responses) are received on two separate threads. The priority of these threads can be set as follows:</p> <ul style="list-style-type: none"> • TS (timeshare)—This is the default class for processes and their associated kernel threads. The actual priority number to be used within this class range from 0 through 59, and are dynamically adjusted in an attempt to allocate processor resources evenly. • RT (real-time)—Threads in the RT class are fixed-priority, with a fixed time quantum. The actual priority number to be used range from 100 through 159, so a RT thread will preempt a system thread. <p>SBR sets the priority to the lesser of the maximum possible RT priority and the configured value (0 = no change).</p> <p>NOTE: For Linux, the smaller the numerical value, the higher the priority.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
AckOnCookieFailure	<p>When this parameter is set to yes, SBR Carrier sends an acknowledgement back for every accounting request it receives.</p> <p>To configure an acknowledgment to be sent despite the error, set AckOnCookieFailure = WithoutSession (which will not create a new session for an Accounting_Start), or set AckOnCookieFailure = WithSession (which will create a new session).</p> <p>Default value for AckOnCookieFailure = No, which will neither create a new session nor send an Accounting-Ack.</p>
AddDestIPAddressAttrToRequest	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier does not add destination address information to RADIUS requests. • If set to 1, Steel-Belted Radius Carrier adds a Funk-Dest-IP-Address attribute identifying the IP address to which the RADIUS request was sent to the attributes in the packet. All processing that can be performed on an attribute included in the request packet, such as check list processing, can be performed on this attribute. <p>Default value is 0.</p> <p>If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius Carrier to strip the attribute from the request before forwarding the request to a downstream server.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
AddDestUDPPortAttrToRequest	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier does not add destination port information to RADIUS requests. • If set to 1, Steel-Belted Radius Carrier adds a Funk-Dest-UDP-Port attribute identifying the UDP port to which the RADIUS request was sent to the attributes in the packet. All processing that can be performed on an attribute included in the request packet, such as check list processing, can be performed on this attribute. <p>Default value is 0.</p> <p>If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius Carrier to strip the attribute from the request before forwarding the request to a downstream server.</p>
AddFunkClientGroupToRequest	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier does not add a Funk-Radius-Client-Group attribute to an incoming RADIUS request. • If set to 1, Steel-Belted Radius Carrier adds a Funk-Radius-Client-Group attribute to the RADIUS request. The value of the Funk-Radius-Client-Group attribute is set to the name of the client group. <p>Default value is 0.</p> <p>NOTE: Enable this option only if you configure RADIUS client groups in Web GUI. For more information about RADIUS client groups, refer to the <i>SBR Carrier Administration and Configuration Guide</i>.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
AddFunkLocationGroupIdToRequest	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier does not add a Funk-Location-Group-Id attribute to an incoming RADIUS request. • If set to 1, Steel-Belted Radius Carrier adds a Funk-Location-Group-Id attribute to an incoming RADIUS request if the request comes from a client in a configured location group. The value of the Funk-Location-Group-Id attribute is set to the name of the location group, which can be used for SQL, LDAP, and check list processing. <p>Default value is 0.</p>
AddSourceIPAddressAttrToRequest	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier does not add source address information to RADIUS requests. • If set to 1, Steel-Belted Radius Carrier adds a Funk-Source-IP-Address attribute identifying the IP address from which the RADIUS request was received to the attributes in the packet. All processing that can be performed on an attribute included in the request packet, such as check list processing, can be performed on this attribute. <p>Default value is 0.</p> <p>If you enable this attribute, the attribute is visible to the proxy module. If your environment proxies requests, you might want to configure Steel-Belted Radius Carrier to strip the attribute from the request before forwarding the request to a downstream server.</p>
AllowNoUserName	<ul style="list-style-type: none"> • If set to any value other than yes (case in-sensitive), any Access-Request without a User-Name attribute is rejected. <p>NOTE: The setting of this parameter, coupled with the setting of the CheckForEmptyUserName parameter, affects how the SBRC server processes RADIUS Access-Requests with no or empty User-Name attributes.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Apply-Login-Limits	<ul style="list-style-type: none"> • If set to yes, the maximum number of concurrent connections for each user is enforced, and connection attempts above the limit are rejected. • If set to no, connections above the limit are allowed, but an event is noted in the server log file. <p>Default value is yes.</p>
AttributeEdit	<ul style="list-style-type: none"> • If set to 1, the attribute editing (filters) feature for proxy and directed realms, and plug-ins is enabled. • If set to 0, the feature is disabled. <p>Default value is 1.</p>
AuthenticateOnly	<ul style="list-style-type: none"> • If set to 1, no response attributes are included in the response packet to an AuthenticateOnly (Service-Type 8) request. • If set to 0, the normal response attributes are included in the response. <p>Default value is 1.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
AuthorizeOnly	<p>By default, SBR Carrier does <i>not</i> accept Authorize-Only requests. SBR Carrier may be configured to accept them by setting AuthorizeOnly=1, in which case, to accept them, the request must also satisfy all of these conditions:</p> <ul style="list-style-type: none"> • The Access-Request contains the Service-Type attribute with a value=Authorize-Only • Message-Authenticator is present and valid • A session already exists in SBR Carrier for the AAA session ID (WiMAX) • At least one authentication method accepts the request. The authentication method (usually SQL or LDAP) must have the AcceptsAuthorizeOnly = 1 in the [Bootstrap] section. <p>NOTE: If set to 0, Authorize-Only requests are <i>not</i> accepted regardless of whether the authentication method (SQL, LDAP or other) has the AcceptsAuthorizeOnly = 1.</p> <p>NOTE: It is not meaningful for an EAP method to accept Authorize-Only requests. Authorize-Only processing does not include authentication, and this setting is only applied to single-step methods which have been configured not to perform authentication (for example, SQL or LDAP authentication).</p> <p>NOTE: SBR Carrier is only able to process Authorize-Only requests for WiMAX sessions because a session must be located in the current session table, indexed by a WiMAX session Id.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
AutoPasswords	<p>If set to yes, support for SHA and UNIXcrypt passwords for authentication against the native database are enabled.</p> <p>This feature may be used to test the use of passwords created with various encryption algorithms which are normally used by plug-ins such as LDAP and SQL. The algorithm is indicated by a token string enclosed by curly braces prepended to the password, for example, {md4}47476919506799271480 for an MD4-encoded password.</p> <p>Supported encryption algorithms and their token strings include message digest algorithm 4 (MD4) hash (md4), secure hash algorithm (SHA) 1 base 64 (sha), salted secure hash algorithm (SSHA) 1 base 64 (ssha), UNIX crypt (crypt), encmd5 (md5), and http digest md5 (http), as well as hex representation of ASCII password (hex).</p> <p>Default value is no (disabled).</p>
Auth-Flood-Queue-Shape	<p>Type of queuing used for authentication requests. You can use one of the following values:</p> <ul style="list-style-type: none"> • FIFO • LIFO • RAND <p>Default value is LIFO.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Auth-Receive-Realtime-Thread-Priority	<p>Authentication requests (as well as proxy responses) are received on two separate threads. The priority of these threads can be set as follows:</p> <ul style="list-style-type: none"> • TS (timeshare)—This is the default class for processes and their associated kernel threads. The actual priority number range from 0 through 59, and are dynamically adjusted in an attempt to allocate processor resources evenly. • RT (real-time)—Threads in the RT class are fixed-priority, with a fixed time quantum. The actual priority number range from 100 through 159, so a RT thread will preempt a system thread. <p>SBR sets the priority to the lesser of the maximum possible RT priority and the configured value (0 = no change).</p> <p>NOTE: For Linux, the smaller the numerical value, the higher the priority.</p>
AuthResponseOnCstFailure	<p>Specifies how the SBRC server responds to requests when the session database cannot be contacted.</p> <ul style="list-style-type: none"> • If set to Reject, the SBRC server sends an Access-Reject when there is a CST failure. • If set to Accept, the SBRC server sends an Access-Accept in spite of a CST failure. • If set to Discard, SBRC server does not send any response. <p>Default value is Reject.</p> <p>NOTE: When the ShutdownOnCSTFailure parameter in the dbclusterRPC.gen file is set to 0, this setting determines how incoming packets are handled when the SSR cluster is down.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
ChallengeCacheLimit	<p>This parameter controls the number of outstanding challenge state objects.</p> <p>Default value is 1000.</p> <p>NOTE: Value of 0 through 999 is interpreted as the minimum value of 1000.</p>
CheckTransactionIdInClass	<p>If set to 1, when matching an accounting request to a session record using the class attribute, the parameter verifies that both TxnId and DbClusterSessionId match.</p> <p>If set to 0, only DbClusterSessionId needs to match.</p> <p>Default value is 0.</p>
CheckForEmptyUserName	<ul style="list-style-type: none"> • If set to any value other than 0, Access-Requests without a value in the User-Name attribute are rejected. <p>The default is 1.</p> <p>NOTE: The setting of this parameter, coupled with the setting of the AllowNoUserName parameter, affects how the SBRC server processes RADIUS Access-Requests with no or empty User-Name attributes.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
CheckMessageAuthenticator	<p>Specifies whether validation of Message-Authenticator occurs on receipt of an Access-Request from a network access server or on receipt of an Access-Accept, Access-Reject, or Access-Challenge from a proxy.</p> <ul style="list-style-type: none"> • If set to 0, validation of received Message-Authenticator attributes is disabled. • If set to 1, validation is performed if the Message-Authenticator attributes are received. Message-Authenticator attributes must be present for EAP messages. • If set to 2, Message-Authenticator attributes are always required and always validated. If these attributes are not present, Steel-Belted Radius Carrier rejects the message. <p>For the WiMAX mobility module, set this to 1.</p> <p>Default value is 0.</p>
ClassAttributeStyle	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier uses unencrypted Class attributes with multiple ASCII keys in Access-Accept packets. • If set to 2, Steel-Belted Radius Carrier uses enhanced/encrypted Class attributes in Access-Accept packets. <p>Default value is 2.</p> <p>NOTE: The ClassAttributeStyle parameter must be set to a value of 2 before you can use attribute embedding. For information about attribute embedding, see “[EmbedInClass] Section” on page 72.</p>
ConvertCallingStationId	<ul style="list-style-type: none"> • If set to 1, the Calling-Station-Id is interpreted as a hex string. • If set to 0, the Calling-Station-Id is interpreted as ASCII. <p>Default value is 0.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
DelegatedIPv6PrefixPoolHint	<p>Specifies whether to treat the Delegated-IPv6-Prefix-Pool attribute as a hint.</p> <ul style="list-style-type: none"> • If set to Yes, the Delegated-IPv6-Prefix-Pool attribute is treated as a hint. If this attribute appears in both the Access-Request packet and the user's return list, the Delegated-IPv6-Prefix-Pool attribute values in both the Access-Request packet and the user's return list are returned. If this attribute does not appear in the Access-Request packet, the Delegated-IPv6-Prefix-Pool attribute value configured in the user's return list is returned. • If set to No, the Delegated-IPv6-Prefix-Pool attribute value configured in the user's return list is returned. <p>Default value is No.</p>
DisablePromptAttribute	<p>The Prompt attribute may be sent during an Access-Challenge. This parameter specifies whether or not to echo the user's response to the Access-Challenge on the client.</p> <ul style="list-style-type: none"> • 0 indicates the user's response to the Access-Challenge is not echoed. • 1 indicates the user's response to the Access-Challenge is echoed. <p>Default value is 0.</p> <p>Steel-Belted Radius Carrier uses the Prompt attribute during authentications. However, some clients do not respond properly to the Prompt attribute, so this parameter provides a way to disable it.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
DisableSecondaryMakeModelSelection	<ul style="list-style-type: none"> • If set to 1, SBR Carrier sets the make/model field with make/model information of the NAD that is found using the NAS-IP-Address attribute, NAS-IPv6-Address attribute, NAS-Identifier attribute, or source address in the received request. SBR Carrier searches for the NAD entry by using the attributes or values in the following order of preference: <ol style="list-style-type: none"> 1. NAS-IP-Address or NAS-IPv6-Address 2. NAS-Identifier 3. Source address • If set to 0, SBR Carrier sets the make/model field according to the proxy target that is being used for the RADIUS transaction. <p>NOTE: This setting affects only proxied packets. For a description about proxied packets, refer to the chapter <i>Administering Proxy RADIUS</i> in the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is 0.</p>
DiscardAccountingRequestOnCstFailure	<p>By default, accounting requests are acknowledged even if the SSR database cannot be contacted. This parameter specifies whether or not accounting requests should be discarded when the session database cannot be contacted, which may be desirable when using load balancing equipment.</p> <ul style="list-style-type: none"> • If set to 1, accounting requests (start, stop, on, off, and interim) are discarded when the session database cannot be contacted. • If set to 0, accounting requests (start, stop, on, off, and interim) are acknowledged when the session database cannot be contacted. <p>NOTE: When the ShutdownOnCSTFailure parameter in the dbclusterRPC.gen file is set to 0, this setting determines how incoming packets are handled when the SSR cluster is down.</p>

Table 16: radius.ini [Configuration] Syntax (continued)

Parameter	Function
DnsServerIPv6AddressHint	<p>Specifies whether to treat the DNS-Server-IPv6-Address attribute as a hint.</p> <ul style="list-style-type: none"> • If set to Yes, the DNS-Server-IPv6-Address attribute is treated as a hint. If this attribute appears in both the Access-Request packet and the user's return list, the DNS-Server-IPv6-Address attribute values in both the Access-Request packet and the user's return list are returned. If this attribute does not appear in the Access-Request packet, the DNS-Server-IPv6-Address attribute value configured in the user's return list is returned. • If set to No, the DNS-Server-IPv6-Address attribute value configured in the user's return list is returned. <p>Default value is No.</p>
DynAuthProxySource	<p>Specifies the IP address of the interface through which the outgoing proxy CoA/DM traffic is routed.</p> <p>The default value is 0.0.0.0. In this case, the interface is variable, chosen appropriately by the Operating System depending on the destination address.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
EnableWiMAXUniqueSessionIdFromNAI	<p>This parameter provides improvements to WiMAX performance and scalability. The improvements include different logic for assigning primary keys to WiMAX tables and for generating the Class attribute in the Access-Accept response.</p> <ul style="list-style-type: none"> • If set to 1, the EnableWiMAXUniqueSessionIdFromNAI parameter is enabled. • If set to 0, the EnableWiMAXUniqueSessionIdFromNAI parameter is disabled. <p>NOTE: When the EnableWiMAXUniqueSessionIdFromNAI parameter is enabled, new session records in the database and the Class attribute in Access-Accept messages are incompatible with the WiMAX logic in previous releases of SBR Carrier. For compatibility with SBR Carrier 7.2.1 and earlier, set EnableWiMAXUniqueSessionIdFromNAI = 0.</p> <p>Default value is 1.</p> <p>For details on migrating from existing SBRC WiMAX installations and new installations using WiMAX, see the section on <i>Migration and New Installations of SBR Carrier with WiMAX</i> in the <i>Migrating from Previous SBR Releases</i> section of the <i>SBR Carrier Installation Guide</i>.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
EnhancedRateStats	<p>Specifies whether support for calculating authentication, accounting, and proxy transaction rate statistics per NAD client and per Called-Station-ID is enabled or disabled.</p> <ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier enables the calculation of NAD client and Called-Station-ID specific rate statistics. In addition to the overall server specific rate statistics, you can view the rate statistics per NAD client and per Called-Station-ID through SNMP and LCI query. • If set to 0, Steel-Belted Radius Carrier disables the calculation of NAD client and Called-Station-ID specific rate statistics. You can view only the overall server specific rate statistics through SNMP and LCI query. <p>Default value is 0.</p>
EnumAttrsWithoutMvpFlagUpdate	<ul style="list-style-type: none"> • If set to 1, plug-ins can add attributes that are not flagged as reply-list attributes to an Access-Accept. • If set to 0, plug-ins cannot add attributes that are not flagged as reply-list attributes to an Access-Accept. <p>Default value is 1.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
FallbackLocal	<p>Specifies whether the session information is maintained in a local file on the SBR Carrier server when the working SSR cluster goes down or SBR Carrier fails to load the SSR cluster during startup.</p> <ul style="list-style-type: none"> • If set to true, session information is maintained in a local file on the SBR Carrier server when the working SSR cluster goes down or SBR Carrier fails to load the SSR cluster during startup. • If set to false, session information is not maintained in a local file on the SBR Carrier server even when the working SSR cluster goes down or SBR Carrier fails to load the SSR cluster during startup. <p>Default value is false.</p> <p>NOTE: This parameter is valid only if the PersistSessions parameter in the radius.ini file is set to 2 (NDB).</p>
ForceUpdate	<p>Enables Steel-Belted Radius Carrier to update the Current Session Table (CST) with additional attributes when an Accounting-Interim packet is received.</p> <p>NOTE: The ForceUpdate parameter is valid only if the UpdateOnInterim parameter is set to 1.</p> <p>Additional accounting attributes that can be updated include the following:</p> <ul style="list-style-type: none"> • User-Name • Called-Station-Id • NAS-Port • Calling-Station-ID • NAS-Port-Type <p>NOTE: The accounting attributes specified with this parameter must be separated by a space. For example, ForceUpdate= User-Name Calling-Station-ID</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
FramedIPAddressHint	<ul style="list-style-type: none"> • If set to Yes, the attribute Framed-IP-Address is treated as a hint. If this attribute appears in the Access-Request and the user's return list is configured to allocate Framed-IP-Address from a pool, the IP address in the Access-Request is returned instead of a newly-allocated IP address. • If set to No, the address is taken from the configured pool of addresses for Framed-IP-Address. The next available address is used. • If set to Check-Pool, the requested address is checked for validity against the pool of addresses for Framed-IP-Address. <p>Default value is no.</p> <p>NOTE: Hints are only applicable when SBR is configured to assign addresses from a pool.</p>
FramedIPv6AddressHint	<p>Specifies whether to treat the Framed-IPv6-Address attribute as a hint.</p> <ul style="list-style-type: none"> • If set to Yes, the Framed-IPv6-Address attribute is treated as a hint. If this attribute appears in both the Access-Request packet and the user's return list, the IPv6 address in the Access-Request packet is returned in the Access-Accept. If this attribute does not appear in the Access-Request packet, the IPv6 address configured in the user's return list is returned. • If set to No, the IPv6 address configured in the user's return list is returned. <p>Default value is No.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
IncRoutedProxyUsageCount	<ul style="list-style-type: none"> • If set to 1, the usage count is incremented both before the Access-Request is proxied, and when the proxy target responds with an Access-Accept. This is consistent with previous releases. • If set to 0, the usage count is incremented only when the proxy target responds with an Access-Accept, it is not incremented before the Access-Request is forwarded to the proxy. <p>Default value is 0.</p>
IpAddrFromClassAttr	<p>If set to Yes, SBR always adds the Framed-Ip-Address in the Access-Accept to the Class attribute, regardless of whether it is allocated from a pool.</p> <p>Default value is No.</p>
JvmMaxHeapSizeInMB	<p>Specifies the maximum Java heap memory the Java Virtual Machine (JVM) can allocate for processing JDBC connections. An out of memory error occurs when the memory exceeds the value configured in this parameter.</p> <p>Default value is 1024 MB.</p>
JvmMinHeapSizeInMB	<p>Specifies the minimum Java heap memory required for processing JDBC connections. If your system does not have the configured minimum memory, the JVM will not be initialized.</p> <p>Default value is 64 MB.</p>
JVMPath	<p>Specifies the location of the JVM used by JDBC plug-ins.</p> <p>NOTE: Do not edit this parameter manually. This parameter is automatically populated after running the SBR Carrier configuration script.</p> <p>For this parameter to work, make sure that this parameter is uncommented.</p>

Table 16: radius.ini [Configuration] Syntax (continued)

Parameter	Function
SerialNum	
and	
LegacyPluginConcurrency	

Table 16: radius.ini [Configuration] Syntax (continued)

Parameter	Function														
	<p>NOTE: PR:1468996 fix is available starting from SBR 8.6.0R13, full builds only.</p> <p>SBR 8.6.0R13 addresses the following limitation in previous builds.</p> <p>Limitation Description—If we consider the generic custom plug-ins like LDAP, TLS, TTLS, PEAP, SQL-JDBC, and ORACLE, the same Prefix ID ("200") is used.</p> <p>Let us consider a scenario, where same User-Name "test" is authenticated by SBR via both LDAP and ORACLE plug-ins. In the previous builds(with out fix)the same Prefix ID "200" will be shown in the <code>"/ShowUserConc -a"</code> output.</p> <p>To overcome this issue SBR should maintain unique ID's for each plug-in. The following are the updated Prefix IDs for each generic plug-in.</p> <table><tr><th>Component</th><th>New Plug-In ID</th></tr><tr><td>LDAP</td><td>400</td></tr><tr><td>TLS</td><td>500</td></tr><tr><td>TTLS</td><td>600</td></tr><tr><td>PEAP</td><td>700</td></tr><tr><td>SQL-JDBC</td><td>800</td></tr><tr><td>ORACLE</td><td>900</td></tr></table> <p>NOTE: The updated behavior will function only when the parameter "LegacyPluginConcurrency" is set to "False" .</p> <p>LegacyPluginConcurrency-If this parameter is set to "False" the latest plug-ins IDs will be used, else the SBR behavior will be similar to prior releases.</p>	Component	New Plug-In ID	LDAP	400	TLS	500	TTLS	600	PEAP	700	SQL-JDBC	800	ORACLE	900
Component	New Plug-In ID														
LDAP	400														
TLS	500														
TTLS	600														
PEAP	700														
SQL-JDBC	800														
ORACLE	900														

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
	<p>Default Value of "LegacyPluginConcurrency" is "False".</p> <p>SerialNum-The parameter is added to "[Bootstrap]" section of *.aut file of the generic plug-ins.</p> <ul style="list-style-type: none"> • The range are limited to 1 through 99. • By Default the parameter "SerialNumber" is commented. • [Bootstrap]; SerialNumber=0 <p>NOTE: In case of multiple plug-ins of the same type, the Ids can be differentiated by adding "SerialNumber" is configured in each corresponding "aut" file listed below.</p> <p>The Final Value of "Id" in the "./ShowUserConc -a" calculation is done like this:.</p> <p>Id = "New Plug-In ID" value + "SerialNum" configured in the *.aut file of the plug-in.</p> <p>If the above mentioned scenario of Limitation is considered, with the latest patch full build, we shall notice the listed output in the "./ShowUserConc.sh -a" .</p> <pre> Id value for ORACLE in "./ShowUserConc -a" = 900 (ORACLE ID) + 1 (SerialNum configured in *.aut file of ORACLE) = 901 Id value for LDAP in "./ShowUserConc -a" = 400 (LDAP ID) + 1 (SerialNum configured in *.aut file of LDAP) = 401 </pre>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function						
	<p>Table 17: hadm@<host_name>:~> ./ShowUserConc.sh -a UserConcurrency:</p> <table> <tr> <th>ID</th><th>Counr</th></tr> <tr> <td>901-Test</td><td>3</td></tr> <tr> <td>401-Test</td><td>4</td></tr> </table> <p>NOTE: Different values of <serialnum> should be used to differentiate different instances of the same generic plug-in, for example ldapauth1.aut and ldapauth2.aut. However, if the different instances use the same backend, <serialnum> should be the same to properly support concurrency limitations.</p>	ID	Counr	901-Test	3	401-Test	4
ID	Counr						
901-Test	3						
401-Test	4						
LookupClientByIPRange	<p>If set to 1, this parameter enables enumeration of NAS clients within IP ranges; set this to support Service-Type mapping of range-defined NAS clients.</p> <p>Default value is 0.</p>						
Login-Limit-Key	<p>Login-Limit-Key is valid only in a Session State Register cluster environment, and only then if the Optional Concurrency and Wholesale Module is installed. If both those conditions are met, the setting controls what user attribute or attributes are counted to determine concurrent session limit compliance.</p> <p>There is no default value. The setting may contain any user attribute string. Multiple attributes may be specified, separated by spaces, up to the 84-character limit of the field.</p>						

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
MainThreadStackSize	<p>Stack size of the main thread. This value specifies the number of bytes that is allocated to a main thread.</p> <p>Main threads are maintenance threads that perform such functions as stale session purging, signal handling, and statistics logging.</p> <p>Default value is 786432 KB.</p> <p>The MainThreadStackSize value must be greater than or equal to the default value.</p>
Max-Auth-Floods	<p>Maximum number of requests that can be stored in the authentication flood queue. You can enter the value in the range 0 to 10,000 * Max-Auth-Threads.</p> <p>Default value is 25.</p>
Max-Auth-Threads	<p>The maximum number of threads available to handle authentication requests. Minimum is 1, maximum is 1,000,000 (limited by memory).</p> <p>Default value is 100.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Max-Auth-Threads-In-Flood	<p>Maximum number of threads from the authentication thread pool that can support the flood queue concurrently. If this value equals the maximum number of threads in the authentication thread pool, all threads are available to serve new requests or queued requests. If this value is less than the maximum number of threads in the authentication thread pool, the difference represents the number of threads reserved for servicing only new authentication requests.</p> <p>When a specific value is not configured for this parameter, then by default this parameter is assigned with half the value of the maximum number of threads (Max-Auth-Threads) available to handle in the authentication requests.</p> <p>Default value is 50.</p> <p>NOTE: This default value is valid only if the default value of Max-Auth-Threads is unchanged.</p>
Max-Acct-Floods	<p>Maximum number of requests that can be stored in the accounting flood queue. You can enter the value in the range 0 through 10,000.</p> <p>Default value is 25.</p>
Max-Acct-Threads	<p>Maximum number of threads available to handle accounting requests. Minimum is 1; maximum is 1,000,000 (limited by memory).</p> <p>Default value is 200.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Max-Acct-Threads-In-Flood	<p>Maximum number of threads from the accounting thread pool that can support the flood queue concurrently. If this value equals the maximum number of threads in the accounting thread pool, all threads are available to serve new requests or queued requests. If this value is less than the maximum number of threads in the accounting thread pool, the difference represents the number of threads reserved for servicing only new accounting requests.</p> <p>When a specific value is not configured for this parameter, then by default this parameter is assigned with half the value of the maximum number of threads (Max-Acct-Threads) available to handle in the accounting requests.</p> <p>Default value is 100.</p> <p>NOTE: This default value is valid only if the default value of Max-Acct-Threads is unchanged.</p>
MaxEngines	<p>The MaxEngines parameter limits the number of Javascript host allocations that can be attempted. When set, worker threads wait for a host to become available. The optimum setting for this parameter may vary depending on the machine configuration and RADIUS traffic.</p> <p>The default is 0, and there is no limit.</p>
Max-Proxy-Floods	<p>Maximum number of requests that can be stored in the proxy flood queue. You can enter the value in the range 0 through 10,000.</p> <p>Default value is 25.</p>
Max-Proxy-Threads	<p>Maximum number of threads available to handle proxied accounting requests when Block=0 is set in the [Acct] section of the RealmName.pro file. Minimum is 1; maximum is 1,000,000 (limited by memory).</p> <p>Default value is 100.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
Max-Proxy-Threads-In-Flood	<p>Maximum number of threads from the proxy thread pool that can support the flood queue concurrently. If this value equals the maximum number of threads in the proxy thread pool, all threads are available to serve new requests or queued requests. If this value is less than the maximum number of threads in the proxy thread pool, the difference represents the number of threads reserved for servicing only new proxy requests.</p> <p>When a specific value is not configured for this parameter, then by default this parameter is assigned with half the value of the maximum number of threads (Max-Proxy-Threads) available to handle in the proxy requests.</p> <p>Default value is 50.</p> <p>NOTE: This default value is valid only if the default value of Max-Proxy-Threads is unchanged.</p>
NasClearRecordsBatchCount	<p>Specifies the number of sessions to be deleted per batch when an Accounting-On or Accounting-Off request is received from the NAD.</p> <p>Default value is 10,000.</p>
NoNullTermination	<ul style="list-style-type: none"> • If set to 0, RADIUS reply attributes of type string are sent with a null character at the end of the string (null terminated string). • If set to 1, RADIUS reply attributes of type string are sent without the null character at the end of the string. Entering a value of 1 for this setting is the equivalent of changing all reply attributes of type string to type stringnz. <p>Default value is 0.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
OverwriteCstDataOnFailure	<ul style="list-style-type: none"> • If set to 1, SBR overwrites the existing session record (based on IP address) on CST constraint violation. • If set to 0, SBR continues processing on CST constraint violation. <p>Default value is 0.</p>
PersistSessions	<p>Specifies how session persistence is maintained.</p> <ul style="list-style-type: none"> • If set to 0 (none), session information is not maintained when SBRC is restarted. This setting applies only to SBRC servers running in standalone mode. • If set to 1 (local), session information is maintained in a local file on the SBRC server, and is available after restarting the server. This setting applies only to SBRC servers running in standalone mode. • If set to 2 (NDB), session information is maintained in the SSR cluster database. This setting is applicable only when the server is running in a SBRC SSR cluster. <p>Default value is NDB.</p> <p>NOTE: You must set the PersistSessions parameter to 2 or NDB to use the Steel-Belted Radius Carrier high availability SSR cluster.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
PhantomTimeout	<p>Specifies the maximum number of seconds for a phantom session record. When a phantom session is created, its expiration timestamp (Sbr_ExpirationTime) is set to its creation timestamp (Sbr_CreationTime) plus the PhantomTimeout value. If a corresponding Accounting-Start or an interim accounting packet is received before the expiration timestamp, the phantom record is upgraded to active status, and its expiration timestamp is upgraded according to the StaleSessionTimeoutSecs setting. If no Accounting-Start or interim accounting packet is received before the expiration timestamp, the phantom record is purged according to settings for stale session purge threads. This highlights the importance of synchronizing clocks amongst SBR Carrier servers in a Session State Register cluster.</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>
ProcessRealmBeforeTunnel	<ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier checks whether a request matches the criteria established for tunnels before it tests whether a request matches the criteria for proxy and directed realms. • If set to 1, Steel-Belted Radius Carrier checks whether a request matches the criteria established for proxy and directed realms before it tests whether a request matches the criteria established for tunnels. <p>Default value is 0.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
ProxyFastFail	<p>Specifies the number of seconds a Steel-Belted Radius Carrier server continues to forward packets to a proxy RADIUS target that appears to be down.</p> <p>A value of 0 disables the feature.</p> <p>Default value is 300.</p> <p>NOTE: This parameter applies only to proxy targets that are used but not assigned to a realm.</p>
ProxySource	<p>Specifies the IP address of the interface through which all outgoing proxy traffic is routed. The IP address specified for ProxySource must be listed in the [Addresses] section of radius.ini.</p> <p>If a ProxySource address is not specified and per-realm control of proxy interfaces is not enabled, Steel-Belted Radius Carrier uses the first interface it finds on the server.</p>
ProxyStripRealm	<ul style="list-style-type: none"> • If set to 1, the proxy realm decoration is stripped before sending the request downstream. • If set to 0, no realm name stripping is performed. <p>Default value is 1.</p> <p>NOTE: This parameter applies only to proxy targets that are used but not assigned to a realm.</p>
Proxy-Flood-Queue-Shape	<p>Type of queuing used for proxy requests. You can use one of the following values:</p> <ul style="list-style-type: none"> • FIFO • LIFO • RAND <p>Default value is LIFO.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
RejectMalformedPacket	<p>Specifies whether to reject the RADIUS request if a malformed attribute is received in the request.</p> <ul style="list-style-type: none"> • If set to 1, SBR Carrier rejects the RADIUS request when a malformed attribute is received in the request. • If set to 0, SBR Carrier skips the malformed attribute and continues processing the RADIUS request when a malformed attribute is received in the request. However, if a packet is severely malformed, then the packet will be dropped. <p>Default value is 0.</p>
RouteIPv6InfoHint	<p>Specifies whether to treat the Route-IPv6-Information attribute as a hint.</p> <ul style="list-style-type: none"> • If set to Yes, the Route-IPv6-Information attribute is treated as a hint. If this attribute appears in both the Access-Request packet and the user's return list, the Route-IPv6-Information attribute values in both the Access-Request packet and the user's return list are returned. If this attribute does not appear in the Access-Request packet, the Route-IPv6-Information attribute value configured in the user's return list is returned. • If set to No, the Route-IPv6-Information attribute value configured in the user's return list is returned. <p>Default value is No.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
SelectIPPoolNameByNasAVPs	<ul style="list-style-type: none">• If set to 0, the IP address pool for a RADIUS client is based on the source IP address in the UDP packet containing the access request.• If set to 1, the IP address pool for a RADIUS client is based on the value of the NAS-IP-Address or NAS-Identifier attribute included in the access request. If the NAS-IP-Address or NAS-Identifier attribute is not present, or if a RADIUS client matching the IP address or identifier cannot be found, the IP address pool for a RADIUS client is based on the source IP address in the UDP packet containing the access request. <p>Default value is 0.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
SendOnlyOneClassAttribute	<p>When a user's identity information is encrypted during authentication, Steel-Belted Radius Carrier uses a special Class attribute to pass the user's encrypted identity to an accounting server. Because this typically requires more than one Class attribute to be included in the Accept response, and because some Access Points do not support echoing more than one Class attribute, you can use the SendOnlyOneClassAttribute parameter to specify how you want Steel-Belted Radius Carrier to forward encrypted user identity information.</p> <ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier creates a Class attribute containing a Class attribute flag, a server identifier, and a transaction identifier. The user identification data that is normally stored in the Class attributes is stored in the current sessions table. When Steel-Belted Radius Carrier receives an accounting request, it looks up the Class information in the current sessions table and uses it as if it had arrived in the accounting request packet. • If set to 0, Steel-Belted Radius Carrier creates one or more Class attributes to return a user's encrypted identity to the Access Point, with the assumption that the AP forwards the Class attribute(s) containing the encrypted user identification information to the accounting server. <p>Default value is 0.</p> <p>For the optional WiMAX mobility module, set this to 1.</p> <p>NOTE: This feature works only if accounting requests go to the same server or cluster that performs authentication. Accounting requests that go to servers other than the authenticating server fail.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
StaleSessionPurgeThreadChunkSize	<p>Specifies the number of stale sessions a SBR Carrier server purges at a time.</p> <p>Default value is 100 sessions.</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>
StaleSessionPurgeThreadSleepMax	<p>Specifies the maximum number of seconds the SBR Carrier server waits before purging stale sessions.</p> <p>Default value is 20 seconds.</p> <p>NOTE: In cluster configurations, each SBR Carrier server periodically purges stale sessions from the session database. To avoid having multiple servers in a cluster try to purge the same stale sessions simultaneously, the StaleSessionPurgeThreadSleepMin and StaleSessionPurgeThreadSleepMax settings provide a short random sleep interval for the stale session purge process.</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
StaleSessionPurgeThreadSleepMin	<p>Specifies the minimum number of seconds SBR Carrier waits before purging stale sessions.</p> <p>Default value is 10 seconds.</p> <p>NOTE: In cluster configurations, each SBR Carrier server periodically purges stale sessions from the Session State Register database cluster. To avoid having multiple servers in a cluster try to purge the same stale sessions simultaneously, the StaleSessionPurgeThreadSleepMin and StaleSessionPurgeThreadSleepMax settings provide a short random sleep interval for the stale session purge process.</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>
StaleSessionTimeoutSecs	<p>Specifies the lifetime for a session (phantom record for which a corresponding accounting start packet is received) in the Current Sessions Table before the session expiration timestamp runs out and the session resources are released.</p> <ul style="list-style-type: none"> • If set to 0, the upgraded phantom record never expires. • If set to a number greater than 0, specifies the number of seconds in the phantom record lifetime. <p>Default value is 86,400 seconds (one day).</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
StartupTimeout	<p>Specifies the number of seconds Steel-Belted Radius Carrier waits for its startup sequence to finish before timing out.</p> <p>Default value is 600 seconds.</p> <p>NOTE: You must set this parameter based on the size of the HST file. If the HST file size is in the range 1.3–2.5 GB, set this parameter to 1600 seconds. If the HST file size is greater than 2.5 GB, set this parameter to 1800 seconds.</p>
StatefulIPv6AddressPoolHint	<p>Specifies whether to treat the Stateful-IPv6-Address-Pool attribute as a hint.</p> <ul style="list-style-type: none"> • If set to Yes, the Stateful-IPv6-Address-Pool attribute is treated as a hint. If this attribute appears in both the Access-Request packet and the user's return list, the Stateful-IPv6-Address-Pool attribute values in both the Access-Request packet and the user's return list are returned. If this attribute does not appear in the Access-Request packet, the Stateful-IPv6-Address-Pool attribute value configured in the user's return list is returned. • If set to No, the Stateful-IPv6-Address-Pool attribute value configured in the user's return list is returned. <p>Default value is No.</p>
StoreClassInSession	<ul style="list-style-type: none"> • If set to always, the class attribute will always be stored in the CST. • If set to default, the class attribute will be stored in the CST depending on other attributes and configuration such as WiMAX mode. <p>Default value is default.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
TreatAddressPoolsAsDisjoint	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier treats each IP address pool as though it operates off its own disjoint address space. This disables the normal checks to ensure that an IP address is allocated only to a single address pool. • If set to 0, a single IP address can be allocated only to a single session and from a single IP address pool. <p>Default value is 0.</p> <p>NOTE: To track allocated resources, Steel-Belted Radius Carrier uses the Class attribute to track IP addresses. This attribute contains the IP pool name and IP address.</p> <p>This parameter is applicable only to standalone servers.</p>
UpdateOnInterim	<p>Specifies whether or not to update the session from phantom to active when the SBRC server receives an accounting packet with the Acct-Status-Type attribute set to a value of Interim-Update.</p> <ul style="list-style-type: none"> • If set to 1, the server changes the state of the session from phantom to active when it receives an interim update. • If set to 0, the server does not change the state of the session from phantom to active when an interim update is received. • If set to Update, the server changes the state of the session from phantom to active when it receives an interim update. The behavior is similar to setting the value as 1. • If set to Add, the server changes the state of the session from phantom to active when it receives an interim update. If a phantom session is not found, then a new session is created based on the values in the interim update. <p>The default value is 0.</p>

Table 16: radius.ini [Configuration] Syntax (continued)

Parameter	Function
UDP-Receive-Buffer-Kbytes	<p>Sets the buffer size of the UDP socket. This value specifies the number of bytes allocated to the UDP socket to process the incoming RADIUS requests.</p> <p>You can enter the value in the range from 256 KB through 16 MB. Default value is 512 KB.</p> <p>CAUTION: In some high load scenarios (more than 500 concurrent transactions per second), the default socket size is not sufficient to process the incoming requests, resulting in packets being dropped from the socket. In this case, you need to increase the buffer size to the maximum value to prevent the packets from being dropped from the socket.</p>
UseProfileCache	<ul style="list-style-type: none"> • If set to 0, user profile results are not cached. • If set to 1, user profile results are cached, improving performance when using profiles. <p>Default value is 0.</p>
UseUserCache	<ul style="list-style-type: none"> • If set to 0, native users entries are not cached. • If set to 1, native user are cached, improving performance when using Native Users. <p>Default value is 0.</p> <p>When caching is enabled, SBRC's memory usage grows until all the user records are cached in memory. The storage space for all native user data (names, passwords, and Attribute Value Pairs) should fit well within the SBRC process memory. Enabling this is most suitable for Native User entries which represent device types, virtual devices, or are otherwise limited in number. Enabling it will increase throughput significantly.</p>

Table 16: radius.ini [Configuration] Syntax (*continued*)

Parameter	Function
WorkerThreadStackSize	<p>Sets the worker thread stack size. The value reflects the number of bytes that will be allocated for each worker thread.</p> <p>Worker threads are created to support authentication, accounting, and proxy operations.</p> <p>Default worker thread stack size value is 1 MB.</p> <p>NOTE: The required memory may be increased if there are thousands of threads configured.</p>
ZombieSessionTimeout	<p>Specifies the number of seconds that a deleted session (a session for which SBR Carrier has received an Accounting-Stop RADIUS message) remains in the Current Sessions Table.</p> <p>Default value is 0 seconds (no grace period).</p> <p>NOTE: This parameter is applicable to standalone servers and servers running in a Session State Register cluster.</p>

[CurrentSessions] Section

The [CurrentSessions] section ([Table 18 on page 67](#)) of **radius.ini** controls the current sessions table.

```
[CurrentSessions]
;CaseSensitiveUsernameCompare = 1
```

Table 18: radius.ini [CurrentSessions] Syntax

Parameter	Function
Enable	<p>If set to 0, the current sessions processing is disabled.</p> <p>This is applicable to standalone servers.</p>

Table 18: radius.ini [CurrentSessions] Syntax (*continued*)

Parameter	Function
CaseSensitiveUsernameCompare	<ul style="list-style-type: none"> • If set to 1, when the server searches its Current Sessions Table for sessions that have the same username, it uses case-sensitive lookups. • If set to 0, the server ignores case. <p>Default value is 0.</p> <p>NOTE: This parameter is applicable only to standalone servers.</p>

For a standalone server, the CST is in local memory, and is configured with the **dbclusterlocal.gen** file when you run the configuration script.

In addition, the setting of the **PersistSessions** parameter in the **radius.ini** file determines whether sessions are restored or not restored when SBR Carrier is restarted.

You cannot configure field names in the local CST (**dbclusterlocal.gen**). However, there are three predefined fields and seven generic fields you can configure using the **sessionTable.ini** file. See *Juniper Networks Steel-Belted Radius Carrier Installation Guide* for information about configuring **sessionTable.ini** file.

[DynAuthProxy]

The [DynAuthProxy] section in the **radius.ini** file controls some global Dynamic Authorization proxy features.

Table 19: [DynAuthProxy] Section

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, this parameter enables the Dynamic Authorization Proxy functionality. • If set to 0, the Dynamic Authorization Proxy functionality is disabled. <p>The default setting is 0.</p>
RequestTimeoutMills	<p>This setting can be used to set the retransmission time (in milliseconds) when forwarding Proxy CoA/DM requests to a NAS client.</p> <p>The default value is 3000, or 3 seconds.</p>
NumAttempts	<p>This setting controls the number of retries before discarding the proxy CoA/DM forwarding requests.</p> <p>The default value is 3.</p>

Table 19: [DynAuthProxy] Section (*continued*)

Parameter	Function
CheckReversePath	<p>This setting controls whether Reverse Path Forwarding checking is done on received Proxy CoA/DM requests.</p> <ul style="list-style-type: none"> • If set to yes, the checking for Reverse Path Forwarding on received proxy CoA/DM requests is enabled. • If set to no, the checking for Reverse Path Forwarding on received proxy CoA/DM requests is disabled. <p>The default is yes.</p>
MessageAuthenticator	<p>The MessageAuthenticator setting, if set to “yes,” causes the Message-Authenticator attribute to be added to every Proxy CoA/DM request forwarded by SBRC. If set to “no,” no Message-Authenticator attribute is forwarded in any Proxy CoA/DM request.</p>
ForwardMethod	<p>This setting determines the method used in finding a NAS target when a CoA/DM proxy request is received.</p> <ul style="list-style-type: none"> • If set to session-table, the setting looks for a matching session in the current sessions table. • If set to direct, the setting tries to match attributes with a configured client. • If set to both, the setting first looks for a matching session in the current sessions table, and then falls back to the direct method if no matching session is found. <p>The default setting is session-table.</p>

[LatencyLog]

The [LatencyLog] section in the **radius.ini** file logs the latency related to authentication or accounting requests received by SBR. A separate file, `latency_<timestamp>.csv`, is created, where timestamp is in the format `yyyymmdd_hhmm`.

Table 20: [LatencyLog] Section

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, this parameter enables the latency log and creates the <code>latency_<timestamp>.csv</code> file. • If set to 0, the latency log functionality is disabled. <p>The default setting is 0.</p>

Table 20: [LatencyLog] Section (*continued*)

Parameter	Function
RollOver	<p>Specifies how often the current latency log file is closed and a new file is opened (a rollover), up to one rollover per minute. Nonzero values indicate the number of minutes until the next rollover.</p> <p>If set to 0, the latency log file rolls over once every 24 hours, at midnight local time.</p> <p>The default value is 0.</p>
RollOverOnStartup	<ul style="list-style-type: none"> • If set to 1, each time SBR is started, it closes the current latency log file and opens a new one. A sequence number <i>_nnnn</i> is appended to the log file name, just as when the maximum size is reached. • If set to 0, each time SBR is started, it appends entries to the previously open latency log file. <p>The default value is 0.</p>

NOTE: When latency log is enabled, the “Enable,” “RollOver,” and “RollOverOnStartup” parameters are read whenever the server receives a SIGHUP (1) signal.

The latency log file contains the following parameters:

Table 21: Latencylog Parameters

Parameter	Description
Date and Time	Logs the date and time of the request.
Thread-Id	Logs the thread-id of the request that is handled.
Type	<p>Logs the type of the request. The type could be one of the following values:</p> <ul style="list-style-type: none"> • auth • acct-start • acct-stop • acct-interim • acct-on • acct-off
Id	In case of authentication and accounting requests, this parameter logs the value of Transaction-id.

Table 21: Latencylog Parameters (continued)

Parameter	Description
Module	<p>Logs the module name where the authentication request is processed. If the latency logger is called from the plug-in module, then this parameter logs the name of the plug-in module. For example, SQL_ORACLE, SBR, or realm name.</p> <p>In case of accounting requests, this parameter logs as None.</p>
Status	<p>Logs the status of the request.</p> <p>In case of authentication requests, this parameter may be "Accept," "Reject," "Challenge," "Ack," or "Discard."</p> <p>In case of accounting requests, this parameter sets the value as "Ack."</p>
Latency	Logs the execution time of the request within the module in milliseconds.
NAS-IP-Address	Logs the source address of the request, which may be IPv4 or IPv6 address.
UDP-Port	Logs the port on which the request has been received.
UserName	Logs the username specified in the request.
Target-Address	<p>Logs the address of the external or local database where the authentication request is validated.</p> <p>In case of accounting requests, this parameter logs the address of the external database.</p>

The following is an example of a sample latency_<time-stamp>.csv file:

```
$cat latency_20111011.csv
"Date","Time","Thread-Id","Type","Id","Module","Status","Latency","NAS-IP-Address","UDP-Port","UserName",
"Target-Address"
"2011-10-13","03:33:50","73","auth","1","SQL_ORACLE","Accept","251","10.206.144.153","3044","suba","182.19.43.5"
"2011-10-13","03:33:50","73","auth","1","SBR","Accept","262","10.206.144.153","3044","suba","10.206.144.153"
"2011-10-13","03:45:50","73","auth","2","LDAP","Accept","242","10.206.144.153","3058","suba","0","184.2.4.64"
"2011-10-13","03:45:50","73","auth","2","PAP","Accept","252","10.206.144.153","3058","suba","10.206.144.153"
"2011-10-13","03:45:57","73","acct-start","2","SBR","Ack","6","10.206.144.153","3060","suba","10.206.144.153"
"2011-10-13","03:46:00","73","auth","4","Bigoo.com","Reject","3","10.206.144.153","3060","suba","192.168.106.153"
"2011-10-13","03:46:00","73","auth","4","SBR","Reject","3","10.206.144.153","3060","suba","10.206.144.153"
"2011-10-13","03:45:50","73","auth","2","SQL_ORACLE","Accept","242","10.206.144.153","3058","suba","182.19.43.5"
"2011-10-13","03:45:50","73","auth","2","SBR","Accept","252","10.206.144.153","3058","suba","10.206.144.153"
"2011-10-13","03:45:57","73","auth","3","SBR","Reject","6","10.206.144.153","3059","suba","10.206.144.153"
"2011-10-13","03:46:00","73","auth","4","SBR","Reject","3","10.206.144.153","3060","suba","10.206.144.153"
```

```
"2011-10-13","04:06:55","75","acct-start","987654321","SR","Ack","5","10.206.144.153","4195","suba","10.206.144.153"
"2011-10-13","04:06:58","75","acct-stop","987654321","SR","Ack","4","10.206.144.153","4195","suba@yahoo.com","10.206.144.153"
```

[EmbedInClass] Section

The [EmbedInClass] section ([Table 22 on page 72](#)) of **radius.ini** identifies attributes that are available during authentication processing which must be made available in accounting requests. Attribute embedding allows billing information to be embedded in a Class attribute returned to Steel-Belted Radius Carrier by a network access server. When Steel-Belted Radius Carrier receives an embedded attribute, it decodes the attribute and places it in the Accounting-Request according to the settings specified in the **classmap.ini** file (described on [“classmap.ini File” on page 212](#)).

NOTE: The **ClassAttributeStyle** parameter in the [Configuration] section of **radius.ini** must be set to a value of 2 before you can use attribute embedding.

The syntax for embedding attributes is:

```
[EmbedInClass]
responseAttribute={ Clear | Encrypt }[,Remove]
```

Table 22: radius.ini [EmbedInClass] Syntax

Parameter	Function
<i>responseAttribute</i>	Identifies the response attribute to be embedded in the RADIUS Class attribute.
Clear	Specifies that the retrieved information is included in the Class attribute in cleartext format.
Encrypt	Specifies that the retrieved information is encrypted before it is included in the Class attribute.
Remove	Optional parameter that removes the embedded attribute from the Accept-Response packet.

[HiddenEAPIdentity] Section

The [HiddenEAPIdentity] section ([Table 23 on page 73](#)) of **radius.ini** allows the known inner identity of EAP/TTLS and EAP/SIM protocols to be included in the Access-Accept message returned in response to an authentication request.

The syntax is:

```
[HiddenEAPIdentity]
IncludeInAcceptResponse=0|1
ResponseAttribute = attributeName[, replaceAttribute]
```

Table 23: radius.ini [HiddenEAPIdentity] Syntax

Parameter	Function
IncludeInAcceptResponse	<ul style="list-style-type: none"> • If set to 0, inclusion of the inner identity in Access-Accept responses is disabled. • If set to 1, Steel-Belted Radius Carrier includes the inner identity in the specified attribute of an Access-Accept response. <p>Default value is 0.</p>
<i>attributeName</i>	Identifies the attribute in which to include the inner identity in an Access-Accept message. If this value is omitted, the User-Name attribute is used. The <i>attributeName</i> value can be any string attribute, including a VSA, that is defined in an attribute dictionary.
<i>[, replaceAttribute]</i>	<p>Identifies the Access-Accept attribute that retains the original value of the attribute specified in the <i>attributeName</i> argument.</p> <p>If a replacement value is not specified, the value of the original attribute is lost.</p>

[IPPoolSuffixes] Section

The [IPPoolSuffixes] section of **radius.ini** lets you define suffixes that can be used to split the IP address pools reserved for a network access server into smaller subcategories.

NOTE: This section is applicable only to standalone servers.

The syntax is:

```
[IPPoolSuffixes]
Suffix1
Suffix2
...
```

For example, to create three categories that append **-Bronze**, **-Silver**, and **-Gold** to IP Address Pool names, this section is defined:

```
[IPPoolSuffixes]
-Bronze
-Silver
-Gold
```

[IPv6] Section

[IPv6]

```
Enable = 0
DynamicNameResolution = 2
IPv6LinkLocalUnicastScopeId = 0
IPv6SiteLocalUnicastScopeId = 0
```

The [IPv6] section ([Table 24 on page 75](#)) of **radius.ini** controls IPv6 network transport features.

Table 24: radius.ini [IPv6] Syntax

Parameter	Function
Enable	<p>Determines whether IPv6 networking is enabled in Steel-Belted Radius Carrier.</p> <ul style="list-style-type: none"> • If set to 0, IPv6 networking is disabled, and other values in the IPv6 section of radius.ini are ignored. • If set to 1, IPv6 networking is enabled. <p>Default value is 1.</p> <p>NOTE: IPv4 networking is always enabled in Steel-Belted Radius Carrier.</p>
DynamicNameResolution	<p>Determines whether the Steel-Belted Radius Carrier server tries to use IPv6 name services (DNSv6) to resolve hostnames.</p> <ul style="list-style-type: none"> • 0—Do not use IPv6 name services. IPv4 name services are not affected by this setting. • 1—Use only IPv6 name services. IPv4 name services are disabled by this setting. • 2—Use IPv6 name services first; use IPv4 name services in case of failure. <p>Default value is 2.</p>
IPv6LinkLocalUnicastScopeld	<p>Specifies an interface name (such as hme0) or index (4) for Solaris hosts.</p> <p>If set to 0, Steel-Belted Radius Carrier does not use link local addresses.</p> <p>Default value is 0.</p> <p>NOTE: The use of IPv6LinkLocalUnicastScopeld parameter has been deprecated.</p>
IPv6SiteLocalUnicastScopeld	<p>Specifies an interface name (such as hme0) or index (4).</p> <p>If set to 0, Steel-Belted Radius Carrier selects the site local scope ID automatically.</p> <p>Default value is 0.</p>

Table 24: radius.ini [IPv6] Syntax (*continued*)

Parameter	Function
UsePools	<p>This setting indicates whether the value of the returned Framed-IPv6-Prefix attribute is calculated using the IPv4 address pools.</p> <p>If set to IPv4, the IPv4 pools are used as the basis for creating the value of the returned Framed-IPv6-Prefix attribute.</p> <p>If set to No, managed IPv6 address pools are not supported.</p> <p>The default value is No.</p>
Pools-IPv6-Prefix-Offset	<p>When UsePools=IPv4, this setting indicates the offset in the Framed-IPv6-Prefix to embed the dynamically assigned IPv4 address. The offset is specified in bits and ranges from 0 through 96. The offset must be a multiple of 8.</p> <p>The default is the last 32 bits of Framed-IPv6-Prefix.</p> <p>For more information about the usage of Framed-IPv6-Prefix, see “Using Managed IPv6 Address Pools” on page 651.</p>

[JavaScript] Section

The [JavaScript] section [Table 25 on page 77](#) of **radius.ini** contains the configuration parameters for the JavaScript engine.

The syntax is:

```
[JavaScript]
;JSEngineRuntimeMemory=8
```

Table 25: radius.ini [JavaScript] Syntax

Parameter	Function
JSEngineRuntimeMemory	<p>Sets the size of the runtime memory arena from which new instances of JavaScript engines are allocated for the core SBR Carrier.</p> <p>NOTE: LDAP and core SBR Carrier use independent instances of JavaScript engines.</p> <p>Default value is 8 MB.</p> <p>NOTE: Increasing the value of JSEngineRuntimeMemory will decrease the frequency of garbage collection but negatively affect performance.</p>

[LDAP] Section

The [LDAP] section ([Table 26 on page 77](#)) of **radius.ini** sets the TCP port number that you want to use for communication between Steel-Belted Radius Carrier and LDAP clients.

The syntax is:

```
[LDAP]
Enable = 1
TCPPort = portNumber
```

Table 26: radius.ini [LDAP] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, the LDAP Configuration Interface is disabled. • If set to 1, the LDAP Configuration Interface is enabled. <p>Default value is 0.</p> <p>NOTE: This parameter is set from your input to the Steel-Belted Radius Carrier configuration script.</p> <p>NOTE: Enabling LCI without changing the access password might leave your Steel-Belted Radius Carrier database vulnerable to access by any LDAP client. For information about using the LDAP configuration interface, see the <i>SBR Carrier Administration and Configuration Guide</i> before you enable this feature.</p>

Table 26: radius.ini [LDAP] Syntax (continued)

Parameter	Function
TCPPort	<p>Specifies the TCP port number that you want to use for communication between Steel-Belted Radius Carrier and LDAP clients.</p> <p>Default value is 667.</p> <p>NOTE: This parameter is set from your input to the Steel-Belted Radius Carrier configuration script, only if you answer "Yes" to the question: "Do you want to enable LCI? [n]:".</p>

[LDAPAddresses] Section

The [LDAPAddresses] section of **radius.ini** lets you specify the interfaces on which Steel-Belted Radius Carrier listens for LDAP Configuration Interface (LCI) requests. If you want to provide these settings, you must add a section called [LDAPAddresses] to the **radius.ini** file. This section contains a list of IP addresses, one per line:

```
[LDAPAddresses]
199.198.197.196
196.197.198.199
```

If the [LDAPAddresses] section is omitted or empty, Steel-Belted Radius Carrier listens for LCI requests on all bound IP interfaces.

NOTE: This parameter is set from your input to the Steel-Belted Radius Carrier configuration script, only if you answer "Yes" to the question: "Do you want to enable LCI? [n]:".

[Logging] Section

The [Logging] section ([Table 28 on page 81](#)) of the **radius.ini** file specifies logging functions for Steel-Belted Radius Carrier.

Log File Naming Conventions and Log Rollover

Steel-Belted Radius Carrier writes to the current server log file until that log file is closed. After closing the file, Steel-Belted Radius Carrier opens a new one and begins writing to it. You can configure how often this rollover of the server log file occurs by setting the Rollover parameter.

The naming conventions for server log files permit more than one file to be generated during a day. [Table 27 on page 79](#) lists the file naming conventions used for different rollover periods. In [Table 27 on page 79](#), *y*= four digit year, *M*= two digit month, *d*= two digit day, *h*= hours digits, and *m*= minutes digits. When more than one file is generated during a day, the sequence number *_nnnnn* starts at *_00000* each day.

Table 27: Server Log File Naming

File Generation Method	File Naming Convention
Default (24 hours)	yyyyMMdd.log
Non-24-hour rollover	yyyyMMdd_hhmm.log
Rollover based on size only	yyyyMMdd_nnnnn.log
Rollover based on both time and size	yyyyMMdd_hhmm_nnnnn.log

For example, if rollover is based on size and multiple rollovers occur on November 21, 2008, they are denoted as:

Nov 21 08:15 20081121_00001.log

Nov 21 08:19 20081121_00002.log

The date matches the system date, and is denoted in a four digit year, two digit month, two digit day, underscore "_", five digit counter which increments per rollover in a given day.

NOTE: If rollover is based only on the size of the log file, the file name format is *yyyyMMdd_nnnn.log*, where *n* is an integer that is incremented each time the file rolls over. Rollover occurs as soon as the current log line causes the file to be longer than the rollover limit (**LogfileMaxMBytes**).

If rollover is on the basis of size (**LogfileMaxMBytes** is **> 0**) and also time (**Rollover** **> 0**), then the log file name format is *yyyyMMdd_HHmm_nnnnn.log*.

The time *HHmm* is the time at which the log was supposed to roll over, even if there no message was logged at that exact time. For example, if you configure the log to roll over every 3 hours, then your log files are called *yyyyMMdd_0300_nnnnn.log*, *yyyyMMdd_0600_nnnnn.log*, and so on. Even if you configure the server at 1:43, and the first message is logged at 4:33, Steel-Belted Radius Carrier bases the rollover starting from midnight every day, so that the times are consistent each day.

Thread Identifiers

The Log-Thread-ID parameter helps debug problems with Steel-Belted Radius Carrier operations by incorporating thread identifiers in log messages for all log levels. Thread identifiers help you parse the diagnostic log when messages about different RADIUS requests are interleaved.

The syntax for including thread identifiers in log messages is:

```
[Logging]
Log-Thread-ID = yes
```

When multiple requests are processed simultaneously, log entries for different requests might appear consecutively in the log file. Configuring the **radius.ini** file to include a thread identification number with log entries correlates the log entries produced while processing each RADIUS request.

The thread identifier appears in parentheses immediately after the date and time. In this example, the Log-Thread-ID of 98 is assigned to one request and 73 is assigned to another.

```
08/24/2008 15:16:27 ../radauthd.c radAuthHandleRequest() 2720 (98) Entering
08/24/2008 15:16:27 (98) Looking up shared secret
08/24/2008 15:16:27 (98) Looking for RAS client 172.25.97.54 in DB
08/24/2008 15:16:27 (98) Matched 172.25.97.54 to RAS client <ANY>
08/24/2008 15:16:27 (98) Parsing request
08/24/2008 15:16:27 (98) Initializing cache entry
08/24/2008 15:16:27 (98) Doing inventory check on request
08/24/2008 15:16:27 (98) Getting info on requesting client
08/24/2008 15:16:27 (98) User-Name : String Value = 1212864080212345

08/24/2008 15:16:27 (73) Authentication Request
08/24/2008 15:16:27 (73) Received from: ip=172.25.97.54 port=4334
08/24/2008 15:16:27 (73)
08/24/2008 15:16:27 (73) Raw Packet :
```

Session Identifiers

The session identifier (**LogSessionID**) further helps with debugging by enabling you to search for log entries associated with a particular user's session. By setting the **LogSessionID** parameter to yes, a session identifier is included in log entries. The session identifier is used throughout authentication and accounting.

The syntax for including session identifiers in log messages is:

```
[Logging]
LogSessionID = yes
```


NOTE: The **ClassAttributeStyle** parameter must be set to a value of 2 before you can use session identifiers.

Use simple search commands or scripts to find a particular user's logged activity. First, find a log entry with data matching that user's identity and note the session identifier. A second search with that identifier yields all messages relating to that user's history in the log file.

The session identifier appears immediately after the thread identifier and is denoted by TxId. The format is:

TIMESTAMP (Thread Id) TxId 0x0000000000000000:00000000: LOG-MESSAGE

Example

11/05/2008 09:51:52 (0056) TxId 0x485c5f8a4911b1fa00000002: Unable to find user test with matching password

NOTE: It is possible that some messages will not include a valid session identifier. Messages logged before the session identifier is learned (before the packet is processed) will have a session identifier of all zeroes. Once the packet is processed, the session identifier is correct.

Enhanced Proxy Logging

Steel-Belted Radius Carrier includes enhanced logging capabilities for troubleshooting proxy target issues. These include presenting transactions between Steel-Belted Radius Carrier and proxy targets in human readable format when the **TraceLevel** parameter is set to 2. In addition, proxy error messages now include the target or realm name. This helps in troubleshooting proxy target issues when multiple proxy targets exist.

Table 28: radius.ini [Logging] Syntax

Parameter	Function
EnhancedDiagnosticLogging	<ul style="list-style-type: none">• If set to no, standard diagnostic logging messages are written to the server log file.• If set to yes, messages relating to proxy retries, proxy timeouts, and LDAP timeouts, as well as standard diagnostic logging messages, are written to the server log file (yyyyymmdd.log). <p>Default value is no.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
EnhancedEAPLogging	<ul style="list-style-type: none"> • If set to no, standard EAP logging messages are written to the server log file in hexadecimal format. • If set to yes, detailed EAP-Message attribute values of EAP-SIM, EAP-AKA, EAP-TLS, and EAP-TTLS authentication protocols along with protocol alerts and error codes are written to the server log file. <p>NOTE: EnhancedEAPLogging=yes is valid only if the TraceLevel parameter in the radius.ini file is set to 2.</p> <p>Default value is yes.</p> <p>NOTE:</p> <ul style="list-style-type: none"> • SBR does not log properly for the Grouped AVP's like Vendor-Specific-Application-ID in Diameter Message. • Enhanced EAP Logging support is provided only for TLS version 1.2. Logging for TLS version 1.1 still remains unsupported.
LogAccept	<ul style="list-style-type: none"> • If set to 1, specifies that messages associated with Accepts that meet the current LogLevel are recorded in the server log file. • If set to 0, messages associated with Accepts are ignored. <p>Default value is 1.</p> <p>The LogAccept setting is re-read whenever the server receives a SIGHUP (1) signal.</p>
LogDir	<p>Sets the destination directory on the local host where server log files are stored.</p> <p>Default value is the Steel-Belted Radius Carrier directory.</p> <p>NOTE: If you specify an alternate destination directory other than the default, ensure that the directory exists before starting the SBR. Otherwise, SBR may fail to function correctly.</p> <p>NOTE: You cannot write server log files to a linked drive.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
LogFileMaxMBytes	<ul style="list-style-type: none"> • If set to 0 (or if setting is absent), the server log file size is ignored and log file names are date-stamped to identify when they were opened (YYYYMMDD.log). • If set to a value in the range 1-2047, the current server log file is closed when it reaches the specified number of megabytes (1024 x 1024 bytes), and a new server log file is opened using the file format (YYYYMMDD_NNNNN.log), where NNNNN is a sequence number. <p>Default value is 0.</p> <p>NOTE: The size of the log file is checked each time a message is logged. The log file might exceed the size specified in LogFileMaxMBytes, because it does not roll over until the next log size check occurs.</p> <p>NOTE: If both LogFileMaxMBytes and MaxSize are present, MaxSize is ignored and the log file size is based on LogFileMaxMBytes (MBytes). If you want to configure the maximum file size in bytes, do not include the LogFileMaxMBytes parameter in this file.</p> <p>NOTE: If LogfileMaxMBytes is set, a new server log file is created whenever the server restarts, even if the log file has not reached the specified number of megabytes. This is an expected behavior.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
LogFilePermissions	<p>Specifies the owner and access permission setting for the system log (yyyymmdd.log) file.</p> <p>Enter a value for the LogFilePermissions setting in <i>owner:group permissions</i> format, where:</p> <ul style="list-style-type: none"> • <i>owner</i> specifies the owner of the file in text or numeric format. • <i>group</i> specifies the group setting for the file in text or numeric format. • <i>permissions</i> specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.</p>
Log-Flush-To-System	<ul style="list-style-type: none"> • If set to no, log flushing is disabled, and log data is queued in a buffer and written to the log at a later time. • If set to yes, log flushing is enabled and log data is written to the log file immediately without being queued. This can impact performance. <p>Default value is no, disabled.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
LogGroup	<p>Specifies the type of server functionality for which you want to log details in the server log file. You can specify the numbers from 0 through 4.</p> <ul style="list-style-type: none"> • 0—All. Includes logs from all the log groups. • 1—Administration. Logs details related to the GUI configuration (both RADIUS and Diameter configurations) and SNMP traps. • 2—SessionControlSuccess and SessionControlFailure. Logs COA/DM messages during session success and failure scenarios. • 3—Diameter Peer State. Logs IP address, port, event, and transition state of the Diameter peer when the Device-Watchdog-Request message is received from the Diameter peer. • 4—Others. Logs the following details: <ul style="list-style-type: none"> • System related information such as system start, system stop, resource failures, and so on. • Error messages during configuring EAP methods and filters by using the Web GUI. • Access-Accept messages including details such as username, policy, authentication method, realm, protocol, Calling-Station-Id, Called-Station-Id, NAD, and so on, during RADIUS to Diameter translation scenarios. • Access-Reject messages with the reason for the reject during RADIUS to Diameter translation scenarios. <p>You can specify more than one number in this parameter; the numbers must be comma separated.</p> <p>Default value is 0.</p> <p>The LogGroup setting is re-read whenever the server receives a SIGHUP (1) signal.</p> <p>NOTE: Configuration logs cannot be disabled. For an example of using log levels with log groups, see <i>SBR Carrier Administration and Configuration Guide</i>.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
LogHighResolutionTime	<ul style="list-style-type: none"> • If set to no, the timestamp for entries in the Steel-Belted Radius Carrier log file (yyyyymmdd.log) are recorded as MM/DD/YYYY/hh:mm:ss(month/date/year/hour:minutes:seconds). • If set to yes, the timestamp for entries in the Steel-Belted Radius Carrier log file (yyyyymmdd.log) are recorded as MM/DD/YYYY/hh:mm:ss.xxx, where xxx represents the number of elapsed milliseconds since the ss value changed. <p>Default value is no.</p> <p>NOTE: If the value for LogLevel is set as 2, then the entries to the server log file will contain both the thread ID (Log-Thread-ID) and timestamps with millisecond (LogHighResolutionTime) details, unless they are explicitly disabled.</p>
LogLevel	<p>Sets the rate at which Steel-Belted Radius Carrier writes entries to the server log file (yyyyymmdd.log):</p> <ul style="list-style-type: none"> • 0—Default, errors. • 1—Log errors and warnings • 2—Debugging messages including info, warnings, and errors <p>Default value is 0.</p> <p>NOTE: If the value for LogLevel is set as 2, then the entries to the server log file will contain both the thread ID (Log-Thread-ID) and timestamps with millisecond (LogHighResolutionTime) details, unless they are explicitly disabled.</p> <p>The LogLevel setting is re-read whenever the server receives a SIGHUP (1) signal.</p>
LogReject	<ul style="list-style-type: none"> • If set to 0, messages associated with Rejects are ignored. • If set to 1, messages associated with Rejects that meet the current LogLevel are recorded in the server log file. <p>Default value is 1.</p> <p>The LogReject setting is re-read whenever the server receives a SIGHUP (1) signal.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
LogSessionID	<ul style="list-style-type: none"> • If set to yes, session identifiers are included in Steel-Belted Radius Carrier log messages. • If set to no, session identifiers are omitted from Steel-Belted Radius Carrier log messages. <p>Default value is no.</p>
Log-Thread-ID	<ul style="list-style-type: none"> • If set to yes, thread identifiers are included in Steel-Belted Radius Carrier log messages. • If set to no, thread identifiers are omitted from Steel-Belted Radius Carrier log messages. <p>Default value is no.</p> <p>NOTE: If the value for LogLevel is set as 2, then the entries to the server log file will contain both the thread ID (Log-Thread-ID) and timestamps with millisecond (LogHighResolutionTime) details, unless they are explicitly disabled.</p>
LogUsesUtc	<ul style="list-style-type: none"> • If set to no (disabled), the time used to timestamp messages in the log file is the local time zone. Use of local time causes timestamps to be automatically adjusted for seasonal adjustments, such as Daylight Saving Time in the United States, if applicable. • If set to yes (enabled), the time used to timestamp messages in the log file is the Coordinated Universal Time (UTC, formerly known as Greenwich Mean Time or GMT) time zone 0. <p>Default value is no (disabled).</p>
MaxSize	<p>The maximum size of a server log file, in bytes.</p> <p>If the server log file reaches or exceeds this size when it is checked, the log file is closed and a new file is started. A value of 0 (the default) means unlimited size.</p> <p>NOTE: If both LogFileMaxMBytes and MaxSize are present, MaxSize is ignored and the log file size is based on LogFileMaxMBytes (MBytes). If you want to configure the maximum file size in bytes, do not include the LogFileMaxMBytes parameter in this file.</p>

Table 28: radius.ini [Logging] Syntax (*continued*)

Parameter	Function
ReplaceUnprintables	<p>Specifies a printable character which is used instead of non-printing characters when SBR Carrier writes messages to the accounting log file. You can define a printable character of ASCII decimal code 32 through 126 (or ASCII hex code 20 through 7E).</p> <p>You can disable the replacement by setting this parameter to no. Setting this parameter to no truncates a line in the accounting log when a non-printing character is encountered.</p> <p>Default value is no.</p> <p>NOTE: Characters of ASCII decimal code 0 through 31 (ASCII hex code 0 through 1F) and 127 through 255 (ASCII hex code 7F through FF) are considered as non-printing characters.</p>
Rollover	<p>Specifies how often the current server log file is closed and a new file opened (a rollover), up to one rollover per minute. Nonzero values indicate the number of minutes until the next rollover.</p> <p>If set to 0, the server log file rolls over once every 24 hours, at midnight local time.</p> <p>Default value is 0.</p> <p>NOTE: Rollover based on time or size, or both only is checked once a minute. Therefore, neither sizes nor times is exact.</p>
TraceLevel	<p>Specifies the RADIUS packet tracing level:</p> <ul style="list-style-type: none"> • 0—Default, no packet tracing • 1—Trace standard packet content • 2—Trace standard and raw packet content <p>Default value is 0.</p> <p>NOTE: Packet traces are written to the server log file and can be a useful tool for troubleshooting interoperability problems.</p>

[MsChapNameStripping] Section

The [MsChapNameStripping] section (Table 29 on page 89) of **radius.ini** specifies whether you want Steel-Belted Radius Carrier to try to strip domain information from usernames when it tries to match its user entry to the username/password hash forwarded by the end user. This feature is useful in situations where the username in the Steel-Belted Radius Carrier database includes characters the end-user host considers domain information, which it deletes before computing its hash of the user’s credentials.

If this feature is enabled:

1. Steel-Belted Radius Carrier scans the username in its database looking for delimiter characters that might indicate a domain is prefixed to the username. If a prefix delimiter character is found, the server strips that character (and all characters to the left of the delimiter), generates its own hash of the user’s credentials, and compares the result to the hashed credentials forwarded by the end user to determine if a match is found.
2. If a prefix delimiter is not found (or if the hashed credentials do not match after the prefix is stripped), Steel-Belted Radius Carrier scans the username looking for delimiter characters that might indicate a domain is suffixed to the username. If a suffix delimiter character is found, the server strips that character (and all characters to the right of the delimiter), generates its own hash of the user’s credentials, and compares the result to the hashed credentials forwarded by the end user to determine if a match is found.
3. If neither a prefix delimiter nor a suffix delimiter is found (or if a delimiter was found but the hashed credentials did not match), the server uses the entire username string to generate the hashed credentials and compares the result to the hashed credentials forwarded by the end user to determine if a match is found.

The syntax for the [MsChapNameStripping] section is:

```
[MsChapNameStripping]
Enable=1
Prefix=\\
Suffix=/@
```

Table 29: radius.ini [MsChapNameStripping] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">• If set to 0 (or omitted), MS-CHAP v2 name stripping is disabled.• If set to 1, MS-CHAP v2 name stripping is enabled. Default value is 0.

Table 29: radius.ini [MsChapNameStripping] Syntax (*continued*)

Parameter	Function
Prefix	<p>A list of as many as five ASCII characters to strip from the prefix. If a space character appears in the list, the entire list must be surrounded by quotation marks.</p> <p>Enter a double backslash (\\) to indicate you want to strip the backslash character. A double backslash counts as one character in the list.</p> <p>Default value is \.</p>
Suffix	<p>A list of as many as five ASCII characters to strip from the suffix. If a space character appears in the list, the entire list must be surrounded by quotation marks.</p> <p>Enter a double backslash (\\) to indicate you want to strip the backslash character. A double backslash counts as one character in the list.</p> <p>Default value is /@.</p>

[PurgeThreadLogging] Section

You can use the [PurgeThreadLogging] section ([Table 30 on page 91](#)) of the **radius.ini** file to specify the attributes to be included in the purged stale session log messages.

The syntax is:

```
[PurgeThreadLogging]
PurgeThreadLogging_attributes=
```

Table 30: radius.ini [PurgeThreadLogging] Syntax

Parameter	Function
PurgeThreadLogging_attributes	<p>Specifies the attributes to be included in the purged stale session log messages printed in the SBR log even if the LogLevel parameter in the radius.ini file is set to 0. You can specify the following attributes in this parameter.</p> <ul style="list-style-type: none"> • Unique-Session-ID—SBR Carrier includes the unique identifier of the purged stale session in the purged stale session log messages. • User-Name—SBR Carrier includes the RADIUS username of the purged stale session in the purged stale session log messages. • NasName—SBR Carrier includes the NAD name of the purged stale session in the purged stale session log messages. • Acct-Session-ID—SBR Carrier includes the accounting session identifier of the purged stale session in the purged stale session log messages. • Calling-Station-ID—SBR Carrier includes the station identifier of the purged stale session in the purged stale session log messages. <p>The attributes specified in this parameter must be separated by a space. For example, PurgeThreadLogging_attributes= User-Name Calling-Station-ID. You can also set the PurgeThreadLogging_attributes parameter to all to include all the preceding attributes in the purged stale session log messages.</p> <p>By default, no attribute is configured in this parameter. In this case, the User-Name and NasName attributes are included in the purged stale session log messages.</p> <p>NOTE: If you have left this parameter empty or misspelled an attribute, only the User-Name and NasName attributes are included in the purged stale session log messages.</p>

[Ports] Section

The [Ports] section ([Table 31 on page 92](#)) of **radius.ini** provides a method for setting the UDP ports used by Steel-Belted Radius Carrier.

- If one or more **UDPAuthPort** settings are specified in the [Ports] section of **radius.ini**, the port numbers in this section are the only ones on which the server listens for authentication requests. Similarly, if one or more **UDPAcctPort** settings are specified, they are the only ones on which the server listens for accounting requests.

You can specify as many as 4096 ports on a Solaris server. If this limit is exceeded, the RADIUS authentication subcomponent fails to initialize.

- If no **UDPAuthPort** or **UDPAcctPort** settings are present in the [Ports] section, the server attempts to read the port numbers associated with **radius** service (authentication) and **radacct** (accounting) in

/etc/services. If successful, the server listens on these port numbers. No more than one port can be specified for the **radius** service or for the **radacct** service.

- If no **UDPAuthPort** settings are present in the [Ports] section and no **radius** service or **radacct** is listed in the **/etc/services** file, the server listens for authentication requests on UDP ports 1645 and 1812 for authentication and UDP ports 1646 and 1813 for accounting.

NOTE: Any failure to bind to one of the selected UDP ports causes the affected subcomponent (authentication or accounting) to fail to initialize.

If you want the server to function as a proxy forwarding server, you can specify a block of UDP port numbers from which the proxy RADIUS ports are allocated. Proxy RADIUS allocates port numbers in sets of eight. Port numbers in an allocated block do not have to be contiguous: if a UDP port number that falls in the proxy RADIUS range is in use, proxy RADIUS skips over it.

Table 31: radius.ini [Ports] Syntax

Parameter	Function
ProxyPortCount	The ProxyPortCount parameter is used to configure SBR Carrier for load when proxies are being used. The setting of ProxyPortCount instructs SBR Carrier how many ports to use from within the number of possible ports defined within UDPProxyPortBlockLength starting with the port value set at UDPProxyPortBlockStart .
DynAuthProxyPortCount	The DynAuthProxyPortCount parameter determines the number of ports that is actually allocated for CoA/DM functionality. The setting of DynAuthProxyPortCount instructs SBR Carrier on how many ports to use from within the number of possible ports defined within UDPDynAuthProxyPortBlockLength starting with the port value set at UDPDynAuthProxyPortBlockStart .
SecureTcpAdminAddress	Specifies the IP address of the administrative interface used for communication between Web GUI and the Steel-Belted Radius Carrier server. If not specified, any network interface on the Steel-Belted Radius Carrier server accepts a connection from Web GUI.

Table 31: radius.ini [Ports] Syntax (continued)

Parameter	Function
SecureTcpAdminPort	<p>Specifies the TCP port used for communication between Web GUI and the Steel-Belted Radius Carrier server.</p> <p>Default value is 1813.</p> <p>NOTE: Consult Juniper Networks Technical Support before changing the port number. Using a non-default port may cause communication problems between Web GUI and the Steel-Belted Radius Carrier server.</p>
TCPControlAddress	<p>Specifies the IP address of the administrative interface on the Steel-Belted Radius Carrier server used for SNMP and CCM/replication communication.</p> <p>If not specified, any network interface on the Steel-Belted Radius Carrier server can be used for SNMP and CCM traffic.</p>
TCPControlPort	<p>Specifies the TCP port used for SNMP and CCM/replication communication.</p> <p>Default value is 1812.</p> <p>NOTE: Consult Juniper Networks Technical Support before changing the port number. Using a non-default port may cause communication problems between Web GUI and the Steel-Belted Radius Carrier server.</p>
UDPAcctPort	<p>Specifies the UDP port(s) used for accounting. If you use more than one port, specify each port number on a separate line.</p> <p>Default values are 1646 and 1813.</p> <p>NOTE: Consult Juniper Networks Technical Support before changing the port number. Using a non-default port may cause communication problems between Web GUI and the Steel-Belted Radius Carrier server.</p>

Table 31: radius.ini [Ports] Syntax (continued)

Parameter	Function
UDPAuthPort	<p>Specifies the UDP port(s) used for authentication. If you use more than one port, specify each port number on a separate line.</p> <p>Default values are 1645 and 1812.</p> <p>NOTE: Consult Juniper Networks Technical Support before changing the port number. Using a non-default port may cause communication problems between Web GUI and the Steel-Belted Radius Carrier server.</p>
UDPDynAuthPort	<p>This parameter indicates the ports that SBRC listens on for proxy CoA/DM messages.</p> <p>Default value is 3799.</p>
UDPProxyPortBlockLength	<p>Specifies the number of addresses in the port number range used for proxy RADIUS communication.</p> <p>Default value is 64.</p>
UDPProxyPortBlockStart	<p>Specifies the starting port number in the port number range used for proxy RADIUS communication.</p> <p>Default value is 28000.</p> <p>NOTE: If you change the default value, select a number range that does not overlap with well-known UDP ports and proprietary UDP ports on your network.</p> <p>NOTE: You might need to configure network firewalls to allow ports in the specified number range to pass.</p>
UDPDynAuthProxyPortBlockStart	<p>Specifies the starting port-number in the port number range used for proxy CoA/DM requests.</p> <p>Default value is 30,000.</p> <p>NOTE: If you change the default value, select a number range that does not overlap with well-known UDP ports and proprietary UDP ports on your network.</p> <p>NOTE: You might need to configure network firewalls to allow ports in the specified number range to pass.</p>

Table 31: radius.ini [Ports] Syntax (continued)

Parameter	Function
UDPPProxyPortBlockLength	Specifies the number of addresses in the port-number range used for proxy CoA/DM requests. Default value is 64.

For example:

```
[Ports]
SecureTcpAdminPort = 1813
SecureTcpAdminAddress = 192.168.12.15
TcpControlPort = 1812
TCPControlAddress = 192.168.15.55
UDPAuthPort = 1645
UDPAuthPort = 1812
UDPAcctPort = 1646
UDPAcctPort = 1813
UDPPProxyPortBlockStart = 28000
UDPPProxyPortBlockLength = 64
```

The UDP port assignments entered in the [Ports] section of the **radius.ini** file override the UDP port assignments specified in the **/etc/services** file. For more information, see [“services File” on page 116](#).

[Self] Section

The [Self] section of **radius.ini** lists all the realm names that the *Steel-Belted Radius Carrier* server handles locally. The syntax is:

```
[Self]
RealmName
RealmName
```

You can use the [Self] section to map a realm name to the *Steel-Belted Radius Carrier* server. If you acquire a batch of new user accounts, users do not have to change how they enter usernames. They can enter the name *User<Delimiter>RealmName* or *RealmName<Delimiter>User* as usual.

When a username comes into Steel-Belted Radius Carrier, if the [Self] section lists *RealmName*, Steel-Belted Radius Carrier recognizes it as the target, and handles the request locally instead of directing the request elsewhere.

[StaticAcctProxy] Section

The [StaticAcctProxy] section of **radius.ini** controls the delivery of accounting messages to additional RADIUS accounting-enabled devices on the network, even when the initial RADIUS transaction is not a proxy RADIUS transaction. The syntax is:

```
[StaticAcctProxy]
target = proxy
```

Where *proxy* identifies the name of the RADIUS accounting-enabled device.

[Status] Section

The [Status] section specifies whether authentication, accounting, and proxy thread and flood information is added to the server log.

Table 32: radius.ini [Status] Syntax

Parameter	Function
Status-Period	<p>Specifies the frequency (in seconds) that the status report is written to the log.</p> <p>Default value is 60 seconds.</p>
Auth-Thread-Flood-Info	<ul style="list-style-type: none"> • If set to yes, an authentication or authorization thread and flood information are included in the status report. • If set to no, an authentication or authorization thread and flood information are not included in the status report. <p>Default value is no.</p>
Acct-Thread-Flood-Info	<ul style="list-style-type: none"> • If set to yes, an accounting thread and flood information are included in the status report. • If set to no, an accounting thread and flood information are not included in the status report. <p>Default value is no.</p>
Proxy-Thread-Flood-Info	<ul style="list-style-type: none"> • If set to yes, a proxy thread and flood information are included in the status report. • If set to no, a proxy thread and flood information are not included in the status report. <p>Default value is no.</p>

Table 32: radius.ini [Status] Syntax (*continued*)

Parameter	Function
DynAuth-Thread-Flood-Info	<ul style="list-style-type: none"> • If set to yes, a dynamic authentication or authorization thread and flood information are included in the status report. • If set to no, a dynamic authentication or authorization thread and flood information are not included in the status report. <p>Default value is no.</p>
Cache-Report	<ul style="list-style-type: none"> • If set to yes, cache information is included in the status report. • If set to no, cache information is not included in the status report. <p>Default value is no.</p>
Cache-Report-Details	<ul style="list-style-type: none"> • If set to yes, detailed cache information is included in the status report. • If set to no, detailed cache information is not included in the status report. <p>Default value is no.</p>
Accounting-Report	<ul style="list-style-type: none"> • If set to yes, accounting statistics are included in the status report. • If set to no, accounting statistics are not included in the status report. <p>Default value is no.</p>
Thread-Count	<ul style="list-style-type: none"> • If set to yes, thread counts are included in the status report. • If set to no, thread counts are not included in the status report. <p>Default value is no.</p>

Following is an example of the log entries if all of the [Status] report parameters are set to **yes**.

```

11/11/2010 11:40:05: ===== Status Report Start (period = 1 sec.) =====
11/11/2010 11:40:05:   start Thread Count Report
11/11/2010 11:40:05:       thread_count: 0 Authentication Threads
11/11/2010 11:40:05:       thread_count: 0 Accounting Threads

```

```

11/11/2010 11:40:05:      thread_count: 0 Proxy Threads
11/11/2010 11:40:05: end Thread Count Report
11/11/2010 11:40:05: Start Threads & Floods
11/11/2010 11:40:05: Auth:
11/11/2010 11:40:05:   Packets, Since Reset.....Arrived.....:      0
11/11/2010 11:40:05:                               Serviced.....:      0
11/11/2010 11:40:05:                               Flooded.....:      0
11/11/2010 11:40:05:                               Dropped.....:      0
11/11/2010 11:40:05:           This Period.....Arrived.....:      0
11/11/2010 11:40:05:                               Serviced.....:      0
11/11/2010 11:40:05:                               Flooded.....:      0
11/11/2010 11:40:05:                               Dropped.....:      0
11/11/2010 11:40:05:   Flood Queue.....Shape.....:      LIFO
11/11/2010 11:40:05:                               Max Configured....:      25
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05:   Thread Pool.....Max Configured....:      100
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05:           In Flood Queue.....Max Configured....:      50
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05: Acct:
11/11/2010 11:40:05:   Packets, Since Reset.....Arrived.....:      0
11/11/2010 11:40:05:                               Serviced.....:      0
11/11/2010 11:40:05:                               Flooded.....:      0
11/11/2010 11:40:05:                               Dropped.....:      0
11/11/2010 11:40:05:           This Period.....Arrived.....:      0
11/11/2010 11:40:05:                               Serviced.....:      0
11/11/2010 11:40:05:                               Flooded.....:      0
11/11/2010 11:40:05:                               Dropped.....:      0
11/11/2010 11:40:05:   Flood Queue.....Shape.....:      LIFO
11/11/2010 11:40:05:                               Max Configured....:      25
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05:   Thread Pool.....Max Configured....:      200
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05:           In Flood Queue.....Max Configured....:      100
11/11/2010 11:40:05:                               High Since Reset...:      0
11/11/2010 11:40:05:                               High This Period...:      0
11/11/2010 11:40:05: End Threads & Floods
11/11/2010 11:40:05: start Cache Report
11/11/2010 11:40:05:      packetcache - 0 entries in the packet cache, cache for

```

```

id = 0
11/11/2010 11:40:05:      packetcache - 0 entries in the packet cache, cache for
id = 1
.
.
.
11/11/2010 11:40:06: end Cache Report
11/11/2010 11:40:06: start Accounting Statistics Report
11/11/2010 11:40:06:      acct_stats: 0 packets received
11/11/2010 11:40:06:      acct_stats: 0 packets cache discarded
11/11/2010 11:40:06:      acct_stats: 0 packets cache responded
11/11/2010 11:40:06:      acct_stats: 0 starts (or equivalent)
11/11/2010 11:40:06:      acct_stats: 0 stops (or equivalent)
11/11/2010 11:40:06:      acct_stats: 0 ons
11/11/2010 11:40:06:      acct_stats: 0 offs
11/11/2010 11:40:06:      acct_stats: 0 unknown acct-status-type
11/11/2010 11:40:06:      acct_stats: 0 phantoms created
11/11/2010 11:40:06:      acct_stats: 0 records deleted by admin
11/11/2010 11:40:06:      acct_stats: 0 records deleted because of on/off
11/11/2010 11:40:06:      acct_stats: 0 records deleted because of port in use
11/11/2010 11:40:06:      acct_stats: 0 starts that replaced a phantom
11/11/2010 11:40:06:      acct_stats: 0 stops that found a phantom session
11/11/2010 11:40:06:      acct_stats: 0 stops that found a start session
11/11/2010 11:40:06:      acct_stats: 0 stops that found no session
11/11/2010 11:40:06: end Accounting Statistics Report
11/11/2010 11:40:06: ===== Status Report End (period = 1 sec.)

```

[Strip] Section

The [Strip] section ([Table 33 on page 100](#)) specifies how Steel-Belted Radius Carrier manipulates the username by stripping the incoming User-Name attribute value of realm names and other decorations.

The [Strip] section (and accompanying [StripPrefix] and [StripSuffix] sections) look like this:

```

[Strip]
Authentication=Yes
Accounting=No
StripPrefixCharacters=@#%
StripSuffixCharacters="! "

[StripPrefix]
PrefixStringToStrip1
PrefixStringToStrip2

```

```
[StripSuffix]
SuffixStringToStrip1
SuffixStringToStrip2
```

Table 33: radius.ini [Strip] Syntax

Parameter	Function
Authentication	<p>If set to yes, the [StripPrefix] and [StripSuffix] rules are used to strip the username before an authentication request is processed.</p> <p>Default value is no.</p>
Accounting	<p>If set to yes, the [StripPrefix] and [StripSuffix] rules are used to strip the username before an accounting request is processed.</p> <p>Default value is no.</p>
StripPrefixCharacters	<p>A list of ASCII characters to strip from the prefix. If a space character appears in the list, the entire list must be surrounded by quotation marks.</p>
StripSuffixCharacters	<p>A list of ASCII characters to strip from the suffix. If a space character appears in the list, the entire list must be surrounded by quotation marks.</p>

[StripPrefix] Section

The [StripPrefix] section lists prefixes you want removed from the beginning of usernames, including the delimiter. If a space character appears in the list, the entire list must be surrounded by quotation marks.

```
[Strip]
Authentication=yes
Accounting=yes

[StripPrefix]
isp.com\
att.net]
```

In this example, Steel-Belted Radius Carrier strips the prefixes **isp.com** and **att.net]** from usernames in authentication and accounting requests.

[StripSuffix] Section

The [StripSuffix] section lists suffixes you want removed from the end of usernames, including the delimiter.

For example:

```
[Strip]
Authentication=yes
Accounting=yes

[StripSuffix]
@myrealm.com
@yahoo.com
```

In this example, Steel-Belted Radius Carrier strips the suffixes **@myrealm.com** and **@yahoo.com** from usernames in authentication and accounting requests.

[UserNameTransform] Section

The [UserNameTransform] section ([Table 34 on page 102](#)) lets you specify a rule for transforming usernames in RADIUS requests from the form in which they are received to a form in which they can be processed. This can be useful when the form in which users supply their names to the network access server is not compatible with the form in which the RADIUS server applies its rules for proxy forwarding or with the form that the authentication system requires.

The username transformation rule used to convert input strings to output strings is based on an *input format* and an *output format*. The username transformation rule is applied to usernames appearing in RADIUS requests. The username from the RADIUS request is parsed based on the input format.

- If the username does not conform to the input format, the rule does not apply and the username is unchanged.
- If the rule does apply, the parsed elements of the username are formatted based on the output format to construct the transformed username:
 1. The User-Name from the Access-Request (or Acct-Start/Acct-Stop) is compared to the input format rule.
 2. If the User-Name matches the rule, it is modified into the output format, and authentication continues.
 3. If the User-Name does not match the input format, no modification occurs, and authentication continues.

The transformed username replaces the original username in RADIUS processing, just as if the transformed username had been included in the request. The decision to proxy-forward the packet is based on the transformed username, and all authentications are based on the transformed username.

Format strings can be any sequence of characters, and can contain embedded variables enclosed in angle brackets (< >). The backslash (\) is an escape character within text, used to represent literal characters. Within variable names, a backslash is treated as a character, not as an escape; and therefore, variable names may not include right angle brackets (>).

Compose the literal text with characters you do not expect to be found in the variable elements. Use punctuation characters such as a slash (/) or an at-sign (@), rather than letters or numbers.

The username transformation rule can be applied to authentication packets, accounting packets, or both.

Example

```
[UserNameTransform]
In=<input format>
Out=<output format>
Authentication=<yes | no>
Accounting=<yes | no>
```

Table 34: radius.ini [UserNameTransform] Syntax

Parameter	Function
In	A format string identifying the input format for usernames. For example, <user>@<realm>.
Out	A format string identifying the output format for usernames. For example, <user>.
Authentication	Set to Yes to enable the transform for authentication requests. Default value is Yes.
Accounting	Set to Yes to enable the transform for accounting requests. Default value is Yes.
Proxy	Set to Yes to enable the transform for proxied requests. Default value is Yes.

For example, these settings transform **george@acme.com** to **george**:

```
In = <user>@<realm>
Out = <user>
```

These settings transform **abc/martha@bigco.com** to **bigco.com::abc/martha**:

```
In = <prefix>/<user>@<realm>
Out = <realm>::<prefix>/<user>
```

[ValidateAuth] and [ValidateAcct] Sections

The [ValidateAuth] and [ValidateAcct] sections ([Table 35 on page 103](#)) of **radius.ini** specify how Steel-Belted Radius Carrier validates usernames in authentication and accounting requests. These sections enable SBR Carrier to examine the User-Name attribute in the incoming packet to determine whether it employs a valid character set.

```
[ValidateAuth]
User-Name = RegularExpression

[ValidateAcct]
User-Name = RegularExpression
```

Table 35: radius.ini [ValidateAuth] and [ValidateAcct] Syntax

Parameter	Function
[ValidateAuth]	This section applies only to authentication servers.
[ValidateAcct]	This section applies only to accounting servers.
User-Name	Names the regular expression against which the User-Name attribute is validated. If the User-Name entry is absent from the section or the regular expression is blank, no validation occurs.

Table 35: radius.ini [ValidateAuth] and [ValidateAcct] Syntax (continued)

Parameter	Function
RegularExpression	<p>The regular expression lists each valid character or range of characters.</p> <p>A dash (-) indicates a range of alphanumeric characters. For example, A-Z indicates every uppercase alphabetic character.</p> <p>A backslash (\) followed by a non-alphanumeric character indicates that character literally, for example \? indicates the question mark.</p> <p>\ is used as an escape character:</p> <p>\a bell (7)</p> <p>\b backspace (8)</p> <p>\t tab (0x09)</p> <p>\n newline (10)</p> <p>\v vertical tab (11)</p> <p>\f formfeed (12)</p> <p>\r return (13)</p> <p>\xnn hex value, where nn is a two-digit hexadecimal number</p> <p>\nnn decimal value, where nnn is a three-digit decimal number</p>

This example permits a string composed only of uppercase and lowercase characters, digits, periods, and commas:

```
User-Name = A-Za-z0-9.,
```

This example permits uppercase and lowercase characters:

```
User-Name = A-Za-z
```

sbrd.conf File

The **sbrd.conf** file (Table 36 on page 107) is an executable Bourne shell script that is invoked by the **sbrd** process to initialize the execution environment for Steel-Belted Radius Carrier.

NOTE: In previous versions of server software, users were instructed to modify the **sbrd** script if they wanted to change its settings. The **sbrd.conf** file makes direct modification to the **sbrd** script unnecessary. Do not modify the **sbrd** script.

For example:

```
#!/bin/sh
#####
# sbrd.conf
#####
# This is an executable Bourne shell script, invoked by sbrd in order to
# initialize the execution environment for Steel-Belted Radius software.
# Among others, RADIUSDIR and SELF are read-only constants defined in the
# sbrd script. Do not attempt to modify these read-only constants.

# Edit these lines to enable configurations that open many files concurrently.
# Management of high file descriptors is required beyond around 224 open files.
# Do not exceed the 1024 file limitation that exists for 32-bit applications.
# Solaris pfiles and Open Source lsof utilities are able to report open files.
ULIMIT_OPEN_FILES=1024 # typically 256 - 1024 inclusive, "disabled", or ""
RADIUS_HIGH_FDS=1 # management of high file descriptors (0=disable, 1=enable)
#ORACLE_MSB_FILE="$ORACLE_HOME/rdbms/mesg/ocius.msb" # absolute filename, or ""

# Edit this line to configure the file mode creation mask, see umask(1).
# Specify an explicit value in order to override the current environment.
# Specify "" to use the current environment as is without modification.
RADIUSUMASK="" # either a valid umask argument, or ""

# Radius executable, options, and arguments
RADIUS="radius"
RADIUSOPTS=""
RADIUSARGS="sbr.xml"
RADIUS_PRIVATE_DIR="$RADIUSDIR"

# Watchdog executable, options, and arguments
WATCHDOGENABLE=0 # Edit this line to enable watchdog (0=disable, 1=enable)
WATCHDOG="radiusd"
WATCHDOGOPTS="--config $RADIUSDIR/radiusd.conf --pidfile $RADIUSDIR/radius.pid"
WATCHDOGOPTS="$WATCHDOGOPTS --logfile $RADIUSDIR/radiusd.log"
WATCHDOGARGS="$RADIUSDIR/$SELF"

#Configuration parameter to start GWrelay process (0=no, 1=yes)
```

```

GWRELAYENABLE=0

#Configuration parameter to configure shutdown timeout in seconds
#(Min=45, Max=360)
SHUTDOWNTIMEOUT=45

# Edit these lines to interpose arbitrary libraries, e.g. libumem.so allocator.
# We prefer /lib/libumem.so over the default /usr/lib/libmtmalloc.so for
# performant, scalable, multi-threaded memory allocation with optional debug.
# Uncomment UMEM_ variables to enable libumem.so debug, see umem_debug(3MALLOC).
# WARNING: Enabling libumem.so debug has a noticeable impact on performance!
RADIUS_LD_PRELOAD="/lib/libumem.so" # a space separated list of libs, or ""
#UMEM_DEBUG="default" # Uncomment to enable libumem.so debug facilities
#UMEM_LOGGING="transaction" # Uncomment to enable libumem.so in-memory logs

# Edit these lines to configure the management of radius specific core files.
# Specify an explicit value in order to override the current environment.
# Specify "disabled" to use the current environment as is without modification.
# Specify "" to use the current environment as is and adjust if inappropriate.
ULIMIT_CORE_SIZE="" # either a valid ulimit -c argument, "disabled", or ""
ULIMIT_CORE_COUNT=3 # either 0 - 999999999, "unlimited", "disabled", or ""

# WARNING: Following parameters are auto-configured. Manual editing is not
recommended.
WEBSERVER_JAVA_HOME=
CUSTOM_JVM_PATH=

```

NOTE: Do not include spaces in parameter settings in the **sbrd.conf** file.

Correct: **ULIMIT_CORE_COUNT=1024**

Incorrect: **ULIMIT_CORE_COUNT = 1024**

Table 36: sbrd.conf Syntax

Parameter	Function
ULIMIT_CORE_SIZE	<p>Specifies the size of core files generated if SBR Carrier fails.</p> <ul style="list-style-type: none"> • If set to a value, ULIMIT_CORE_SIZE specifies the maximum size for core files in 512-byte blocks (Solaris). • If set to disabled, SBR Carrier uses the current environment without changes. • If set to "" (two double-quotes with no space between), SBR Carrier uses the current environment, making adjustments as needed. <p>Default value is "".</p>
ULIMIT_CORE_COUNT	<p>Specifies the number of core files maintained on the SBR Carrier server. If the maximum number of core files already exists on the server, SBR Carrier discards the oldest core files and generates a new core file if it fails.</p> <ul style="list-style-type: none"> • If set to a number in the range 0–999,999,999, the server maintains the specified number of core files. • If set to unlimited, SBR Carrier does not discard existing core files if it generates a new one. • If set to disabled, SBR Carrier uses the current environment without changes. • If set to "" (two double-quotes with no space between), SBR Carrier uses the current environment, making adjustments as needed. <p>Default value is 3.</p>

Table 36: sbrd.conf Syntax (*continued*)

Parameter	Function
ULIMIT_OPEN_FILES	<p>Specifies the number of open files that the SBR Carrier process can have open at one time.</p> <ul style="list-style-type: none"> • If set to a number in the range 256–1024, the server maintains the specified number of open files. • If set to disabled, SBR Carrier uses the current environment without changes. • If set to "" (two double-quotes with no space between), SBR Carrier uses the current environment, making adjustments as needed. <p>Default value is 1024.</p> <p>NOTE: ULIMIT_OPEN_FILES should never be set less than 256. RADIUS_HIGH_FDS should always be set to 1 unless Juniper Networks Technical Support advises otherwise. Together these parameters ensure that SBR Carrier is always able to open at least 256 regular files. This is especially important for sites that configure multiple Oracle plug-ins.</p>
RADIUSMASK	<p>Specifies the file permissions that are withheld when new log files are created.</p> <ul style="list-style-type: none"> • If set to an umask argument, log files are created with the specified permissions withheld from Owner, Group, and Other users. • If set to "", log files are created with the default access permissions established by the ambient umask for Owner, Group, and Other users. <p>For information about how to configure and use umask to control file permission settings, and about using the user file creation mode mask, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>

Table 36: sbrd.conf Syntax (continued)

Parameter	Function
RADIUS_HIGH_FDS	<ul style="list-style-type: none"> • If set to 0, management of file descriptors is disabled. • If set to 1, management of file descriptors is enabled. <p>Default value is 1.</p> <p>NOTE: RADIUS_HIGH_FDS should always be set to 1 unless Juniper Networks Technical Support advises otherwise.</p> <p>ULIMIT_OPEN_FILES should never be set less than 256. Together these parameters ensure that SBR Carrier is always able to open at least 256 regular files. This is especially important for sites that configure multiple Oracle plug-ins.</p>
ORACLE_MSB_FILE	<p>Specifies the absolute path to the locale-specific Oracle message file.</p> <ul style="list-style-type: none"> • If you enter "", Oracle will open multiple instances of this read-only file. • If RADIUS_HIGH_FDS is 1 and you specify a valid ORACLE_MSB_FILE (absolute path ending in a file name), SBR Carrier avoids opening multiple instances of this read-only file, thereby helping to ensure that the server is always able to open at least 256 regular files. This is especially important for sites that configure multiple Oracle plug-ins. <p>Default value is "".</p>
RADIUS	<p>Default value is "radius".</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>
RADIUSOPTS	<p>Specifies options used when running SBR Carrier.</p> <p>Default value is "".</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>
RADIUSARGS	<p>Default value is "sbr.xml".</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>

Table 36: sbrd.conf Syntax (*continued*)

Parameter	Function
RADIUS_PRIVATE_DIR	Default value is "\$RADIUSDIR". NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.

Table 36: sbrd.conf Syntax (*continued*)

Parameter	Function
RADIUS_LD_PRELOAD	

Table 36: sbrd.conf Syntax (continued)

Parameter	Function
	<p>Specifies an arbitrary space-separated list of libraries to be interposed on the RADIUS process. In particular, this parameter overrides mtmalloc with the new umem memory allocator.</p> <p>If commented out or set to "", the parameter does not override anything and the sbrd process uses the mtmalloc memory allocator as in previous releases.</p> <p>The default value <code>"/lib/libumem.so"</code>, uses the umem memory allocator, which provides improved memory handling, instead of mtmalloc.</p> <p>NOTE: In addition to improved performance considerations, the umem memory allocator offers optional debug features that are controlled by the UMEM_DEBUG and UMEM_LOGGING parameters. See the Solaris <code>umem_debug(3MALLOC)</code> manual pages for more information.</p> <p>UMEM_DEBUG</p> <p>This parameter enables and controls debug features of the umem memory allocator that is enabled by the RADIUS_LD_PRELOAD parameter.</p> <ul style="list-style-type: none"> • If commented out or set to "", disables umem debugging for better performance. In this case, the UMEM_LOGGING parameter must also be commented out or set to "". • If uncommented, this parameter should be set to "default" in order to enable umem debugging. In this case, the UMEM_LOGGING parameter must also be uncommented and set to "transaction". <p>Default value is commented out, that is, umem debugging are disabled for better performance.</p> <p>CAUTION: Enabling debug features of the umem memory allocator will noticeably impact SBR performance and memory utilization.</p> <p>UMEM_LOGGING</p> <p>This parameter enables and controls debug features of the umem memory allocator that is enabled by the RADIUS_LD_PRELOAD parameter.</p> <ul style="list-style-type: none"> • If commented out or set to "", disables umem in-memory

Table 36: sbrd.conf Syntax (continued)

Parameter	Function
	<p>debug logs for better performance. In this case, the UMEM_DEBUG parameter must also be commented out or set to "".</p> <ul style="list-style-type: none"> • If uncommented, should be set to "transaction" in order to enable umem in-memory debug logs. In this case, the UMEM_DEBUG parameter must also be uncommented and set to "default". <p>Default value is commented out, that is, umem in-memory debug logs are disabled for better performance.</p> <p>CAUTION: Enabling debug features of the umem memory allocator will noticeably impact SBR performance and memory utilization.</p>
WATCHDOGENABLE	<ul style="list-style-type: none"> • If set to 0, auto-restart, which restarts the SBR Carrier server if it fails, is disabled. • If set to 1, auto-restart is enabled. <p>Default value is 0.</p> <p>This parameter is set from your input to the Steel-Belted Radius Carrier configuration script.</p>
WATCHDOG	<p>Specifies the name of the auto-restart module.</p> <p>Default value is radiusd.</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>

Table 36: sbrd.conf Syntax (continued)

Parameter	Function
WATCHDOGOPTS	<p>Specifies the options for the auto-restart module.</p> <p>Options are:</p> <ul style="list-style-type: none"> • --config—Specifies the configuration file. • --pidfile—Specifies the file that contains the server process ID. • --logfile—Specifies the server log file name. If syslog is not available, log messages are written to the server log file. <p>NOTE: By default, the filename specified in the --config, --pidfile, or --logfile option is assumed to be located in the <code>\$RADIUSDIR</code> directory. If you want to specify a file in a different directory, you must add the directory path along with the filename—for example, <code>/tmp/radiusd.conf</code>.</p> <ul style="list-style-type: none"> • --debug—(Optional) Enables the debugging mode. • --syslog—(Optional) Specifies the syslog connection method. • --force—(Optional) Forces a server restart as needed. This option should be used with caution as hard kill signals (SIGKILL) may be sent as a last resort. <p>Default value is --config \$RADIUSDIR/radiusd.conf --pidfile \$RADIUSDIR/radius.pid --logfile \$RADIUSDIR/radiusd.log.</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>
WATCHDOGARGS	<p>Default value is \$RADIUSDIR/\$SELF.</p> <p>NOTE: Do not change this value unless instructed to do so by Juniper Networks Technical Support.</p>
GWRELAYENABLE	<p>Specifies whether to start the GWrelay process while executing the <code>./sbrd start</code> script.</p> <ul style="list-style-type: none"> • If set to 1, the GWrelay process is started. • If set to 0, the GWrelay process is not started. You should manually start the GWrelay process (if applicable) by using the <code>./sbrd start GWrelay</code> command. <p>NOTE: This parameter is automatically populated after running the SBR Carrier configuration script.</p>

Table 36: sbrd.conf Syntax (continued)

Parameter	Function
SHUTDOWNTIMEOUT	<p>Specifies the maximum number of seconds to wait for outstanding database transactions when the server is in the process of shutting down.</p> <p>You can enter a value in the range from 45 through 360 seconds. Default value is 45 seconds.</p> <p>NOTE: You must set this parameter based on the memory usage of SBR Carrier process. If the memory is in the range 8–20 GB, set this parameter to a value greater than 200 seconds. If the memory is greater than 20 GB, set this parameter to 360 seconds.</p>
SS7LDAP_ADDRESS=127.0.0.1	<p>This parameter is automatically populated after running the Steel-Belted Radius Carrier configuration script and answering “Yes” to the question: “Do you want to configure for use with SIGTRAN? [n]:”.</p> <p>Do not modify this parameter without consulting Juniper Networks Technical Support.</p>
SS7LDAP_PORT=389	<p>This parameter is automatically populated after running the Steel-Belted Radius Carrier configuration script and answering “Yes” to the question: “Do you want to configure for use with SIGTRAN? [n]:”.</p> <p>Do not modify this parameter without consulting Juniper Networks Technical Support.</p>
CUSTOM_JVM_PATH	<p>This parameter specifies the path where the libjvm.so file is located. This parameter also sets the LD_LIBRARY_PATH environment variable to point to the location of the libjvm.so file.</p> <p>NOTE: Do not edit this parameter manually. This parameter is automatically populated after running the SBR Carrier configuration script.</p>
WEBSERVER_JAVA_HOME	<p>This parameter specifies the path where the Java 1.8.0 or later version that is used to start the webserver for launching the Web GUI is installed in your system.</p> <p>NOTE: Do not edit this parameter manually. This parameter is automatically populated after running the SBR Carrier configuration script.</p>

services File

The **services** file can be used to assign default UDP ports for RADIUS communications to and from the SBR Carrier server. Steel-Belted Radius Carrier reads the **services** file at startup. Among the items of information in the **services** file are the port assignments for RADIUS authentication and accounting services.

Figure 1 on page 116 illustrates part of a sample **services** file.

Figure 1: Sample Services File

```
# This file contains port numbers for well-known services
# defined by IANA. Format:
#
# <service> <port number>/<protocol> [aliases...] [#<comment>]
#
echo          7/tcp
echo          7/udp
discard       9/tcp  sink null
discard       9/udp  sink null
systat        11/tcp  users    #Active users
systat        11/tcp  users    #Active users
daytime       13/tcp
```

The location of the **services** file is:

/etc/ (may be mapped using NIS or NIS+)

If no entry for **radius** or **radacct** is found in the **services** file, Steel-Belted Radius Carrier uses the default UDP ports (1645 and 1812 for authentication, 1646 and 1813 for accounting).

Steel-Belted Radius Carrier can be configured to use any available UDP ports for authentication and accounting:

1. Use a text editor to open the **services** file.
2. To set the port for authentication, set the value of the **radius** parameter. For example:

```
radius 1812/udp # RADIUS authentication protocol
```

3. To set the port for accounting, set the value of the **radacct** parameter. For example:

```
radacct 1813/udp # RADIUS accounting protocol
```

NOTE: Port number assignments made in the **radius.ini** file override the assignments made in this file. See “[Ports] Section” on page 91 for more information.

You can determine the ports that Steel-Belted Radius Carrier is using at any time by examining the server log file for that time period.

NOTE: If another RADIUS server is running on the same host, you must modify the port numbers in radius.ini [Ports] section to avoid port number conflicts if the other RADIUS server binds to the default ports before Steel-Belted Radius Carrier starts.

servtype.ini File

The **servtype.ini** file configures service type mapping in Steel-Belted Radius Carrier. Service type mapping allows a single user to have multiple authorization attribute sets based on the service type the user is requesting. The service type is determined based on request attributes using rules that may differ depending on the network access server.

Using static configuration parameters in the **servtype.ini** file, you can specify, on a device-by-device basis, a mapping of request attributes and values to service type strings. These strings can be attached to the username as a prefix or as a suffix. The elaborated username is used for both authentication and authorization, and for allowing different authorizations based on service type requested.

Refer to the *SBR Carrier Administration and Configuration Guide* for information about how to configure and use service type mapping.

[Settings] Section

The [Settings] section (Table 37 on page 118) of **servtype.ini** controls how the service type string is attached to the username before performing a lookup in the Native User database.

NOTE: If Prefix and Suffix are both set to 0 in the [Settings] section, service type mapping is disabled.

Table 37: servtype.ini [Settings] Syntax

Parameter	Function
Prefix	<p>Specifies whether the service type string is prefixed to the username before performing a lookup in the Native User database.</p> <ul style="list-style-type: none"> • If set to 1, the service type string is prefixed to the username. • If set to 0, the service type string is not prefixed to the username. <p>Default value is 0.</p>
Suffix	<p>Specifies whether the service type string is suffixed to the username before performing a lookup in the Native User database.</p> <ul style="list-style-type: none"> • If set to 1, the service type string is suffixed to the username. • If set to 0, the service type string is not suffixed to the username. <p>Default value is 0.</p>
Default	<p>Mapping name that is used when an Access-Request message is received from a network access server not listed in the [NAS] section of <i>servtype.ini</i>.</p> <p>If you do not configure a Default setting and the server cannot determine the mapping in any other way, the server ignores the service type and authenticates the user without it.</p>

[NAS] Section

The [NAS] section of the ***servtype.ini*** file lets you map network access devices to [Mapping] sections. The syntax for [NAS] is:

```
[NAS]
NASname = mappingName
NASname = mappingName
```

Each *NASname* entry in the [NAS] section must match the name of a RADIUS client entry in the Steel-Belted Radius Carrier database. When an Access-Request is received, its NAS-IP-Address attribute is matched to a RADIUS client entry in the database. If a match can be found and the RADIUS client name matches a *NASname* in the [NAS] section, Steel-Belted Radius Carrier looks for a corresponding mapping section in the ***servtype.ini*** file.

[MappingName] Section

Each [MappingName] section of the **servtype.ini** file identifies the strings to be added to the username for lookups in the Native User database, which allows Steel-Belted Radius Carrier to retrieve the appropriate return list, and specifies the rules an incoming Access-Request packet must meet before Steel-Belted Radius Carrier returns an Access-Accept message. The name of each [MappingName] section must match a *mappingName* entry in the [NAS] section.

The syntax for each [MappingName] section is:

```
[mapping]
ServiceTypeString
RADIUSattribute = value
~RADIUSattribute = value
```

ServiceTypeString is a string added to the username.

Each rule is a statement about an attribute that must be present in the incoming Access-Request packet. Each rule must be indented with a tab character, followed by a *RADIUSattribute = value* string, followed by a carriage return. Every component of the rule is optional, so there are many syntax variations.

If a rule includes a *RADIUSattribute* field, this field must identify a standard or vendor-specific RADIUS attribute that is known to the server. If a rule provides an optional *value* field, this field must name a valid possible value for that attribute.

If the *RADIUSattribute* field for a rule is preceded by a tilde (~), then the specified *RADIUSattribute*, if present in the Access-Request packet, must have a value other than *value* for the rule to be true. If the *RADIUSattribute* is not present in the Access-Request packet, or if it is present and has the *value* specified, the rule is false and authorization fails.

Example

```
[Settings]
Prefix=1
Suffix=0
Default=defaultmap

[NAS]
nas1=nas1map
nas2=nas2map

[nas1map]
ppp:
    Framed-Protocol=1
```

```

        Service-Type=2
vpn:
    Framed-Protocol=6
    ~Service-Type=2
other:
    Framed-Protocol
    Service-Type
[nas2map]
analog:
    NAS-Port-Type=1
isdn:
    NAS-Port-Type=2
[defaultmap]
ppp:

```

update.ini File

The **update.ini** initialization file controls what information is updated when Steel-Belted Radius Carrier receives a SIGHUP (1) or SIGUSR2 (17) signal, which is sent by means of the signal command.

When Steel-Belted Radius Carrier receives a SIGHUP (1) or SIGUSR2 (17) signal, it performs the tasks specified in the [HUP] and [USR2] sections of the **update.ini** file. You can perform tasks selectively by modifying **update.ini** to toggle specific settings; for example, you can issue a SIGHUP (1) signal to initiate one set of tasks, and then modify **update.ini** and issue another SIGHUP (1) signal to initiate a different set of tasks.

The **update.ini** file installed with Steel-Belted Radius Carrier causes Steel-Belted Radius Carrier to re-read all settings when it receives a SIGHUP (1) signal and to clear its statistics when it receives a SIGUSR2 (17) signal.

[HUP] and [USR2] Sections

The [HUP] section of **update.ini** specifies what tasks Steel-Belted Radius Carrier performs when it receives a SIGHUP (1) signal. The [USR2] section of **update.ini** specifies what tasks Steel-Belted Radius Carrier performs when it receives a SIGUSR2 (17) signal.

Example

```

[HUP]
UpdateLogAndTraceLevel = 1
UpdateLogfilePermissions = 1

```



```

UpdateProxy = 1
UpdateDHCPPools = 1
Update3GPP = 1
UpdateWiMAX = 1
UpdateAutoStop = 1
UpdateValuePools = 1
UpdatePlugins = 1
UpdateThreadsAndFloods = 1
UpdateStatusPeriodAndInfo = 1
UpdateSessionTable = 1
UpdateIPPPools = 1
UpdateEap = 1
UpdateAdminAccess = 1
UpdateStatLog = 1
ResetStats = 0
UpdateAccounting = 1
UpdateEnhancedRateStats = 1
UpdateEnhancedEAPLogging = 1

```

Table 38 on page 121 lists the settings that may be present in the [HUP] or [USR2] section of **update.ini**.

Table 38: update.ini Syntax

Parameter	Function
ResetStats	<ul style="list-style-type: none"> • If set to 0, do not reset Steel-Belted Radius Carrier statistics to 0 when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, reset Steel-Belted Radius Carrier statistics to 0 when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 0 in the [HUP] section. Default value is 1 in the [USR2] section.</p>
Update3GPP	<ul style="list-style-type: none"> • If set to 0, do not update 3GPP settings from 3gpp.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update 3GPP settings from 3gpp.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>

Table 38: update.ini Syntax (continued)

Parameter	Function
UpdateThreadsAndFloods	<ul style="list-style-type: none"> • If set to 0, threads and floods status report counters are not cleared when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, threads and floods status report counters are cleared when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateStatusPeriodAndInfo	<p>Reinitializes the (huppable) values in the [Status] section of the radius.ini file.</p> <ul style="list-style-type: none"> • If set to 0, status report period and enable/disable for auth/acct/proxy thread floods are not read. • If set to 1, status report period and enable/disable for auth/acct/proxy thread floods are read. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateWiMAX	<ul style="list-style-type: none"> • If set to 0, do not update WiMAX settings from the wimax.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update WiMAX settings from wimax.ini when a SIGHUP (1) or USR2 signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateAdminAccess	<ul style="list-style-type: none"> • If set to 0, do not update administrator access settings when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update administrator access settings when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>

Table 38: update.ini Syntax (continued)

Parameter	Function
UpdateAutoStop	<ul style="list-style-type: none"> • If set to 0, do not update the Proxy AutoStop settings (by re-reading the AcctAutoStopEnable setting in radius.ini) when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update the Proxy AutoStop settings (by re-reading the AcctAutoStopEnable setting in radius.ini) when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateDHCPPools	<ul style="list-style-type: none"> • If set to 0, do not update DHCP pool settings specified in dhcp.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update DHCP pool settings specified in dhcp.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateEAP	<ul style="list-style-type: none"> • If set to 0, do not update EAP settings specified in eap.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update EAP settings specified in eap.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateIPools	<ul style="list-style-type: none"> • If set to 0, do not update IP pool information when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update IP pool information when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateLogAndTraceLevel	<ul style="list-style-type: none"> • If set to 0, do not update log and trace levels specified in radius.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update log and trace levels specified in radius.ini when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>

Table 38: update.ini Syntax (continued)

Parameter	Function
UpdateLogfilePermissions	<ul style="list-style-type: none"> • If set to 0, do not update logfile permissions specified in the logfile configuration files when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update logfile permissions specified in the logfile configuration files when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdatePlugins	<ul style="list-style-type: none"> • If set to 0, do not update plug-ins that support dynamic re-reading of configuration settings when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update plug-ins that support dynamic re-reading of configuration settings when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p> <p>NOTE: The TLS, TTLS, and PEAP plug-ins currently support dynamic configuration updates.</p>
UpdateProxy	<ul style="list-style-type: none"> • If set to 0, do not update realm configuration when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update realm configuration (by re-reading proxy.ini, *.pro, and *.dir files) when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateSessionTable	<ul style="list-style-type: none"> • If set to 0, do not update current session table information when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, updates current session table when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>

Table 38: update.ini Syntax (continued)

Parameter	Function
UpdateValuePools	<ul style="list-style-type: none"> • If set to 0, do not update attribute value pool settings (in *.rr files when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, update attribute value pool settings (in *.rr files when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateStatLog	<ul style="list-style-type: none"> • If set to 0, do not update configuration in the statlog.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, reads statlog.ini file and the configuration is updated when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateAccounting	<p>If set to 0, do not update Accounting settings from the account.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received.</p> <p>If set to 1, update Accounting settings from the account.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received.</p> <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>
UpdateEnhancedRateStats	<p>If set to 0, do not update enhanced rate statistics settings from the radius.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received.</p> <p>If set to 1, update enhanced rate statistics settings from the radius.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received.</p> <p>Default value is 1 in the [HUP] section. Default value is 0 in the [USR2] section.</p>

Table 38: update.ini Syntax (*continued*)

Parameter	Function
UpdateEnhancedEAPLogging	<ul style="list-style-type: none"> • If set to 0, do not update the enhanced EAP logging setting from the radius.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received. • If set to 1, updates the enhanced EAP logging setting from the radius.ini file when a SIGHUP (1) or SIGUSR2 (17) signal is received. <p>Default value is 1 in the [HUP] section.</p> <p>Default value is 0 in the [USR2] section.</p>

Auto-Restart Files

When enabled, the auto-restart module (**radiusd** script), acts as a watchdog daemon monitoring the status of the SBR Carrier executable, and restarting it as needed. Auto-restart is disabled by default. Regardless of whether auto-restart is enabled, the SBR Carrier software is always started and stopped using this command:

```
/opt/JNPRsbr/radius/sbrd start radius
```

```
/opt/JNPRsbr/radius/sbrd stop radius
```

Auto-restart is enabled by configuring the WATCHDOG* parameters in the **sbrd.conf** file, and fine-tuned by configuring the **radiusd.conf** file. See [“sbrd.conf File” on page 104](#), and [“radiusd.conf File” on page 128](#).

Perl must be installed on the Steel-Belted Radius Carrier server if you want to use the automatic restart module.

NOTE: If Perl version 5 is not installed, the **radiusd** script will not run, even if enabled by configuration, and SBR Carrier will operate without the auto-restart module running.

Perl SNMP Support

You can configure the auto-restart module to send SNMP traps to record auto-restart events. Perl SNMP support resides in the Perl **SNMP_Session** module, which provides access to remote SNMP agents.

Perl SNMP support allows Steel-Belted Radius Carrier to send SNMP traps to a variety of SNMP agents, including the Sun Management Center, which is distributed with some Sun hardware platforms. Sun Management Center is not required to run **radiusd**.

Perl System Log Support

The optional perl package **syslog.ph** is used to log the watchdog daemon status. You can configure auto-restart to send system log messages to record auto-restart events. To use system log reporting, you can use the **h2ph** utility to create a **syslog.ph** file. This example assumes **site_perl/5.005** is in **@INC**:

```
su - root
cd /user/include/sys
/usr/perl15/bin/h2ph -d /usr/perl15/site_perl/5.005 syslog.h
```

If you do not want to use system log, use the **-d** or **--logfile** options for the **radiusd** command to open a regular log file (**radiusd.log**).

sbrd.conf File

To enable the auto-restart module, you must edit the **sbrd.conf** file using the following procedure:

NOTE: The following procedure is dependent upon you enabling the auto startup scripts when you install SBR Carrier.

1. If SBR Carrier is already running, become superuser and type this command to stop the server:

```
/opt/JNPRsbr/radius/sbrd stop radius
```

2. Edit **sbrd.conf** to set **WATCHDOGENABLE=1** (or 0 to disable)
3. Type this command to restart the server:

```
/opt/JNPRsbr/radius/sbrd start radius
```

NOTE: If you are running the optional SSR module, the command to start/stop only the SSR cluster is **/opt/JNPRsbr/radius/sbrd start/stop ssr**. This controls SSR without starting or stopping the SBR.

radiusd.conf File

The default **radiusd.conf** settings cause the auto-restart feature to work this way:

If the **radius** server executable fails to respond to status polling from **radiusd** within 17 seconds, **radiusd** attempts to stop **radius** using **SIGTERM** (a polite shutdown). If **radius** does not shut down within 60 seconds, **SIGKILL** (a hard kill) is used to stop it. After shutdown by either method, **radiusd** starts a new **radius** child process. If this radius child does not respond to status polling within 60 seconds of startup, it is presumed dead; a misconfiguration of the server is assumed; and **radiusd** terminates with a critical error.

While the auto-restart module is enabled, all informational, debugging, warning, error, and critical messages from **radiusd** are recorded here:

- **Syslog**—Messages are written to the **syslog** system logging facility.
- **Log file**—If **syslog** is not available, messages are written to the server log file specified using the **--logfile** option in the **sbrd.conf** **WATCHDOGOPTS** parameter.

NOTE: If Perl is not installed in the **/usr/local/bin/** directory, this error message occurs when you start the *SBR Carrier* server:

./S90sbrd: /RadiusHome/radiusd: not found

To fix this error, edit the first line of the **radiusd** file in the **RADIUS** directory so that the directory structure points to the correct Perl interpreter executable:

#!/usr/local/bin/perl

radiusd.conf Configuration File

The **radiusd.conf** configuration file ([Table 39 on page 128](#)) provides settings for the **radiusd** auto-restart module.

Table 39: radiusd.conf Syntax

radiusd.conf Parameter	Function
WatchdogIntervalPing	Number of seconds the automatic-restart module waits between sending status inquiries. Default value is 5 seconds.
WatchdogIntervalMaxPong	Number of seconds the automatic-restart module waits for a reply before issuing a SIGTERM (shutdown) message. Default value is 17 seconds.

Table 39: radiusd.conf Syntax (*continued*)

radiusd.conf Parameter	Function
WatchdogIntervalMaxStartup	<p>Number of seconds during which the server is expected to be able to start up.</p> <p>Default value is 60 seconds.</p>
WatchdogIntervalMaxShutdown	<p>Number of seconds during which the server is expected to be able to shut down.</p> <p>Default value is 60 seconds.</p>
SnmpManager = <i>hostname</i> <i>community port version</i>	<p>Identifier for an SNMP management station you want to receive traps from the automatic-restart module. You can specify more than one SNMP management station.</p> <p>For each SNMP management station, enter:</p> <ul style="list-style-type: none"> • <i>hostname</i>—IP address of the SNMP management station. • <i>community</i>—SNMP community string. • <i>port</i>—UDP port number used for SNMP trap messages. UDP port 162 is the default. • <i>version</i>—SNMP version number. Default value is 1. <p>If SnmpManager is undefined, SNMP traps may still be logged, but are not transmitted on the network.</p>
SnmplInterface	<p>Identifies the IP network interface to be used to generate SNMP trap messages. You can specify interfaces by name or by IP address.</p> <p>If you enter any, the first IPv4 interface the automatic-restart module finds is used.</p> <p>If you leave this parameter blank, generation of SNMP trap messages is disabled.</p>

Table 39: radiusd.conf Syntax (continued)

radiusd.conf Parameter	Function
SnmpCommandTrap	<p>Specifies how SNMP trap messages are forwarded:</p> <ul style="list-style-type: none"> You can specify the pathname and filename for a module or executable whose syntax matches the SMC snmptrap utility. For example: <code>/opt/SUNWsymon/util/bin/sparc-sun-solaris2.8/snmptrap</code> You can specify SNMP_Session.pm to deliver SNMP traps to the management station using the Perl modules. If you leave the parameter blank, SNMP trap messages are not generated. Default value is blank.
SnmpCommandUptime	<p>Specifies how the automatic-restart module determines elapsed time for timestamps in trap messages.</p> <p>You can specify the pathname and filename for a module or executable whose syntax matches the SMC uclock utility. For example: <code>/opt/SUNWsymon/util/bin/sparc-sun-solaris2.8/uclock</code></p> <p>If you leave the parameter blank, the automatic restart module calculates elapsed time relative to its own start time. Default value is blank.</p>
SnmpEnterprise	<p>Specifies the OID prefix for enterprise-specific trap messages, which is used to select the appropriate MIB for decoding traps.</p> <p>Default value is 1.3.6.1.4.1.1411.1.1.</p> <p>If you leave the parameter blank, SNMP trap messages are not generated.</p>
SnmpGenericTrapType= 6	<p>Specifies the enterprise-specific trap type, which must be 6 according to the SNMPv1 standard. Do not change this value without a specific reason.</p>

Table 39: radiusd.conf Syntax (continued)

radiusd.conf Parameter	Function
SnmpTrapWatchdogStarted	<p>Specifies the trap type for messages indicating that the automatic-restart module is started.</p> <p>Default value is 113.</p> <p>Enter 0 to disable this type of trap.</p>
SnmpTrapWatchdogStopped	<p>Specifies the trap type for messages indicating that the automatic-restart module is stopped.</p> <p>Default value is 114.</p> <p>Enter 0 to disable this type of trap.</p>
SnmpTrapWatchdogRadiusStarted	<p>Specifies the trap type for messages indicating that the RADIUS server is restarted.</p> <p>Default value is 115.</p> <p>Enter 0 to disable this type of trap.</p>
SnmpTrapWatchdogRadiusTerm	<p>Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has sent the SIGTERM signal.</p> <p>Default value is 5028.</p> <p>Enter 0 to disable this type of trap.</p>
SnmpTrapWatchdogRadiusKill	<p>Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has sent the KILL signal.</p> <p>Default value is 5029.</p> <p>Enter 0 to disable this type of trap.</p>
SnmpTrapWatchdogAborted	<p>Specifies the trap type for messages indicating that the RADIUS server is not responding and that the automatic-restart module has given up and terminated.</p> <p>Default value is 10051.</p> <p>Enter 0 to disable this type of trap.</p>

Table 39: radiusd.conf Syntax (*continued*)

radiusd.conf Parameter	Function
SnmpTrapWatchdogFailedInit	<p>Specifies the trap type for messages indicating that the automatic-restart module failed to start, which may indicate a misconfiguration issue.</p> <p>Default value is 10052.</p> <p>Enter 0 to disable this type of trap.</p>

Authentication Configuration Files

IN THIS CHAPTER

- [authlog.ini File | 133](#)
- [authReport.ini File | 144](#)
- [authReportAccept.ini File | 147](#)
- [authReportBadSharedSecret.ini File | 151](#)
- [authReportReject.ini File | 155](#)
- [authReportUnknownClient.ini File | 162](#)
- [blacklist.ini File | 166](#)
- [lockout.ini File | 167](#)
- [redirect.ini File | 169](#)
- [statlog.ini File | 171](#)

This chapter describes the usage and settings for the initialization files used by Steel-Belted Radius Carrier to authenticate users and to record the results of authentication events. Initialization files are loaded at startup time. The following topics are included in this chapter:

NOTE: Throughout this chapter, the term *attributes* refers to both standard RADIUS *attributes* and structured attributes. For information about specifying structured attributes, see [“Structured Attributes” on page 199](#).

authlog.ini File

The **authlog.ini** initialization file contains information that controls how RADIUS authentication request attributes are logged in the comma-delimited `yyyymmdd.authlog` file.

[Alias/name] Sections

You can create one or more [Alias/name] sections in **authlog.ini** (Table 40 on page 134) to associate attributes of different names, but identical meaning. For example, one network access server vendor might call an attribute **Auth-Connect-Type** and another might call it **AuthConn-Typ**, yet the two attributes would both map to **Auth-Conn-Type**.

Each [Alias/name] section permits you to map one RADIUS authentication request attribute that is already being logged by Steel-Belted Radius Carrier to any number of other attributes. You can provide as many [Alias/name] sections as you want, using this syntax for each section:

```
[Alias/name]
VendorSpecificAttribute=
VendorSpecificAttribute=
```

Table 40: authlog.ini [Alias/name] Syntax

Parameter	Function
name	The preferred attribute name. The name attribute must be one that you are currently logging to a column in the Steel-Belted Radius Carrier authentication request log file (.authlog). Therefore, it must be listed in the [Attributes] section of authlog.ini .
VendorSpecificAttribute	Each entry is given on one line. An equal sign (=) must immediately follow each VSA name, without any intervening space. Improperly formatted entries are considered invalid and are ignored.

Each *VendorSpecificAttribute* in the list is logged to the *name* column in the authentication request log file. Because you are listing these attributes in an [Alias/name] section, make sure they are not listed in the [Attributes] section or they are logged to their own columns as well as to the *name* column.

All of the attribute names that you reference in an [Alias/name] section must be defined in a dictionary file that is already installed on the Steel-Belted Radius Carrier server. This includes *name* and each *VendorSpecificAttribute* entry.

In the following example, the standard RADIUS attribute **Auth-Conn-Type** is mapped to the vendor-specific attributes **AuthConnect-Type** and **AuthConn-Typ**. Values encountered for all three attributes are logged in the AuthOctetPackets column in the authentication request log file:

```
[Alias/Auth-Conn-Type]
Auth-Conn-Typ=
Auth-Connect-Type=
```

[Attributes] Section

The [Attributes] section of **authlog.ini** lists all the attributes logged in the authentication request log file. These include attributes in Access-Request messages received from the network access server (NAS). Attributes that Steel-Belted Radius Carrier returns to the NAS are not logged in this file. When you install Steel-Belted Radius Carrier, the **authlog.ini** file is set up so that all standard RADIUS attributes and all supported vendor authentication attributes are listed.

You can configure what is logged to the authentication request log file by rearranging the order of attributes in the [Attributes] section. You can delete or comment out attributes you do not want or that do not apply to your equipment. This lets you design the content and column order of any spreadsheets that you plan to create based upon the authentication request log file.

The syntax of the [Attributes] section is:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attributes]
User-Name=
NAS-IP-Address=
NAS-Port=
Service-Type=
Framed-Protocol=
Framed-IP-Address=
Framed-IP-Netmask=
Framed-Compression=
```

The [Attributes] section lists one *AttributeName* on each line. You must ensure that an equal sign (=) immediately follows each *AttributeName*, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each *AttributeName* in the [Attributes] section must be defined in a standard RADIUS dictionary file (**.dict** file), a subattribute dictionary file (**.jdict** file), or vendor-specific dictionary file (**.dict**) installed on the Steel-Belted Radius Carrier server.

NOTE: The first five attributes in each authentication log file entry (Date, Time, RASClient, FullName, and ACC/REJ) are always enabled, and cannot be reordered or deleted. Therefore, these attributes do not appear in the **authlog.ini** file [Attributes] section.

[Configuration] Section

The [Configuration] section of **authlog.ini** specifies the location of the **yyymmdd.authlog** file.

Table 41: authlog.ini [Configuration] Syntax

Parameter	Function
LogDir	<p>Specifies the destination directory on the local host where yyymmdd.authlog files are stored.</p> <p>Default value is the directory where Steel-Belted Radius Carrier is installed.</p> <p>NOTE: With directed realms, you can maintain multiple authentication log locations.</p>

[Syslog] Section

The [Syslog] section of the **authlog.ini** file enables authentication request information to be written to the system log file. Compatible applications (such as rsyslog) can be used to forward these system log messages to a remote server or database. The format of the system log message is the same as that of the authentication log message.

Parameter	Function
Enable	<p>Enables authentication request information to be written to the system log file.</p> <p>If set to 1, this setting enables writing of authentication requests to the system log file.</p> <p>If set to 0, this setting disables writing of authentication requests to the system log file.</p> <p>The default value is 0.</p> <p>NOTE: This setting is independent of the Enable setting in the [Configure] section of the authentication log.</p>
Facility	<p>This parameter sets the system log facility.</p> <p>The default value is Daemon, but could be set to Local[X], where X = 0–7.</p>
Severity	<p>This parameter sets the severity of the system log message. The value could be Info or Notice.</p> <p>The default value is Info.</p>

Following are some configuration examples of **syslog.conf** and **rsyslog.conf** files:

Example 1—To write all authlog messages to /var/adm/messages using the LOCAL3 facility and LOG_INFO severity:

1. Configure the **authlog.ini** file as:

```
[Syslog]
Enable = 1
Facility = local3
Severity = Info
```

2. Add the following statement in the /etc/syslog.conf file:

```
*.err;kern.debug;daemon.notice;mail.crit;local3.info    /var/adm/messages
```

3. Run the following command:

```
kill -HUP `pgrep syslogd`
```

4. Restart the sbrd process.

```
./sbrd restart
```

5. Authlog messages are written to the system log (/var/adm/messages).

NOTE: In Linux, you need to use the **rsyslog.conf** file instead of **syslog.conf**. By default, system logs are stored under /var/log/messages.

Example 2—To write all authlog messages to a SQL database (Linux configuration example):

1. Configure the **authlog.ini** file in the local server as:

```
[Syslog]
Enable = 1
Facility = daemon
Severity = Info
```

2. Restart the sbrd process.

```
./sbrd restart
```

3. Update the `/etc/rsyslog.conf` file in the local server as:

```
#### MODULES ####
$Modload ommysql #provide Mysql support
$ModLoad imuxsock.so # provides support for local system logging (e.g. via logger
    command)
$ModLoad imklog.so # provides kernel logging support (previously done by rklogd)
$ModLoad immark.so # provides --MARK-- message capability

#Provides TCP system log  reception
$ModLoad imtcp.so
$InputTCPServerRun 514

$template test,"insert into rsyslog5(host,pid,facility,priority,datetime,msg)
values('%hostname%',

'%syslogtag:R,ERE,1,BLANK:\[([0-9]{1,5})\]--end%', '%syslogfacility%', '%syslogpriority%',
'%timereported:::date-mysql%', '%msg%')",SQL

*. * >sbr-lin1.englab.juniper.net,test,testuser,testpassword;test
```

NOTE: The statement `*. * >sbr-lin1.englab.juniper.net,test,testuser,testpassword;test` is in the format `*. * >dbhost,dbname,dbuser,dbpassword;dbtemplate`.

4. Restart the rsyslog service.

```
Service rsyslog restart
```

5. Authlog messages are written to the SQL DB of the SQL server.

Example 3—To write all authlog messages to a remote server (Linux configuration example):

1. Configure the **authlog.ini** file in the local server as:

```
[Syslog]
Enable = 1
```

```
Facility = daemon
Severity = Info
```

2. Restart the sbrd process.

```
./sbrd restart
```

3. Update the `/etc/rsyslog.conf` file in the local server as:

```
*.* @@192.168.1.1:514
```

NOTE: Here, **192.168.1.1:514** is a remote SBR server.

4. Restart the rsyslog service.

```
Service rsyslog restart
```

5. Update the `/etc/rsyslog.conf` file in the remote server as:

```
# Provides TCP system log reception
$ModLoad imtcp.so
$InputTCPServerRun 514
*.info;mail.none;authpriv.none;cron.none /var/log/messages
```

6. Restart the rsyslog service.

```
Service rsyslog restart
```

7. Authlog messages are written to the remote server's system log (`/var/log/messages`).

[Settings] Section

Steel-Belted Radius Carrier writes all authentication request data to the current authentication request log file (`yyyymmdd.authlog`) until that log file is closed. When Steel-Belted Radius Carrier closes an

authentication request log file, it immediately opens a new one and begins writing authentication request data to it.

You can configure how often this rollover of the authentication request log file occurs.

The naming conventions of the authentication request log files support the fact that Steel-Belted Radius Carrier can create more than one file per day. In the examples in [Table 42 on page 140](#), *y* =year digit, *m* =month digit, *d* =day digit, and *h* =hour digit. The extra sequence number *_nnnnn* starts at *_00000* each day.

Table 42: Authentication Log Rollover

File Generation Method	File Naming Convention
Default (24 hours)	yyyymmdd.authlog
Non-24-hour rollover	yyyymmdd_hhmm.authlog
Rollover due to size	yyyymmdd_nnnnn.authlog
Rollover due to size or startup when non24hour time in effect	yyyymmdd_hhmm_nnnnn.authlog

The [Settings] section of **authlog.ini** in [Table 43 on page 140](#) controls which entries are written to the authentication request log file, and ensure the compatibility of these entries with a variety of database systems. These rollover settings can be present in the [Settings] section.

Table 43: authlog.ini [Settings] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, the authentication request log is disabled and other settings are ignored. • If set to 1, the authentication request log is enabled. <p>Set Enable to 1 for Authentication servers. For efficiency, set Enable to 0 for nonauthentication servers.</p> <p>Default value is 0.</p>

Table 43: authlog.ini [Settings] Syntax (continued)

Parameter	Function
LogAssignedIpAddress	<ul style="list-style-type: none"> • If set to 1, LogAssignedIpAddress is enabled and the framed IP address is displayed in the authlog.log file as Assigned-IP-Address. • If set to 0, LogAssignedIpAddress is disabled and the framed IP address is not displayed in the authlog.log file. <p>The default value is 0 (Disabled).</p> <p>Here is a sample output displaying the header and log message</p> <p>Header:</p> <pre>"Date", "Time", "RAS-Client", "Full-Name", "Acc/Rej", "User-Name", "NAS-IP-Address", "NAS-Port", "Service-Type", "Framed-Protocol", "Framed-IP-Address", "Framed-IP-Netmask", "Framed-Compression", >Login-IP-Host", "Callback-Number", "State", "Called-Station-Id", "Calling-Station-Id", "NAS-Identifier", "Proxy-State", "Event-Timestamp", "NAS-Port-Type", "Port-Limit", "Login-LAT-Port", "Assigned-IP-Address"</pre> <p>Log Message:</p> <pre>"11/11/2010", "01:42:51", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1975", "t1.internet", "2", "10.206.144.1" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1976", "t1.internet", "2", "10.206.144.5" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1977", "t1.internet", "2", "10.206.144.7" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1978", "t1.internet", "2", "10.206.144.8" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1979", "t1.internet", "2", "10.206.144.10" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1980", "t1.internet", "2", "10.206.144.14" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1981", "t1.internet", "2", "10.206.144.15" "11/11/2010", "01:43:06", "<ANY>", "ROOT", "ACCEPT", "ROOT", "10.206.144.123", "1982", "t1.internet", "2", "10.206.144.16"</pre>

Table 43: authlog.ini [Settings] Syntax (continued)

Parameter	Function
LogFilePermissions	<p>Specifies the owner and access permission setting for the authorization request log (yyyyymmdd.authlog) file.</p> <p>Enter a value for the LogFilePermissions setting in <i>owner:group permissions</i> format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, userw:1007 rw-r - - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.</p> <p>The default mask is -rw - - - - -.</p>
LineSize	<p>Specifies the maximum number of characters in a line in the authentication request log. You can enter a number in the range 1024–32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>
MaxSize	<ul style="list-style-type: none"> • If set to a number greater than 0, specifies the maximum number of bytes for an authentication request log file. If the authentication request log file equals or exceeds this limit when the log size is checked, the log file is closed and a new file started. • If set to 0, the authentication request log has no maximum size. <p>Default value is 0.</p>
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the authentication request log entries.</p> <p>Default value is 1.</p>

Table 43: authlog.ini [Settings] Syntax (continued)

Parameter	Function
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the authentication request log entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the authentication request log entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the authentication request log entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the authentication request log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the authentication request log entries.</p> <p>Default value is 1.</p>
RollOver	<p>Specifies how often the current authentication request log file is closed and a new file opened (a rollover), up to one rollover per minute.</p> <ul style="list-style-type: none"> • If set to 0, the authentication request log rolls over once every 24 hours, at midnight local time. • If set to a number in the range 1–1440, specifies the number of minutes until the next rollover. <p>Default value is 0.</p>

Table 43: authlog.ini [Settings] Syntax (continued)

Parameter	Function
RollOverOnStartup	<ul style="list-style-type: none"> • If set to 1, each time Steel-Belted Radius Carrier is started, it closes the current authentication request log file and opens a new one. A sequence number <i>_nnnnn</i> is appended to the log file name, just as when MaxSize is reached. • If set to 0, each time Steel-Belted Radius Carrier is started, it appends entries to the previously open authentication request log file. <p>Default value is 0.</p>
Titles	<ul style="list-style-type: none"> • If set to 1, each time a new authentication request log file is created, the title line (containing column headings) is written to the file. • If set to 0, the line is not written. <p>Default value is 1.</p>
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

authReport.ini File

The **authReport.ini** initialization file controls whether Steel-Belted Radius Carrier generates the following reports:

- Authentication acceptance report
- Authentication rejection report
- Unknown authentication client report
- Invalid shared secret report

If enabled, these reports are written to the *radiusdir/authReports* directory on the Steel-Belted Radius Carrier server.

NOTE: The settings in the **authReport.ini** file are overwritten when the Web GUI is used to enable or disable these reports.

[AcceptReport] Section

The [AcceptReport] section of **authReport.ini** ([Table 44 on page 145](#)) enables or disables generation of the authentication acceptance report. The settings for the authentication acceptance report are specified in the **authReportAccept.ini** file, which is described on [“authReportAccept.ini File” on page 147](#).

Sample syntax is:

```
[AcceptReport]
Enable = 1
```

Table 44: authReport.ini [AcceptReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">• If set to 1, Steel-Belted Radius Carrier periodically generates the authentication acceptance report.• If set to 0, the authentication acceptance report is not generated. Default value is 0.

[BadSharedSecretReport] Section

The [BadSharedSecretReport] section of **authReport.ini** ([Table 45 on page 146](#)) enables or disables generation of the invalid shared secret report. The settings for the invalid shared secret report are specified in the **authReportBadSharedSecret.ini** file, which is described on [“authReportBadSharedSecret.ini File” on page 151](#).

Sample syntax is:

```
[BadSharedSecretReport]
Enable = 1
```

Table 45: authReport.ini [BadSharedSecretReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier periodically generates the invalid shared secret report. • If set to 0, the invalid shared secret report is not generated. <p>Default value is 0.</p>

[RejectReport] Section

The [RejectReport] section of **authReport.ini** ([Table 46 on page 146](#)) enables or disables generation of the authentication rejection report. The settings for the authentication rejection report are specified in the **authReportReject.ini** file, which is described on [“authReportReject.ini File” on page 155](#).

Sample syntax is:

```
[RejectReport]
Enable = 1
```

Table 46: authReport.ini [RejectReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier periodically generates the authentication rejection report. • If set to 0, the authentication rejection report is not generated. <p>Default value is 0.</p>

[UnknownClientReport] Section

The [UnknownClientReport] section of **authReport.ini** ([Table 47 on page 147](#)) enables or disables generation of the unknown authentication client report. The settings for the unknown authentication client report are specified in the **authReportUnknownClient.ini** file, which is described on [“authReportUnknownClient.ini File” on page 162](#).

Sample syntax is:

```
[UnknownClientReport]
Enable = 1
```

Table 47: authReport.ini [UnknownClientReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier periodically generates the unknown authentication client report. • If set to 0, the unknown authentication client report is not generated. <p>Default value is 0.</p>

authReportAccept.ini File

The **authReportAccept.ini** initialization file specifies options for the authentication acceptance report, which is an ASCII comma-delimited file that contains information about successful authentications by the Steel-Belted Radius Carrier server.

[Attributes] Section

The [Attributes] section of **authReportAccept.ini** lists the attributes logged in the acceptance log.

You can configure what is logged to the acceptance report by entering attributes in the [Attributes] section in the sequence you want them to appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the acceptance report.

The syntax of the [Attributes] section is:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attributes]
User-Name=
NAS-IP-Address=
```

The [Attributes] section lists one *AttributeName* on each line. You must ensure that an equal sign (=) immediately follows each *AttributeName*, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each *AttributeName* in the [Attributes] section must be defined in a standard RADIUS dictionary file (.dct file), a subattribute dictionary file (.jdict file), or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius Carrier server.

NOTE: The first four attributes in each acceptance report entry (Date, Time, RASClient, and FullName) are always enabled, and cannot be reordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the **authReportAccept.ini** file.

[Settings] Section

The [Settings] section of **authReportAccept.ini** ([Table 48 on page 148](#)) specifies the operational characteristics of the authentication acceptance report.

If the **MaxMinutesPerFile** parameter is set to 0, the file name of the authentication acceptance report is `accepts_yyyymmdd.csv` (where *yyyymmdd* identifies the date the report was generated.) If the **MaxMinutesPerFile** parameter is set to a value greater than 0, the file name of the report is `accepts_yyyymmdd_hhmm.csv` (where *yyyymmdd* identifies the date and *hhmm* identifies the time the report was generated.)

Sample syntax is:

```
[Settings]
BufferSize = 131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = user:1007 rw-r- - - -
MaxMinutesPerFile = 0
QuoteInteger = 1
QuoteIpAddress = 1
QuoteBinary = 1
QuoteText = 1
QuoteTime = 1
UTC = 0
```

Table 48: **authReportAccept.ini** [Settings] Syntax

Parameter	Function
BufferSize	The size of the buffer used in the logging process, in bytes. Default value is 131072.

Table 48: authReportAccept.ini [Settings] Syntax (*continued*)

Parameter	Function
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains each authentication acceptance report.</p> <p>Default value is 1 (one day).</p>
LineSize	<p>The maximum size of a single log line. The allowable range is 1024 to 32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>
LogFilePermissions	<p>Specifies the owner and access permission setting for the authentication acceptance log (accepts_yyyymmdd.csv) file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p>
MaxMinutesPerFile	<p>Specifies how often the current authentication acceptance report is closed and a new file opened.</p> <ul style="list-style-type: none"> • If set to <i>n</i> (where <i>n</i> is a number greater than 0), a new report is generated every <i>n</i> minutes. • If set to 0, a new report is generated once every 24 hours, at midnight local time. <p>Default value is 0.</p> <p>NOTE: The value entered for MaxMinutesPerFile determines the file name of the generated report.</p>

Table 48: authReportAccept.ini [Settings] Syntax (*continued*)

Parameter	Function
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>

Table 48: authReportAccept.ini [Settings] Syntax (*continued*)

Parameter	Function
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

authReportBadSharedSecret.ini File

The **authReportBadSharedSecret.ini** initialization file specifies options for the invalid shared secret report, which is an ASCII comma-delimited file that records information about requests received from known RADIUS clients that used an invalid shared secret. This condition is only detectable if the authentication request contained a Message-Authenticator attribute, which is required if credentials are of an EAP type but optional if credentials are PAP, CHAP, or MS-CHAP v2. (In the case of PAP, an invalid shared secret is not detected, but results in an Access-Reject response because the user password is decrypted into incorrect characters.)

If the **MaxMinutesPerFile** parameter is set to 0, the file name of the bad shared secret report is **badSharedSecret_yyyymmdd.csv** (where *yyyymmdd* identifies the date the report was generated.) If the **MaxMinutesPerFile** parameter is set to a value greater than 0, the file name of the report is **badSharedSecret_yyyymmdd_hhmm.csv** (where *yyyymmdd* identifies the date and *hhmm* identifies the time the report was generated).

[Attributes] Section

The [Attributes] section of **authReportBadSharedSecret.ini** lists the attributes logged in the invalid shared secret report.

You can configure what is logged to the invalid shared secret report by entering attributes in the [Attributes] section in the sequence you want them to appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the silent discard/bad shared secret report.

The syntax of the [Attributes] section is:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attributes]
User-Name=
NAS-IP-Address=
```

The [Attributes] section lists one *AttributeName* on each line. You must ensure that an equal sign (=) immediately follows each *AttributeName*, with no spaces in between. Improperly formatted entries are ignored.

Each *AttributeName* in the [Attributes] section must be defined in a standard RADIUS dictionary file (.dct file), a subattribute dictionary file (.jdict file), or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius Carrier server.

NOTE: The first three attributes in each invalid shared secret report entry (Date, Time, and RADIUSClient) are always enabled, and cannot be reordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the **authReportBadSharedSecret.ini** file.

[Settings] Section

The [Settings] section of **authReportBadSharedSecret.ini** specifies the operational characteristics of the invalid shared secret report. Sample syntax is:

```
[Settings]
BufferSize = 131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = user:1007 rw-r- - - -
MaxMinutesPerFile = 0
QuoteBinary = 1
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
UTC = 0
```

Table 49: authReportBadSharedSecret.ini [Settings] Syntax

Parameter	Function
BufferSize	The size of the buffer used in the logging process, in bytes. Default value is 131072.

Table 49: authReportBadSharedSecret.ini [Settings] Syntax (*continued*)

Parameter	Function
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains each invalid shared secret report.</p> <p>Default value is 1 (one day).</p>
LineSize	<p>The maximum size of a single log line. The allowable range is 1024 to 32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>
LogFilePermissions	<p>Specifies the owner and access permission setting for the invalid shared secret report (badSharedSecret_yyyymmdd.csv) file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p>
MaxMinutesPerFile	<p>Specifies how often the current report is closed and a new file opened.</p> <ul style="list-style-type: none"> • If set to <i>n</i> (where <i>n</i> is a number greater than 0), a new report file is generated every <i>n</i> minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. <p>Default value is 0.</p> <p>NOTE: The value entered for MaxMinutesPerFile determines the file name of the generated report.</p>

Table 49: authReportBadSharedSecret.ini [Settings] Syntax (continued)

Parameter	Function
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>

Table 49: authReportBadSharedSecret.ini [Settings] Syntax (continued)

Parameter	Function
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

authReportReject.ini File

The **authReportReject.ini** initialization file specifies options for the authentication rejection report, which is an ASCII comma-delimited file that records authentication rejections.

If the **MaxMinutesPerFile** parameter is set to 0, the file name of the authentication rejection report is **rejects_yyyymmdd.csv** (where **yyyymmdd** identifies the date the report was generated.) If the **MaxMinutesPerFile** parameter is set to a value greater than 0, the file name of the report is **rejects_yyyymmdd_hhmm.csv** (where **yyyymmdd** identifies the date and **hhmm** identifies the time the report was generated).

[Attributes] Section

The [Attributes] section of **authReportReject.ini** lists the attributes logged in the authentication rejection report.

You can configure what is logged to the authentication rejection report by entering attributes in the [Attributes] section in the sequence you want them to appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the reject report.

The syntax of the [Attributes] section is:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attributes]
Service-Type=
Source-IP-Address=
Source-UDP-Port=
```

The [Attributes] section lists one *AttributeName* on each line. You must ensure that an equal sign (=) immediately follows each *AttributeName*, with no spaces in between. Improperly formatted entries are ignored.

Each *AttributeName* in the [Attributes] section must be defined in a standard RADIUS dictionary file (.dct file), a subattribute dictionary file (.jdict file), or vendor-specific dictionary file (.dct) installed on the Steel-Belted Radius Carrier server.

The following attributes in each authentication rejection report entry are always enabled, and cannot be reordered or deleted:

- Date—Identifies the date of the authentication rejection.
- Time—Identifies the time of the authentication rejection.
- RADIUS-Client—Identifies the RADIUS client that received the authentication rejection.
- User-Name—Identifies the name of the user that was rejected.
- Reject-Method—Identifies the most relevant authentication method that rejected the user. If this information is unavailable, the parameter is set to **Unknown**.
- Rejected-Device—Identifies the MAC address or the outer NAI of the device that was rejected. If this information is unavailable, the parameter is set to **Unknown**.
- Reject-Reason—Identifies the reason for the authentication rejection. [Table 50 on page 156](#) describes the reject reason codes supported by SBRC.

Table 50: Reject Reason Codes

Reason Code	Reject Reason
AUTH_ERR_001	EAP-NAK received; client requesting EAP protocol 0,21
AUTH_ERR_003	Filter (ASNGW_JS) script execution failed
AUTH_ERR_004	Unable to find user with matching password
AUTH_ERR_005	EAP-NAK received; client requesting EAP protocol 0,13
AUTH_ERR_006	Received request with unmatched state attribute
AUTH_ERR_007	EAP-TTLS: Required User-Name attribute not present in inner authentication request
AUTH_ERR_008	EAP-TTLS authentication failed - client issued alert for invalid certificate type
AUTH_ERR_011	Server issued alert as unknown root certificate authority

Table 50: Reject Reason Codes (*continued*)

Reason Code	Reject Reason
AUTH_ERR_012	No mobility keys found for NAI
AUTH_ERR_013	Client issued alert as client closed the session before handshake was completed
AUTH_ERR_014	Tunneled authentication rejected
AUTH_ERR_016	Required Message-Authenticator attribute missing
AUTH_ERR_017	Too many or too few authentication attributes in request
AUTH_ERR_018	Conflicting authentication methods in packet
AUTH_ERR_019	Missing User-Name attribute in request
AUTH_ERR_020	Multiple User-Name attributes in request
AUTH_ERR_021	User-Name attribute in request too long
AUTH_ERR_022	Correlation ID not assigned
AUTH_ERR_023	Request contained invalid payload
AUTH_ERR_026	User is blocklisted
AUTH_ERR_029	Invalid Session-Timeout value
AUTH_ERR_032	Unable to get session record
AUTH_ERR_036	Proxy authentication failed
AUTH_ERR_037	SQL Error 0 resulted in hard failure
AUTH_ERR_038	Failed to initialize cache for request
AUTH_ERR_040	System error
AUTH_ERR_041	General post-processing error
AUTH_ERR_042	Username or credential incorrect

Table 50: Reject Reason Codes (*continued*)

Reason Code	Reject Reason
AUTH_ERR_043	Invalid credentials
AUTH_ERR_044	<ul style="list-style-type: none"> Invalid credential or user Rejecting request username not matching the regular expression configured in ValidateAuth (radius.ini) <p>NOTE:</p> <ul style="list-style-type: none"> In case of Invalid Password scenario in proxy directed realm case, Instead of printing "Tunneled authentication reject" for TTLS. "AUTH_ERR_043", "user found, but password validation failed" for TTLS with SQL In case of Invalid Password scenario in proxy directed realm case, Instead of printing "Tunneled authentication reject" for TTLS. "AUTH_ERR_044" and "ldap auth user not authenticated" for TTLS with LDAP.
AUTH_ERR_045	User locked out
AUTH_ERR_046	Access error
AUTH_ERR_047	Invalid request
AUTH_ERR_048	Unknown error
AUTH_ERR_049	EAP Challenge Timeout due to delayed client
AUTH_ERR_050	EAP Challenge Timeout due to unresponsive client
AUTH_ERR_097	Error retrieving IDs and MIP from challenge cache

- Reject-Log—Identifies the reason for the authentication request in language supplied by the authentication method. If a reason is not supplied, the parameter is set to **Unavailable**.

These attributes do not appear in the [Attributes] section of the **authReportReject.ini** file.

NOTE: If you modify the [Attributes] section and then restart the SBR Carrier, a new log file **reject_yyyymmdd_nnnnn.csv** is created.

[Settings] Section

The [Settings] section of **authReportReject.ini** specifies the operational characteristics of the authentication rejection report. Sample syntax is:

```
[Settings]
UTC = 0
BufferSize = 131072
MaxMinutesPerFile = 0
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = user:1007 rw-r- - - -
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
QuoteBinary = 1
```

Table 51: authReportReject.ini [Settings] Syntax

Parameter	Function
BufferSize	<p>Specifies the size of the buffer used in the logging process, in bytes.</p> <p>Default value is 131072.</p>
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains each rejection report.</p> <p>Default value is 1 (one day).</p>
LineSize	<p>Specifies the maximum size of a single log line. The allowable range is 1024 to 32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>

Table 51: authReportReject.ini [Settings] Syntax (*continued*)

Parameter	Function
LogFilePermissions	<p>Specifies the owner and access permission setting for the authentication rejection report (rejects_yyyymmdd.csv) file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p>
MaxMinutesPerFile	<p>Specifies how often the current report is closed and a new file opened.</p> <ul style="list-style-type: none"> • If set to <i>n</i> (where <i>n</i> is a number greater than 0), a new report file is generated every <i>n</i> minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. <p>Default value is 0.</p> <p>NOTE: The value entered for MaxMinutesPerFile determines the file name of the generated report.</p>
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>

Table 51: authReportReject.ini [Settings] Syntax (*continued*)

Parameter	Function
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

authReportUnknownClient.ini File

The **authReportUnknownClient.ini** initialization file specifies options for the unknown authentication client report, which is an ASCII comma-delimited file produced by the Steel-Belted Radius Carrier server that identifies requests received from unknown RADIUS clients.

If the **MaxMinutesPerFile** parameter is set to 0, the file name of the unknown authentication client report is **unknownClient_yyyymmdd.csv** (where *yyymmdd* identifies the date the report was generated.) If the **MaxMinutesPerFile** parameter is set to a value greater than 0, the file name of the report is **unknownClient_yyyymmdd_hhmm.csv** (where *yyymmdd* identifies the date and *hhmm* identifies the time the report was generated.)

[Attributes] Section

The [Attributes] section of **authReportUnknownClient.ini** lists the attributes logged in the unknown client report.

You can configure what is logged to the unknown client log by entering attributes in the [Attributes] section in the sequence you want them to appear. This lets you design the content and column order of any spreadsheets that you plan to create based upon the unknown client log.

The syntax of the [Attributes] section is:

```
[Attributes]
AttributeName=
AttributeName=
```

For example:

```
[Attributes]
User-Name=
```

The [Attributes] section lists one *AttributeName* on each line. You must ensure that an equal sign (=) immediately follows each *AttributeName*, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each *AttributeName* in the [Attributes] section must be defined in a standard RADIUS dictionary file (**.dct** file), a subattribute dictionary file (**.jdict** file), or vendor-specific dictionary file (**.dct**) installed on the Steel-Belted Radius Carrier server.

NOTE: The first six attributes in each unknown client report entry (Date, Time, Source-IP-Address, Source-UDP-Port, Target-IP-Address, and Target-UDP-Port) are always enabled, and cannot be reordered or deleted. Therefore, these attributes do not appear in the [Attributes] section of the **authReportUnknownClient.ini** file.

[Settings] Section

The [Settings] section of **authReportUnknownClient.ini** specifies the operational characteristics of the unknown authentication client report. Sample syntax is:

```
[Settings]
BufferSize = 131072
DaysToKeep = 1
LineSize = 4096
LogfilePermissions = user:1007 rw-r- - - -
MaxMinutesPerFile = 0
QuoteBinary = 1
QuoteInteger = 1
QuoteIpAddress = 1
QuoteText = 1
QuoteTime = 1
UTC = 0
```

Table 52: authReportUnknownClient.ini [Settings] Syntax

Parameter	Function
BufferSize	<p>The size of the buffer used in the logging process, in bytes.</p> <p>Default value is 131072.</p>
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains each unknown client report.</p> <p>Default value is 1 (one day).</p>
LineSize	<p>The maximum size of a single log line. The allowable range is 1024 to 32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>

Table 52: authReportUnknownClient.ini [Settings] Syntax (continued)

Parameter	Function
LogFilePermissions	<p>Specifies the owner and access permission setting for the unknown authentication client report (unknownClient_yyyymmdd_hhmm.csv) file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r - - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p>
MaxMinutesPerFile	<p>Specifies how often the current report is closed and a new file opened.</p> <ul style="list-style-type: none"> • If set to n (where n is a number greater than 0), a new report file is generated every n minutes. • If set to 0, a new report file is generated once every 24 hours, at midnight local time. <p>NOTE: The value entered for MaxMinutesPerFile determines the file name of the generated report.</p> <p>Default value is 0.</p>
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>

Table 52: authReportUnknownClient.ini [Settings] Syntax (*continued*)

Parameter	Function
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the report are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the application that processes the entries.</p> <p>Default value is 1.</p>
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

blacklist.ini File

The **blacklist.ini** configuration file enables and configures blacklist settings. Only one profile can be created for the purposes of blocklisting, and any login attempt that matches that profile is blocked. An authentication request matches the blacklist profile if the attributes in the request match the check list attributes of the profile. The profile can contain multiple attributes, and if any of the attributes match those of the profile, the attempt is rejected.

The **blacklist.ini** file contains one configuration section called [Settings] ([Table 53 on page 166](#)), which has the following settings:

```
[Settings]
Enable = <0|1>
IncludeProxy = <0|1>
Profile = profile
```

Table 53: blacklist.ini Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, blocklisting is enabled. • If set to 0, blocklisting is disabled. <p>Default value is 0.</p>
IncludeProxy	<ul style="list-style-type: none"> • If set to 1, blocklisting is configured to include proxy requests, meaning that it is applied to all authentication requests. • If set to 0, blocklisting is configured only to local authentication requests. <p>Default value is 0.</p>
Profile	Specifies the name of the blacklist profile in the Steel-Belted Radius Carrier database.

The following example enables the blacklist feature and specifies Steel-Belted Radius Carrier use the **BlockedNumbers** profile to filter authentication requests.

```
[Settings]
Enable = 1
IncludeProxy = 0
Profile = BlockedNumbers
```

The **BlockedNumbers** profile called by this **blacklist.ini** file specifies check list attributes Steel-Belted Radius Carrier uses to reject authentication requests. The following entries in the **BlockedNumbers** profile identify **Calling-Station-Id** phone numbers used by rogue users you want to block.

```
Calling-Station-Id = 617-999-9119
Calling-Station-Id = 800-515-7889
```

lockout.ini File

The **lockout.ini** configuration file enables and configures account lockout settings. Account lockout lets you disable an account after a configurable number of failed login attempts within a configurable period. For example, if a user enters an incorrect password three times within two minutes, Steel-Belted Radius Carrier can lock out the user’s account temporarily. During the lockout period, the user cannot log in, even with the correct password. Attempts to authenticate against a locked out account cause Steel-Belted Radius Carrier to respond with an Access-Reject message immediately.

The **lockout.ini** file contains one configuration section called [Settings] ([Table 54 on page 167](#)), which has settings similar to the following:

--	--

Table 54: lockout.ini [Setting] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, lockout is disabled. • If set to 1, lockout is enabled. <p>Default value is 0.</p>
Lockout	<p>Specifies the lockout period in seconds.</p> <p>Default value is 600 seconds (10 minutes).</p>
Rejects	<p>Specifies the number of rejected attempts before lockout.</p> <p>Default value is 3.</p>
Within	<p>Specifies the period in seconds during which a specified number of rejects causes a lockout.</p> <p>Default value is 120 seconds (two minutes).</p>

[ClientExclusionList] Section

You can add a ClientExclusionList section to the **lockout.ini** file. Use this section to list clients that are excepted from the lockout functionality. Enter one client name per line. For example,

```
[ClientExclusionList]
exampleclient1
exampleclient2
```

[UserExclusionList] Section

You can add a UserExclusionList section to the **lockout.ini** file. Use this section to prevent certain reserved usernames, such as anonymous, from being locked out. Enter one username per line. For example:

NOTE: If you enable the lockout facility in Steel-Belted Radius Carrier and you use a tunneled authentication method (TTLS or PEAP) with a prefetch-capable method (native user, SQL, or LDAP) and an enabled EAP protocol (MS-CHAP v2, MD5-Challenge, TLS), then you must enable Handle via Auto-EAP First in that prefetch-capable method to prevent the outer username (anonymous) from being added to the lockout list.

Otherwise, when Steel-Belted Radius Carrier receives an authentication request that uses an unconfigured EAP method, Steel-Belted Radius Carrier rejects the user (because the EAP method is not configured) and add the outer username (anonymous) to its lockout list. This results in all users with an outer authentication name of anonymous being rejected until the lockout period expires.

NOTE: When running a Session State Register cluster, the account lockout configuration (**lockout.ini**) and state information (number of times each user has supplied a wrong password in a given time period) is maintained locally on each server in the cluster, not in the high-availability database. Consequently, a user who is locked out on one SBR Carrier server can request access from a different SBR Carrier server participating in the same Session State Register cluster.

redirect.ini File

Account redirection lets you flag an account for special processing after a configurable number of failed login attempts within a configurable time period. The **redirect.ini** initialization file specifies the settings used for account redirection when users forget or mis-enter their passwords.

[Settings] Section

The [Settings] section of **redirect.ini** ([Table 55 on page 169](#)) enables and configures account redirection settings.

Table 55: redirect.ini [Settings] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, account redirection is disabled. • If set to 1, account redirection is enabled. <p>Default value is 0.</p> <p>NOTE: Account redirection and account lockout are incompatible. Do not enable account redirection if account lockout is enabled.</p>
Lockout	<p>The number of seconds in the account redirection lockout period. For example, a lockout period of 86,400 seconds locks a user out for one day if account redirection processing fails to authenticate the user.</p> <p>Default value is 600 seconds (10 minutes).</p>
Profile	<p>The name of the global profile that supplies the values and attributes used for the user after account redirection is triggered.</p> <p>Default value is Redirect.</p>
Redirect	<p>The number of seconds during which a user is in redirect state. If the redirection period elapses without another user authentication request, the user is returned to normal state.</p> <p>Default value is 120 seconds.</p>
Rejects	<p>The number of rejected attempts before redirection.</p> <p>Default value is 3.</p>

Table 55: redirect.ini [Settings] Syntax (*continued*)

Parameter	Function
Within	The period in seconds during which a specified number of rejects causes account redirection. Default value is 180 seconds (3 minutes).

For example, the following [Settings] section of **redirect.ini** specifies that, if a user fails authentication three times within 180 seconds, the user account is placed into redirect state. If the user does not submit another authentication request within 120 seconds of entering redirect state, the user account is restored to normal state.

```
[Settings]
Enable = 0
Rejects = 3
Within = 180
Redirect = 120
Profile = RedirectProfile
Lockout = 86400
```

If the user submits another authentication request within 120 seconds of entering redirect state, the user is accepted without authentication or authorization processing, the user's account is placed into accept-pending state, and the RADIUS accept message for the user contains the values and attributes specified in the global **RedirectProfile** profile. (These values or attributes can be used by an external customer process to direct the user to a secure webpage that asks for alternative authentication information or billing information; the external process might then mail the user an access password if the user satisfies the external process requirements.)

When a user is in accept-pending state, the user's next authentication request determines whether Steel-Belted Radius Carrier accepts or locks out the user:

- If the next authentication is successful, the user account is returned to normal state.
- If the next authentication fails to accept the user, the user account is locked out for 86,400 seconds (one day). During this lockout period, authentication requests for this user are rejected automatically, even if the user enters the correct password.

[ClientExclusionList] Section

The [ClientExclusionList] section of **redirect.ini** identifies the RADIUS clients that are excluded from account redirection processing. Each entry in the [ClientExclusionList] section of **redirect.ini** consists of the name of a RADIUS client device, as configured in the Steel-Belted Radius Carrier database.

statlog.ini File

The **statlog.ini** initialization file configures the Steel-Belted Radius Carrier statistics log file (**yyyymmdd.statlog**), which periodically records server statistics to a comma-delimited ASCII file. The statistics log provides a mechanism for creating snapshots of user-selected server statistics.

NOTE: The statistics per NAD client and per Called-Station-ID are not captured in the statlog file.

- The first line in a ***.statlog** file lists all column headings in double quotes ("Date", "Time", ...).
- The first column in a ***.statlog** file identifies the current date in *yyyy-mm-dd* format in double-quotes ("2006-02-13"). The ***.statlog** file uses the local date, not the date in the UTC time zone, when it records date information.
- The second column in a ***.statlog** file identifies the current time in *hh:mm:ss* format in double-quotes ("14:13:22"). The ***.statlog** file uses the local time, not the time in the UTC time zone, when it records time information.
- If statistics logging is enabled, a new statistics logging file is created the first time the server is started each day. At midnight, the server closes the old statistics log file and starts a new one with a file name that reflects the new date.
- If you restart the Steel-Belted Radius Carrier server and a ***.statlog** file exists for the current day, the server appends new information to the existing ***.statlog** file. When the server is restarted, the timer for capturing statistics snapshots is restarted; for example, a server configured to record statistics every 10 minutes captures statistics at 14:10:00. If the server is restarted at 14:15:00, it captures system statistics immediately (14:15:00) and 10 minutes thereafter (14:25:00); it does not try to capture statistics at 14:20:00 (10 minutes after the capture before the restart).
- If you change the order or contents of the list of statistics to be recorded and restart the server, Steel-Belted Radius Carrier detects the change and writes an entry with the new column headers to the current ***.statlog** file before writing new data records into the file.

NOTE: When you change the order or contents of the list of statistics recorded in the ***.statlog** file, Steel-Belted Radius Carrier creates the **statloghdr.dat** checkpoint file in the **radiusdir** directory. Do not modify or delete the **statloghdr.dat** file.

[Settings] Section

The [Settings] section of **statlog.ini** ([Table 56 on page 172](#)) specifies whether the statistics log file is enabled, who can access the statistics log file, how frequently the server writes information to the statistics log file, and the number of days statistics log files are retained.

Table 56: statlog.ini [Settings] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, the Steel-Belted Radius Carrier server periodically writes statistics information to the yyyymmdd.statlog file. • If set to 0, the Steel-Belted Radius Carrier server does not update the yyyymmdd.statlog file. <p>Default value is 0.</p>
LogDir	<p>Specifies the destination directory on the local host where statistics log files (yyyymmdd.statlog) are stored.</p> <p>Default value is the directory where SBR Carrier is installed.</p>
LogFilePermissions	<p>Specifies the owner and access permission setting for the *.statlog file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r- - - - specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and other users cannot access the log file.</p>

Table 56: statlog.ini [Settings] Syntax (*continued*)

Parameter	Function
Interval-Seconds	<p>Specifies the number of seconds that the SBR Carrier server waits before writing new statistics information to the statistics log.</p> <p>You can enter a value ranging from 10 through 3600 seconds. On Solaris systems, this value ranges from 1 through 3600 seconds.</p> <p>Default value is 600 seconds (10 minutes).</p> <p>NOTE: The written statistics information may become garbled under extreme load when the interval is less than 60 seconds.</p>
Days-To-Keep	<p>Specifies the number of days (in the range 1–365) the statistics log is retained by the Steel-Belted Radius Carrier server. When the specified number of days has elapsed, the statistics log is automatically purged.</p> <p>Default value is 7 days.</p>

For example:

```
[Settings]
Enable = 1
;LogfilePermissions = someuser:1007 rw-r- - - -
;Interval-Seconds = 600
;Days-To-Keep = 7
LogDir = /opt/JNPRsbr/
```

[Statistics] Section

The [Statistics] section of **statlog.ini** ([Table 57 on page 174](#)) identifies the statistics you want included in the snapshot. Each entry in this section takes the format *Source/Statistic*, where *Source* identifies the LCI statistics container that holds the statistic counter you want and *Statistic* identifies the statistic by name.

Statistics are written to the log file in the order in which they are listed in the [Statistics] section.

Table 57: statlog.ini [Statistics] Syntax

Parameter	Function
<i>Source</i>	<p>Specifies the name of the LCI statistics container that holds the specified statistic.</p> <p>Supported values for <i>Source</i> are:</p> <ul style="list-style-type: none"> • Server • Authentication • Accounting • Proxy • Rate • KPI
<i>Statistic</i>	<p>Specifies the name of the statistic you want to record in the log.</p> <p>Statistics for the Server are:</p> <ul style="list-style-type: none"> • Accounting-Threads • Authentication-Threads • Proxy-Threads • High-Acct-Threads • High-Acct-Threads-Since-Reset • High-Auth-Threads • High-Proxy-Threads • High-Auth-Threads-Since-Reset • High-Total-Threads • High-Total-Threads-Since-Reset • High-Proxy-Threads-Since-Reset • Max-Acct-Threads • Max-Auth-Threads • Max-Total-Threads • Max-Proxy-Threads • Total-Threads

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
-----------	----------

Table 57: statlog.ini [Statistics] Syntax (continued)

Parameter	Function
	<p>Authentication statistics are:</p> <ul style="list-style-type: none"> • Accept—Specifies the total number of Access-Accept responses sent from SBR Carrier to NAD • Dropped-Packet—Specifies the total number of RADIUS authentication requests dropped by SBR Carrier because the flood queue is exceeded • Failed-Authentication—Specifies the total number of failed authentication requests that are rejected because the username, password, or shared secret is invalid • Failed-On-Check-List—Specifies the total number of requests that are authenticated by SBR Carrier but did not meet the checklist requirements • Insufficient-Resources—Specifies the total number of authentication requests that are rejected due to a problem in internal (such as memory) or pooled (such as IP addresses and concurrency slots) resources • Invalid-Request—Specifies the total number of invalid RADIUS authentication requests received by SBR Carrier. Invalid requests are caused by one of the following reasons: <ul style="list-style-type: none"> • NAD sends incorrectly formed packets to SBR Carrier. • NAD does not conform to the RADIUS standard. • Proxy-Failure—Specifies the number of times a proxy RADIUS authentication packet has failed because of one of the following reasons: <ul style="list-style-type: none"> • The proxy RADIUS server is not able to find a target server. • All targets in a proxy realm are in fast-fail mode. • Reject—Specifies the total number of Access-Reject responses sent from SBR Carrier to NAD • Rejected-By-Proxy—Specifies the number of authentication reject responses received from the proxy RADIUS target server • Silent-Discard—Specifies the number of authentication requests that are silently discarded by the realm-selection script, or because of unknown clients, unknown request types, malformed requests, unavailable CST, or phantom creation failure. The silent discard also occurs when no shared secret is configured for the client, or if the request does not match the defined length or contains an invalid Message-Authenticator size. <p>NOTE: The Reject counter is not incremented if the Silent-Discard</p>

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
	<p>counter is incremented.</p> <ul style="list-style-type: none">• Total-Retry-Packets—Specifies the total number of duplicate authentication requests received by SBR Carrier• Total-Transactions—Specifies the total number of authentication requests received by SBR Carrier irrespective of whether the requests are accepted, rejected, or silently discarded• Transactions-Retried—Specifies the number of authentication requests for which duplicates are received by SBR Carrier

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
-----------	----------

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
	<p>Accounting statistics are:</p> <ul style="list-style-type: none"> • Dropped-Packet—Specifies the total number of RADIUS accounting packets dropped by SBR Carrier because the flood queue is exceeded • Insufficient-Resources—Specifies the total number of accounting requests that are silently discarded because of a problem in internal or pooled resources, or database (Oracle or JDBC plug-ins) configuration failure. The database configuration failure is caused by one of the following reasons: <ul style="list-style-type: none"> • The RADIUS server is unable to read the product attribute of a client. • The dictionary attribute of a client is missing or invalid. • The database attribute type is not consistent with the dictionary attribute type. • Invalid-Client—Specifies the total number of accounting requests that are silently discarded because the clients are unknown • Invalid-Request—Specifies the total number of invalid RADIUS accounting requests received by SBR Carrier. Invalid requests are caused by one of the following reasons: <ul style="list-style-type: none"> • NAD sends incorrectly formed packets to SBR Carrier. • NAD does not conform to the RADIUS standard. • Invalid-Shared-Secret—Specifies the total number of accounting requests silently discarded because the shared secret entered is incorrect • Interim—Specifies the total number of Accounting-Interim requests received by SBR Carrier • Off—Specifies the total number of Accounting-Off requests received by SBR Carrier • On—Specifies the total number of Accounting-On requests received by SBR Carrier • Proxy-Failure—Specifies the number of times a proxy RADIUS accounting packet has failed due to one of the following reasons: <ul style="list-style-type: none"> • The proxy RADIUS server is not able to find a target server. • All targets in a proxy realm are in fast-fail mode. • Start—Specifies the total number of Accounting-Start requests received by SBR Carrier • Stop—Specifies the total number of Accounting-Stop requests

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
	<p>received by SBR Carrier</p> <ul style="list-style-type: none">• Total-Retry-Packets—Specifies the total number of duplicate accounting requests received by SBR Carrier• Total-Transactions—Specifies the total number of Accounting-Start, Accounting-Stop, Accounting-Interim, Accounting-On, and Accounting-Off requests received by SBR Carrier• Transactions-Retried—Specifies the number of accounting requests for which duplicates are received

Table 57: statlog.ini [Statistics] Syntax (continued)

Parameter	Function
	<p>Proxy statistics are:</p> <ul style="list-style-type: none"> • Accounting—Specifies the total number of accounting requests forwarded from the proxy server to the target RADIUS server • Authentication—Specifies the total number of authentication requests forwarded from the proxy server to the target RADIUS server • Insufficient-Resources—Specifies the number of proxy requests that are failed or discarded due to one of the following reasons: <ul style="list-style-type: none"> • Memory allocation failed while the proxy response was being parsed • Socket error occurred while the proxy response was being received • No IP addresses were available in the IP pool while the proxy response was being parsed • Invalid-Response—Specifies the number of invalid proxy RADIUS responses received from the target server. Invalid responses are caused by one of the following reasons: <ul style="list-style-type: none"> • The target RADIUS server sends incorrectly formed packets to SBR Carrier. • The target RADIUS server does not conform to the RADIUS standard. • The message authenticator is invalid. • Invalid-Shared-Secret—Specifies the number of proxy responses that are dropped because the signature is invalid • Timed-Out—Specifies the number of proxy requests that failed despite several retry attempts • Total-Retry-Packets—Specifies the total number of proxy retries made by the proxy server • Total-Transactions—Specifies the total number of authentication and accounting requests that are forwarded from the proxy server to the target RADIUS server • Transactions-Retried—Specifies the number of proxy requests for which one or more transmission retries are made

Table 57: statlog.ini [Statistics] Syntax *(continued)*

Parameter	Function
	<p>Statistics for the Rate are:</p> <ul style="list-style-type: none"> • Acct-Start-Average-Rate • Acct-Start-Current-Rate • Acct-Start-Peak-Rate • Acct-Stop-Average-Rate • Acct-Stop-Current-Rate • Acct-Stop-Peak-Rate
	<ul style="list-style-type: none"> • Auth-Accept-Average-Rate • Auth-Accept-Current-Rate • Auth-Accept-Peak-Rate • Auth-Reject-Average-Rate • Auth-Reject-Current-Rate • Auth-Reject-Peak-Rate • Auth-Request-Average-Rate • Auth-Request-Current-Rate • Auth-Request-Peak-Rate
	<p>Statistics for the Rate are:</p> <ul style="list-style-type: none"> • Proxy-Acct-Fail-Proxy-Average-Rate • Proxy-Acct-Fail-Proxy-Current-Rate • Proxy-Acct-Fail-Proxy-Peak-Rate • Proxy-Acct-Request-Average-Rate • Proxy-Acct-Request-Current-Rate • Proxy-Acct-Request-Peak-Rate • Proxy-Auth-Rej-Proxy-Average-Rate • Proxy-Auth-Rej-Proxy-Current-Rate • Proxy-Auth-Rej-Proxy-Error-Average-Rate

Table 57: statlog.ini [Statistics] Syntax (continued)

Parameter	Function
	<ul style="list-style-type: none"> • Proxy-Auth-Rej-Proxy-Error-Current-Rate • Proxy-Auth-Rej-Proxy-Error-Peak-Rate • Proxy-Auth-Rej-Proxy-Peak-Rate • Proxy-Auth-Request-Average-Rate • Proxy-Auth-Request-Current-Rate • Proxy-Auth-Request-Peak-Rate • Proxy-Fail-Badresp-Average-Rate • Proxy-Fail-Badresp-Current-Rate • Proxy-Fail-Badresp-Peak-Rate • Proxy-Fail-Badsecret-Average-Rate • Proxy-Fail-Badsecret-Current-Rate • Proxy-Fail-Badsecret-Peak-Rate
	<ul style="list-style-type: none"> • Proxy-Fail-Missingresr-Average-Rate • Proxy-Fail-Missingresr-Current-Rate • Proxy-Fail-Missingresr-Peak-Rate • Proxy-Fail-Timeout-Average-Rate • Proxy-Fail-Timeout-Current-Rate • Proxy-Fail-Timeout-Peak-Rate • Proxy-Retries-Average-Rate • Proxy-Retries-Current-Rate • Proxy-Retries-Peak-Rate
	<p>Key Performance Indicators (KPI) statistics are:</p> <ul style="list-style-type: none"> • KPI/Challenge-Objects—Specifies the number of challenge cache objects used by SBR Carrier at a given time • KPI/Max-Challenge-Objects—Specifies the configured challenge cache objects maximum limit • KPI/High-Challenge-Objects-Since-Reset—Specifies the peak number of challenge cache objects in use since the start of SBR Carrier or since the statistics reset

For example:

```
[Statistics]
Server/Authentication-Threads
Server/Accounting-Threads
```

Server/Total-Threads
Server/Max-Acct-Threads
Server/Max-Auth-Threads
Server/Max-Total-Threads
Server/High-Auth-Threads
Server/High-Acct-Threads
Server/High-Total-Threads
Server/High-Acct-Threads-Since-Reset
Server/High-Auth-Threads-Since-Reset
Server/High-Total-Threads-Since-Reset

Authentication/Accept
Authentication/Reject
Authentication/Silent-Discard
Authentication/Total-Transactions
Authentication/Dropped-Packet
Authentication/Invalid-Request
Authentication/Failed-Authentication
Authentication/Failed-On-Check-List
Authentication/Insufficient-Resources
Authentication/Proxy-Failure
Authentication/Rejected-By-Proxy
Authentication/Transactions-Retried
Authentication/Total-Retry-Packets

Accounting/Start
Accounting/Stop
Accounting/Interim
Accounting/On
Accounting/Off
Accounting/Total-Transactions
Accounting/Dropped-Packet
Accounting/Invalid-Request
Accounting/Invalid-Client
Accounting/Invalid-Shared-Secret
Accounting/Insufficient-Resources
Accounting/Proxy-Failure
Accounting/Transactions-Retried
Accounting/Total-Retry-Packets

Proxy/Authentication
Proxy/Accounting
Proxy/Total-Transactions
Proxy/Timed-Out

Proxy/Invalid-Response
 Proxy/Invalid-Shared-Secret
 Proxy/Insufficient-Resources
 Proxy/Transactions-Retried
 Proxy/Total-Retry-Packets

 Rate/Auth-Request-Current-Rate
 Rate/Auth-Request-Average-Rate
 Rate/Auth-Request-Peak-Rate
 Rate/Auth-Accept-Current-Rate
 Rate/Auth-Accept-Average-Rate
 Rate/Auth-Accept-Peak-Rate
 Rate/Auth-Reject-Current-Rate
 Rate/Auth-Reject-Average-Rate
 Rate/Auth-Reject-Peak-Rate
 Rate/Acct-Start-Current-Rate
 Rate/Acct-Start-Average-Rate
 Rate/Acct-Start-Peak-Rate
 Rate/Acct-Stop-Current-Rate
 Rate/Acct-Stop-Average-Rate
 Rate/Acct-Stop-Peak-Rate
 Rate/Proxy-Auth-Request-Current-Rate
 Rate/Proxy-Auth-Request-Average-Rate
 Rate/Proxy-Auth-Request-Peak-Rate
 Rate/Proxy-Acct-Request-Current-Rate
 Rate/Proxy-Acct-Request-Average-Rate
 Rate/Proxy-Acct-Request-Peak-Rate
 Rate/Proxy-Fail-Timeout-Current-Rate
 Rate/Proxy-Fail-Timeout-Average-Rate
 Rate/Proxy-Fail-Timeout-Peak-Rate
 Rate/Proxy-Fail-Badresp-Current-Rate
 Rate/Proxy-Fail-Badresp-Average-Rate
 Rate/Proxy-Fail-Badresp-Peak-Rate
 Rate/Proxy-Fail-Badsecret-Current-Rate
 Rate/Proxy-Fail-Badsecret-Average-Rate
 Rate/Proxy-Fail-Badsecret-Peak-Rate
 Rate/Proxy-Fail-Missingresr-Current-Rate
 Rate/Proxy-Fail-Missingresr-Average-Rate
 Rate/Proxy-Fail-Missingresr-Peak-Rate
 Rate/Proxy-Retrieves-Current-Rate
 Rate/Proxy-Retrieves-Average-Rate
 Rate/Proxy-Retrieves-Peak-Rate
 Rate/Proxy-Auth-Rej-Proxy-Current-Rate
 Rate/Proxy-Auth-Rej-Proxy-Average-Rate

Rate/Proxy-Auth-Rej-Proxy-Peak-Rate
Rate/Proxy-Acct-Fail-Proxy-Current-Rate
Rate/Proxy-Acct-Fail-Proxy-Average-Rate
Rate/Proxy-Acct-Fail-Proxy-Peak-Rate
Rate/Proxy-Auth-Rej-Proxy-Error-Current-Rate
Rate/Proxy-Auth-Rej-Proxy-Error-Average-Rate
Rate/Proxy-Auth-Rej-Proxy-Error-Peak-Rate

KPI/Challenge-Objects
KPI/Max-Challenge-Objects
KPI/High-Challenge-Objects-Since-Reset

Attribute Processing Files

IN THIS CHAPTER

- Dictionary Files | 187
- Structured Attributes | 199
- classmap.ini File | 212
- filter.ini File | 213
- sample.rr File | 220
- spi.ini File | 222
- vendor.ini File | 224
- Adding NAS Location Information to Access-Request Messages | 230

This chapter describes the usage and settings for the Steel-Belted Radius Carrier attribute processing and dictionary files that control RADIUS attributes. The following topics are included in this chapter:

Dictionary Files

For each product listed in the **vendor.ini** file, Steel-Belted Radius Carrier provides **.dct** (text) and **.dic** (xml) dictionary files. Dictionary files enable Steel-Belted Radius Carrier to exchange attributes with RADIUS or Diameter clients. Like initialization files, dictionary files are loaded at startup time, and reside in the Steel-Belted Radius Carrier directory.

.dct Files

The **.dct** dictionary files identify the attributes Steel-Belted Radius Carrier expects when receiving RADIUS requests from a specific type of device. The **.dct** dictionary files also identify the attributes Steel-Belted Radius Carrier includes when sending a RADIUS response to a specific type of device. [Figure 2 on page 188](#) illustrates the format of a dictionary file.

Figure 2: Sample Dictionary (.dct) File

```
#####
# Juniper.dct - RADIUS dictionary for Juniper M-160 and M-40Es

# (See README.DCT for more details on the format of this file)
#####
# Use the RADIUS specification attributes
#
@radius.dct

#
# Juniper specific parameters
#
MACRO Juniper-VSA(t,s) 26 [vid=4874 type1=%t% len1=+2 data=%s%]

ATTRIBUTE Juniper-Local-User-Name      Juniper-VSA(1, string) r
ATTRIBUTE Juniper-Allow-Commands       Juniper-VSA(2, string) r
ATTRIBUTE Juniper-Deny-Commands        Juniper-VSA(3, string) r
ATTRIBUTE Juniper-Allow-Configuration  Juniper-VSA(4, string) r
ATTRIBUTE Juniper-Deny-Configuration   Juniper-VSA(5, string) r

#####
# Juniper.dct - Juniper Networks dictionary
#####
```

.dct File Location

The **.dct** dictionary files must be placed in the same directory as the Steel-Belted Radius Carrier daemon. During initialization, Steel-Belted Radius Carrier reads the file **dictionary.dcm** in the server directory to get a list of files with an extension of **.dct** (standard dictionary files) and uses the list to create a primary dictionary, which includes all known attributes.

.dct File Records

Records in a **.dct** dictionary file must begin with one of the keywords listed in [Table 58 on page 188](#).

Table 58: Dictionary File Keywords

Keyword	Function
@	Include the referenced file
ATTRIBUTE	Define a new attribute
VALUE	Define a named integer value for an attribute
MACRO	Define a macro used to simplify repetitive definitions

Table 58: Dictionary File Keywords (*continued*)

Keyword	Function
OPTIONS	Define options beyond the scope of attribute definitions
#	Ignore this text (comment)

Editing .dct Dictionary Files

The product-specific files shipped with Steel-Belted Radius Carrier reflect specific vendors' implementations of RADIUS clients. Therefore, you do not usually need to modify the .dct dictionary files shipped with Steel-Belted Radius Carrier. However, if your network access server vendor provides information about a new product, a new attribute, or a new value for an attribute, you can add this information to your existing Steel-Belted Radius Carrier configuration by editing dictionary files.

NOTE: If dictionary entries are changed after tunnel, user, or profile attributes have been entered, existing attributes may become no longer editable or orderable. To edit such attributes, delete and re-enter them. This is working as designed.

Before you edit an existing .dct dictionary file or create a new one, you must do the following to integrate your changes into Steel-Belted Radius Carrier:

1. Add a new vendor-product entry to **vendor.ini** so that you can reference the new dictionary while configuring Steel-Belted Radius Carrier.
2. Place your dictionary file in the same directory as the Steel-Belted Radius Carrier daemon.
3. Edit the **dictiona.dcm** file so that it includes your new dictionary file.
4. Stop and restart the server.

Include Records

Records in a .dct dictionary file that begin with the @ character are treated as special include records. The string that follows the @ character identifies the name of a .dct dictionary file whose contents are to be included. For example, the entry **@vendorA.dct** includes all of the entries in the file **vendorA.dct**.

Include records are honored only one level deep. For example, if file **vendorA.dct** includes file **radbase.dct** and **radbase.dct** includes **radacct.dct**, **vendorA.dct** incorporates records in **radbase.dct** but not those in **radacct.dct**.

Primary Dictionary for .dct Files

The primary dictionary **dictiona.dcm** consists of include records that reference vendor-specific dictionaries. The order in which vendor-specific dictionaries are included in the primary dictionary has significance only if two vendor-specific dictionaries contain conflicting definitions for the same attribute or attribute value.

The first definition of an attribute or attribute value takes precedence over later definitions of the same attribute or attribute value. For example, if primary dictionary **dictiona.dcm** consists of the following include records:

```
@vendorA.dct
@vendorB.dct
@vendorC.dct
```

then attributes and attribute values defined in **vendorA.dct** override attributes and attribute values defined in **vendorB.dct** or **vendorC.dct**, and attributes and attribute values in **vendorB.dct** override attributes and values defined in **vendorC.dct**

ATTRIBUTE Records

Attribute records (Table 59 on page 190) conform to the following syntax:

ATTRIBUTE *attrib_name attrib_id syntax_type flags*

Table 59: ATTRIBUTE Record Syntax

Parameter	Function
<i>attrib_name</i>	Name of the attribute (up to 31 characters with no embedded blanks).
<i>attrib_id</i>	Integer in the range 0–255 identifying the attribute's encoded RADIUS identifier.
<i>syntax_type</i>	Syntax type of the attribute.
<i>flags</i>	Defines whether an attribute appears in the check list, the return list (or both), whether it is multi-valued and whether it is orderable.

NOTE: One limitation of standard dictionary files (the **attrib_id** of all the attribute records must be unique) is waived for the primary dictionary file. Multiple vendors can define different attribute names for the same attribute identifier (assuming the attribute identifier is not already used in the base RADIUS specification). Because attributes in the Steel-Belted Radius Carrier database are stored by name (rather than by **attrib_id**), this introduces no ambiguity into the database.

The following example illustrates a typical attribute record:

```
ATTRIBUTE Framed-IP-Netmask 9 ipaddr Cr
```

This attribute record specifies all of the following:

- An attribute named **Framed-IP-Netmask** is supported.
- The attribute's encoded RADIUS identifier is 9.
- The attribute must use the syntax of an IP address.
- Flag characters specify that the attribute can appear multiple times in a check list (C) and at most one time in a return list for User or Profile entries (r) in the Steel-Belted Radius Carrier database.

Attribute Name and Identifier

No two attribute records in a single dictionary file can have the same **attrib_name** or **attrib_id**. If a duplicate **attrib_name** or **attrib_id** is encountered, the later definition of the attribute is ignored in favor of the earlier one.

Syntax Type Identifier

Standard **syntax_type** identifiers are listed in [Table 60 on page 191](#).

Table 60: Syntax Type Identifiers

Syntax Type	Function
hexadecimal	Hexadecimal string.
int1	1-byte (8-bit) unsigned decimal number.
int2	2-byte (16-bit) unsigned decimal number.
int4, integer	4-byte (32-bit) unsigned decimal number.
signed-integer	4-byte (32-bit) signed decimal number. A number with a 1 in the first bit position is interpreted as a negative number.
ipaddr	IP address or IP netmask attribute.
ipaddr-pool	IPv4 address selected from an IP address pool.
string	String attribute (includes null terminator).
stringnz	String attribute (without null terminator).
time	Time attribute (number of seconds since 00:00:00 GMT, 1/1/1970).
constant	null-value attribute (2-octet) containing only type and length
ipv6addr	IPv6 address attribute (per RFC-3162)

Table 60: Syntax Type Identifiers (*continued*)

Syntax Type	Function
ipv6prefix	IPv6 prefix attribute (per RFC-3162)
ipv6interface	IPv6 interface attribute (per RFC-3162)

NOTE: Signed integer support is limited to attributes received in packets and processing relating to those attributes, such as accounting logs, authentication logs, authentication reports, and SQL plug-ins. Web GUI does not support signed integers, and treats signed and unsigned integers as unsigned integers.

Compound Syntax Types

In addition to the standard **syntax_type** identifiers listed in [Table 60 on page 191](#), the dictionary can accommodate compound syntax types for use in defining vendor-specific attributes. Instead of a single **syntax_type** identifier, one or more of the options listed in [Table 61 on page 192](#) can be combined inside square brackets to form a compound syntax type.

Table 61: Compound Syntax Types

Option	Function
vid= <i>nnn</i>	The device manufacturer's SMI Network Management Private Enterprise code (assigned by ISO) in decimal form.
typeN= <i>nnn</i>	Type setting for vendor-specific attribute as defined in the RADIUS specification; <i>N</i> specifies the length of the field (in bytes), <i>nnn</i> specifies the decimal value of the field.
lenN= <i>nnn</i>	Length field for vendor-specific attribute as defined in the RADIUS specification; <i>N</i> specifies the length of the field (in bytes), <i>nnn</i> specifies the decimal value of the field (a plus sign before the value indicates that the length of the data portion is to be added to <i>nnn</i> to obtain the actual length).
fillN= <i>nnn</i>	Fill field setting for non-integer tunneled attributes; <i>N</i> specifies the size of the field to be filled with the value specified by <i>nnn</i> .
data=syntax_type	The actual data to be included in the attribute; the syntax can be any of the standard syntax types.

Table 61: Compound Syntax Types (*continued*)

Option	Function
tag=nnn	<p>Tunnel attributes include a tag field, which may be used to group attributes in the same packet which refer to the same tunnel. Since some vendors' equipment does not support tags, this syntax type is optional and must be present for the attribute to include a tag field.</p> <p>A value of 0 indicates that the field is present but ignored.</p>

An example of a vendor-specific attribute definition follows:

```
ATTRIBUTE vsa-xxx 26 [vid=1234 type1=1 len1=+2 data=string] R
```

Flag Characters

The **flags** setting consists of the concatenation of one or more flag characters from the list in [Table 62 on page 193](#).

Table 62: Flag Characters

Flag Character	Meaning
b or B	<p>Indicates that an attribute may be bundled in a single Vendor-Specific-Attribute for a particular vendor id. It may be included as one of a series of subattributes within a single VSA.</p> <p>NOTE: The bundled option must be specified in all dictionaries that use the same Vendor ID.</p>
c	Attribute can appear once within a user or profile checklist.
C	Attribute can appear multiple times within a user or profile checklist.
r	Attribute can appear once within a user or profile returnlist.
Include-auth-only	Indicates the attribute can be included in an Accept-Response to a WiMAX reauthentication.
R	Attribute can appear multiple times within a user or profile returnlist.
t	Attribute can appear once within a tunnel attribute list.
T	Attribute can appear multiple times within a tunnel attribute list.

Table 62: Flag Characters (*continued*)

Flag Character	Meaning
o or O	Attribute is orderable; the administrator can control the order in which such attributes are stored in the Steel-Belted Radius Carrier database (this flag makes sense only for multi-valued attributes).
salt-encrypt	Causes Steel-Belted Radius Carrier to salt-encrypt the attribute.

VALUE Records

Value records (Table 63 on page 194) are used to define names for specific integer values of previously defined integer attributes. Value records are never required, but are appropriate where specific meaning can be attached to an integer value of an attribute. The value record must conform to the following syntax:

VALUE *attrib_name* *value_name* *integer_value*

Table 63: VALUE Records

Parameter	Function
<i>attrib_name</i>	Name of the attribute (up to 31 characters with no embedded blanks)
<i>value_name</i>	Name of the attribute value (up to 31 characters with no embedded blanks)
<i>integer_value</i>	Integer value associated with the attribute value

No two value records in a .dct dictionary file can have the same **attrib_name** and **value_name** or the same **attrib_name** and **integer_value**. If a duplicate is encountered, the later definition of the attribute value is ignored in favor of the earlier one (the earlier one is considered to be an override).

The following example illustrates the use of the VALUE record to define more user-friendly attribute values for the Framed-Protocol attribute:

ATTRIBUTE	Framed-Protocol	7	integer	Cr
VALUE	Framed-Protocol	PPP	1	
VALUE	Framed-Protocol	SLIP	2	

Using these dictionary records, the administrator need not remember that the integer value 1 means PPP and the integer value 2 means SLIP when used in conjunction with the Framed-Protocol attribute. Instead, the Steel-Belted Radius Carrier Administrator program lets you choose from a list of attribute values including PPP and SLIP.

Macro Records

Macro records ([Table 64 on page 195](#)) are used to streamline the creation of multiple vendor-specific attributes that include many common parameters. A macro record can be used to encapsulate the common parts of the record. The macro record must conform to the following syntax:

MACRO *macro_name*(*macro_vars*) *subst_string*

Table 64: MACRO Records

Parameter	Function
<i>macro_name</i>	Name of the macro
<i>macro_vars</i>	One or more comma-delimited macro variable names
<i>subst_string</i>	String into which macro variables are to be substituted; any sequence of characters conforming to the format %x % for which a macro variable called x has been defined undergo the substitution process

The following example illustrates the use of a macro that simplifies the specification of multiple vendor-specific attributes:

```
MACRO Cisco-VSA(t, s) 26 [vid=9 type1=%t% len1=+2 data=%s%]
ATTRIBUTE Cisco-xxx Cisco-VSA(1, string) R
ATTRIBUTE Cisco-yyy Cisco-VSA(4, int4) C
ATTRIBUTE Cisco-zzz Cisco-VSA(9, ipaddr) r
```

The macro preprocessor built into the Steel-Belted Radius Carrier dictionary processing translates the records in the preceding example to the following records before being processed.

```
ATTRIBUTE Cisco-xxx 26 [vid=9 type1=1 len1=+2 data=string] R
ATTRIBUTE Cisco-yyy 26 [vid=9 type1=4 len1=+2 data=int4] C
ATTRIBUTE Cisco-zzz 26 [vid=9 type1=9 len1=+2 data=ipaddr] r
```

OPTION Records

By default, each vendor-specific attribute is encoded in a single VSA attribute. The format of a VSA attribute is described in [Table 65 on page 195](#).

Table 65: VSA Attribute Format

Bits	Content
0 - 7	Type: contains the value 26

Table 65: VSA Attribute Format (*continued*)

Bits	Content
8 - 16	Length of data in bytes
17 - 47	Vendor ID
48 - on	Vendor data

The OPTION setting can be used to enable the attributes of a particular vendor ID to be bundled within a single VSA.

The OPTION record must conform to the following format:

```
OPTION bundle-vendor-id = vid
```

NOTE: You must set the B flag for attribute bundling to occur. For a particular vendor-specific attribute to be bundled, you must set the OPTION record for the vendor's vendor-ID and set the B (or b) flag for the specific attribute.

The Nortel Rapport dictionary supports this option, for example. If you want to combine Nortel's vendor-specific attributes in a single VSA, provide the entry:

```
OPTION bundle-vendor-id=562
```

This is because 562 is Nortel's Vendor ID, as set in the MACRO record. The Nortel Rapport vendor-specific attributes now are concatenated within the vendor-data portion of a RADIUS VSA attribute (up to 249 octets).

.dic Files

The **.dic** dictionary files are the XML format of the **.dct** dictionary files. The **.dic** dictionary files identify the attributes SBR Carrier expects when receiving Diameter requests from a specific type of device or while sending a Diameter or CoA/DM requests to a specific type of device.

In the **.dic** file, the top level element must be tagged as **<Dictionary>**. The allowable child elements of the **<Dictionary>** element are **<?dict>**, **<vendor>**, and **<attributes>**. The following example illustrates the format of a **.dic** dictionary file.

```
<Dictionary>

<?dict import = "radius.dic" ?>

<vendor id = "WiMAX"      vid ="24757"/>

<attribute id = "WiMAX-Device-Authentication-Ind"      type = "2"      format =
"integer" vendor="WiMAX" sensitive = "false">
<constant name = "Successful"      value = "1"/>
<constant name = "Unsucessful"      value = "2"/>
</attribute>

<attribute id = "WiMAX-GMT-Time-Zone-Offset"      type = "3"      format = "integer"
vendor="WiMAX" sensitive = "false">
</attribute>

<attribute id = "WiMAX-AAA-Session-ID"      type = "4"      format = "octets"
vendor="WiMAX" sensitive = "false">
</attribute>

</Dictionary>
```

.dic File Location

The **.dic** dictionary files must be placed in the same directory as the SBR Carrier daemon. During initialization, SBR Carrier reads the primary file **Dictionaries.xml** in the server directory to get a list of files with an extension of **.dic**.

<?dict> Element

The **<?dict>** element is used to include the records of the specified **.dic** file. The only allowable XML attribute of the **<Dictionary>** element is **import**. Include records are honored only one level deep. For example, if the **vendorA.dic** file includes the **diabase.dic** file and **diabase.dic** includes **diaacct.dic**, then **vendorA.dic** incorporates records in **diabase.dic** but not those in **diaacct.dic**.

<vendor> Element

You can use the **<vendor>** element to streamline the creation of multiple vendor-specific attributes that include many common parameters. You may include multiple **<vendor>** elements for the **<Dictionary>** element. The allowable XML attributes of the **<vendor>** element are:

- **id**—Specifies the name of the vendor

- vid—Specifies the device manufacturer's SMI network management private enterprise code (assigned by ISO) in decimal form

<attribute> Element

The **<attribute>** element is used to define attributes for the **.dic** dictionary file. You may include multiple **<attribute>** elements for the **<Dictionary>** element, each of which provides supplemental details of an attribute in the **.dic** file. [Table 66 on page 198](#) lists the XML attributes allowed in the **<attribute>** element.

Table 66: XML Attributes Allowed in the <attribute> Element

XML Attribute Name	Description
id	Specifies the identifying name of the attribute.
type	Specifies a unique integer type code of the attribute.
format	Specifies the data type of the attribute. This XML attribute can be set to: <ul style="list-style-type: none"> • string—String including null terminator (0–254 octets) • ipaddr6—IPv6 address (16 octets) • ipaddr—IP address (4 octets) • integer—4-byte (32-bit) unsigned decimal number • int8—8-byte (64-bit) unsigned decimal number • time—Number of seconds since 00:00:00 GMT, 1/1/1970 • octets—Raw octets
sensitive	Specifies whether to print the attribute value in the server log file. <ul style="list-style-type: none"> • If set to true, the attribute value is not printed in the server log file. • If set to false, the attribute value is printed in the server log file.
tag	Specifies whether to include a tag field to the attribute. <ul style="list-style-type: none"> • If set to 1, a tag field is included to the attribute. • If set to 0, a tag field is not included to the attribute.
preserve	Specifies whether to format the tag field even if the tag attribute is set to 0. <ul style="list-style-type: none"> • If set to true, the packet builder always formats the tag field even if the tag attribute is set to 0. • If set to false, the packet builder omits the tag field if the tag attribute is set to 0.
vendor	Specifies the vendor ID of the attribute if it is a VSA,

The **<attribute>** element may have **<constant>** child elements. The **<constant>** element defines a named value, which a simple attribute may have, for example:

```
<attribute id = "WiMAX-Device-Authentication-Ind"    type = "2"    format =
"integer" vendor="WiMAX" sensitive = "false">
<constant name = "Successful"    value = "1"/>
<constant name = "Unsucessful"    value = "2"/>
</attribute>
```

Editing .dic Dictionary Files

The product-specific files shipped with SBR Carrier reflect specific vendors' implementations of Diameter clients. Therefore, you do not usually need to modify the .dic dictionary files shipped with SBR Carrier. However, if your network access server vendor provides information about a new product, a new attribute, or a new value for an attribute, you can add this information to your existing SBR Carrier configuration by editing dictionary files.

Before you edit an existing .dic dictionary file or create a new one, you must do the following to integrate your changes into SBR Carrier:

1. Place your dictionary file in the same directory as the SBR Carrier daemon.
2. Edit the **Dictionaries.xml** primary file so that it includes your new dictionary file.
3. Stop and restart the server.

Structured Attributes

Steel-Belted Radius Carrier natively supports structured attributes that contain subattributes. Subattributes are values in a RADIUS packet that are not stored as a RADIUS AVP, or vendor-specific-attribute (VSA), but rather are packed with other subattributes into a RADIUS VSA. In a RADIUS packet, there may be multiple RADIUS VSAs that contain subattributes. The RADIUS VSA, which consists of multiple subattributes, is sometimes referred to as a *structured attribute* because it contains structured data.

Before Release 7.0, Steel-Belted Radius Carrier only interpreted AVPs and not the subattributes contained within an AVP. For some specific cases, plug-ins were available for Steel-Belted Radius Carrier to copy the subattributes from the AVP container and represent them in the RADIUS request as if they had been received as separate AVPs. In the RADIUS response, Steel-Belted Radius Carrier re-assembled the RADIUS AVPs from their contained subattribute values. This process was known as packet *flattening/unflattening*.

The dictionary mechanism of Steel-Belted Radius Carrier has been extended to allow for XML declaration of structured AVP contents. When an AVP with an associated structure definition is received, its internal subattribute values are automatically parsed and become available to any component within Steel-Belted

Radius Carrier that processes RADIUS requests. Similarly, any subattribute values that are populated into the RADIUS response are formatted as part of the structured RADIUS AVP according to the same XML structure definition.

If you previously used the packet flattening/unflattening method, we recommend that you migrate to using subattributes.

NOTE: This guide uses the following terminology when discussing structured attributes and subattributes: Throughout this guide the term attribute can refer to both the structured attribute and subattribute. A distinction has been made only where necessary.

- Attribute—used to represent a standard RADIUS attributes in the packet.
- Structured or parent attribute—used to describe an attribute that contains subattributes, rather than a conventional simple data type such as an integer. This may be a parent attribute, or it may itself be a subattribute.
- Subattribute—refers to the data items within a Structured Attribute. While the subattributes are frequently in TLV format, occasionally they are missing Type, Length, or both.

Structured Attribute Dictionary Definitions

The existing dictionary definitions of Steel-Belted Radius Carrier have been extended to support structured attributes. Existing dictionary definitions are defined in **.dct** files, while structured attribute definitions are defined in **.jdict** files. Like the initialization files, the **.jdict** files are loaded at startup time. These dictionary files identify the structured attributes Steel-Belted Radius Carrier expects to receive in RADIUS requests from a specific type of device, and the structured attributes that Steel-Belted Radius Carrier includes when sending a RADIUS response to a specific type of device.

The **.jdict** definition files reside in the **radiusdir/subattributes** directory. All files in this directory, ending in **.jdict**, are parsed by Steel-Belted Radius Carrier as subattribute definition files. After the definitions are parsed, they are merged into the **.dct** definitions. Each structured attribute definition must correspond to a **.dct** definition.

NOTE: Structured attributes must be defined to be in hexadecimal format in the **.dct** file.

NOTE: *.dic dictionary files do not support structured attribute definitions.

XML Format of Dictionary Files

Subattributes definitions in the .dict files require a specific XML format. Figure 3 on page 201 shows an example file followed by a description of each XML element.

Figure 3: Example of a Structured Attribute Dictionary File

```
<dictionary name="radius.dct">
  <attribute vendor='5535' type='73' name="3GPP2-Diff-Services-Marking" format='group'>
    <sequence type='1' name="Class-Flags">
      <integer name="AF-DSCPs" bitwidth='1' />
      <integer name="EF-DSCPs" bitwidth='1' />
      <integer name="Experimental-Marking" bitwidth='1' />
      <integer bitwidth='5' default='0' /><!-- anonymous padding -->
    </sequence>
    <sequence type='2' name="Max-Class-Marking">
      <integer name="value" bitwidth='6' />
      <integer bitwidth='10' default='0' />
    </sequence>
    <sequence type='3' name="Reverse-Tunnel-Marking">
      <integer name="value" bitwidth='6' />
      <integer bitwidth='10' default='0' />
    </sequence>
  </attribute>
  <attribute vendor='5535' name="3GPP2-Service-Option-Profile" type='74' format='sequence'>
    <integer name="Max-Service-Links" bitwidth='32' />
    <group name="Service-Options">
      <sequence type='1' name="Service-Option" isMultiple="true">
        <integer name="Value" bitwidth='8' />
        <integer name="Max-Count" bitwidth='8' default='0' />
      </sequence>
    </group>
  </attribute>
  <attribute vendor='5535' name="3GPP2-Auth-Flow-Profile-Id" type='131' format='group'>
    <integer type='1' name="Forward-Profile-Id" bitwidth='16' isMultiple="true" />
    <integer type='2' name="Reverse-Profile-Id" bitwidth='16' isMultiple="true" />
    <integer type='3' name="Bidirectional-Profile-Id" bitwidth='16' isMultiple="true" />
  </attribute>
</dictionary>
```

The example dictionary in Figure 3 on page 201 contains three structured attributes: 3GPP2-Diff-Services-Marking, 3GPP2-Service-Option-Profile and 3GPP2-Auth-Flow-Profile-ID.

The following XML elements can be used to define subattributes.

<dictionary> Element

The top level element must be tagged **<dictionary>**, and must specify the name of the .dict file being augmented with the subattribute details in its **name** attribute.

In the example in Figure 3 on page 201, the name of the dictionary file is **radius.dct**.

The only allowable child of the **<dictionary>** element is **<attribute>**.

<attribute> Element

The only allowable child of the **<dictionary>** element is the **<attribute>** element. You may have multiple **<attribute>** elements for each **<dictionary>** element, each of which provides supplemental detail for a structured attribute in the .dict file (which must be of type hexadecimal).

Mandatory XML attributes of **<attribute>** elements are:

- **name:** the identifying name of the attribute, matching that in the **.dct** file. This must be unique across all dictionaries.
- **vendor:** the vendor id for the attribute if it is a VSA, which is the typical case.
- **type:** the integer type code of the attribute (this is unique for a given vendor)
- **format:** either a *sequence* or *group*, specifying the structure of the attribute:
 - A *group* is an unordered collection of subattributes. Attributes defined as children of the group may occur in any order, or not at all.
 - A *sequence* is an ordered collection of subattributes. Each child attribute must occur in the final output, in the defined order.

Optional XML attributes of **<attribute>** elements are:

- **hasContinuationFlag:** indicates that this attribute starts with a continuation octet, where the payload of the attribute may be split over multiple VSAs. This is currently only used in WiMAX attributes.

The allowable children elements of the **<attribute>** element include: **<group>**, **<sequence>**, **<integer>**, **<string>**, **<octets>**, **<IPv4>**, **<IPv6>**.

Subattribute Elements

The allowable children elements for the **<attribute>** element include: **<group>**, **<sequence>**, **<integer>**, **<string>**, **<octets>**, **<IPv4>**, **<IPv6>**.

These children elements each represent a subattribute which may be a child of any structured attribute (top level, or a group or sequence subattribute). **<group>** and **<sequence>** indicate that this is a structured subattribute. The other elements represent simple data types. They all share common configuration and may contain the following attributes:

- **name** (mandatory): indicates the identifying name of the subattribute, which must be unique only within its parent. The subattribute is addressable using the full pathname to this attribute, analogous to files and directories in a file system.
- **type** (optional): if present, indicates the type code at the start of this attribute's encoding. By default, if a type code is specified, it is assumed that the attribute encoding also includes a length octet.
- **hasLength** (optional): used to indicate an unusual attribute encoding with either a type code without a length code, or a length code without a type code. Default: if a **type** is present, the default is true; otherwise, the default is false.
- **isMultiple** (optional): indicates whether this subattribute may appear multiple times. Default false.
- **default** (optional): used to indicate the default value for this attribute. This is of use in a **<sequence>** parent, where values are required for each child. In some cases, a sequence contains anonymous padding bits or octets, this allows the padding value to be specified.

<integer> elements may have the attribute **bitwidth**, which indicates the size of the integer in bits. By default, this is 32, representing a 4 octet integer. It may be any number from 1 through 32. When a

<sequence> attribute has unused bits or octets, define them as anonymous integer elements with **default="0"**.

<string> and **<octet>** elements may have the attribute **length**, to indicate they are of a specified fixed length.

Any of these elements are allowable children of **<group>** or **<sequence>** elements. The other, simple subattribute types, may only have **<constant>** child elements.

<constant>

The **<constant>** element defines, in its **name** and **value** attributes, a named value which a simple attribute may have, for example:

```
<integer name='Tariff-Switch-Support' bitwidth='1' default='0'>
  <constant value='0' name='False' />
  <constant value='1' name='True' />
</integer>
```

Functional Areas That Use Subattributes

Packet Parsing and Formatting

A structured attribute, with a definition in the **.jdict** file, is parsed into subattributes, and their hierarchical structure is visible if the packet is logged in the Steel-Belted Radius Carrier log file. If the subattribute parsing fails, messages are output, and the subattributes are not available; only the parent attribute has its raw binary payload.

When formatting the response packet, created subattributes are processed, but any error encountered prevents the packet from being sent. Steel-Belted Radius Carrier is not able to determine the severity, or consequences of the error, so packet formatting must fail outright. Examples of errors are integer attribute value overflows, or sequence attributes where absent children have no defined default values.

Features that Support the Use of Structured and Subattributes

In general, where standard RADIUS attributes can be defined, structured attributes may also be defined. The following features of Steel-Belted Radius Carrier can use structured attributes:

- Optional WiMAX Mobility module
- Optional SIM authentication module when using Kineto INC
- Accounting in Steel-Belted Radius Carrier core (ASCII) database, as well as SQL and LDAP external databases
- Return lists and check lists (configurable through Web GUI and LCI)
- SQL and LDAP authorization plug-ins (used for optional authentication modules)
- JavaScript

- Filters
- Attribute Maps (used when proxying)

Subattributes in return lists and check lists can be configured using Web GUI or *Steel-Belted Radius Carrier* LDAP Configuration Interface (LCI).

However, most of the features that use subattributes are configured using Steel-Belted Radius Carrier configuration files. Refer to the appropriate chapter in the *SBR Carrier Administration and Configuration Guide* for configuring these features.

Refer to subattributes in configuration files, using the “ ” notation. For example:

“A.b.c”

Where attribute “A” is a group attribute containing a sequence subattribute “b”, which contains a simple attribute “c”.

Subattributes are addressable only by their full *pathname*, which must include all interim group or sequence attributes.

For example, in the dictionary file shown in [Figure 5 on page 209](#), the individual service option value is addressed using the following “.” notation:

3GPP2-Service-Option-Profile.Service-Options.Service-Option.Value

In this example you see the integer “Value” attribute within the **Service-Option** sequence attribute, within the **Service-Options** group attribute, within the structured (parent) **3GPP2-Service-Option-Profile** group attribute.

NOTE: If you are using the optional Session State Register (High Availability) module, you cannot use subattributes in the **sessionTable.ini** file to place subattribute values in the sessions table RadAttr fields. This also applies to standalone server.

Single Subattribute Insertion

You do not need to add every subattribute in a tree. Steel-Belted Radius Carrier handles partially defined trees by automatically supplying subattributes that have a default value specified in the **.jdict** definition files. Additionally, if a subattribute is added that does not have a suitable parent or group defined, Steel-Belted Radius Carrier automatically creates one.

Steel-Belted Radius Carrier can create structured (parent) attributes automatically. When an individual subattribute is to be created (for SQL authentication or accounting, LDAP authentication, JavaScripting, or filtering), the associated structured (parent) attribute is automatically created as needed, either because it does not yet exist, or because to use an existing one violates the **isMultiple=false** status of an attribute in the **.jdict** definition.

Steel-Belted Radius Carrier uses the following algorithm to decide which structured attribute to use, or create, to hold the subattribute:

- Obtain a candidate (parent) structured attribute, if possible: Starting at the top-most attribute type, find the final instance (if any) in the packet. Within that parent attribute, find the final instance of the next level attribute, and so on, until no instance is found or Steel-Belted Radius Carrier has reached the level of the parent attribute and obtained a candidate.
- If a candidate (parent) structured attribute is found, but cannot accept another instance of the insertion subattribute (i.e. an instance of the insertion subattribute already exists, and it is not declared to be multi-valued), discard the candidate parent.
- If no candidate (parent) structured attribute exists, create a new one.
- Add the insertion subattribute to the (parent) structured attribute that was obtained.

Attribute Filtering

Structured attributes may be specified in attribute filters in the same way standard RADIUS attributes are specified. Attribute filters allow you to set up rules for filtering attributes into and out of RADIUS packets. You cannot replace a subattribute with a parent attribute, or vice versa.

See [“Single Subattribute Insertion” on page 204](#) for a description of how structured (parent) attributes are automatically created when a single subattribute is created.

For more information about the attribute filtering capabilities of Steel-Belted Radius Carrier see [“filter.ini File” on page 213](#) in this guide, and see information about setting up filters in the *SBR Carrier Administration and Configuration Guide*.

NOTE: Use the Web GUI to maintain settings in the **filter.ini** file. Do not edit the **filter.ini** file manually.

Proxy Attribute Mapping

Subattributes may be named in the attribute mapping functionality when proxy routing is configured on the Steel-Belted Radius Carrier server. Attribute mapping allows you to map the presence, absence, or specific value of an attribute or subattribute in the incoming packet to a specific realm.

For more information about attribute mapping see the [AuthAttributeMap] and [AcctAttributeMap] sections of [“proxy.ini File” on page 268](#). Also see information about administering proxy RADIUS in the *SBR Carrier Administration and Configuration Guide*.

Structured Attributes in Return Lists and Check Lists

A check list attribute is an item of information that must accompany a request for connection before the connection can be authenticated. A return list attribute is an item of information that Steel-Belted Radius Carrier includes in the Access-Accept message when a connection request is approved.

Structured attributes can be defined as a whole in return lists and check lists. The return lists and check lists can be defined through the LCI or Web GUI.

NOTE: Structured attributes (VSAs with subattributes) defined in return lists are treated as whole units. They are added to the reply message as a whole unit, rather than their subattributes being added individually to any existing response VSAs. In this way they are treated just as unstructured VSAs.

For example:

- Attribute "ParentAttr" is defined as being a multivalue return list attribute, with possible subattributes "ChildAttrA" and "ChildAttrB".
- A response already has a copy of "ParentAttr" with subattribute "ChildAttrA", for example from an authentication process.
- A profile specifies that "ParentAttr" must be added with subattribute "ChildAttrB".

The result is a response with two ParentAttr structured attributes:

ParentAttr

ChildAttrA

ParentAttr

ChildAttrB

The result is *not* be a response with a single ParentAttr:

ParentAttr

ChildAttrA

ChildAttrB

For more information about adding a check list or return list to a user entry or profile, and using the LDAP configuration interface, see the *SBR Carrier Administration and Configuration Guide*.

Using Web GUI to Configure Subattributes in Return Lists and Checklists

Subattributes can be added to return lists and checklists in User and Profile entries. To add a subattribute in a return list or check list using Web GUI, first you add the parent attribute, and then you add the appropriate subattributes using the **Add Child** button.

For more information about adding a check list or return list to a user entry or profile, see the *SBR Carrier Administration and Configuration Guide*.

Using the LCI to Define Subattributes in Return Lists and Check Lists

The LCI (LDAP Configuration Interface) has been extended to facilitate the definition of subattributes in check lists and return lists in XML format. When defining subattributes using the LCI, you must define the entire structured attribute hierarchy in XML format. Individual subattributes cannot be defined in the XML format.

For more information about using the LCI to define structured attributes in check lists and return lists, see the *SBR Carrier Administration and Configuration Guide*.

Javascript

Structured attributes may be specified using the '.' notation in JavaScripts.

See [“Single Subattribute Insertion” on page 204](#) for a description of how parent attributes are automatically created when a single subattribute is created.

Converting Previous Attribute Flatteners with Subattributes

Before Release 7.0, Steel-Belted Radius Carrier used attribute flattening as an interim solution for handling structured attributes, specifically for using Kineto INC and 3GPP2 Rev.A attributes. This flattening capability handled specific attributes by extracting subattributes and putting their values into specially defined structured (parent) attributes in the Steel-Belted Radius Carrier RADIUS dictionary. For example, the attribute: **3GPP2-Service-Option-Profile. Max-Service-Links** was copied to the structured attribute **Funk-3GPP2-Max-Service-Conns**. Conversion to using the new subattribute capability is simply a matter of replacing the attribute name **Funk-3GPP2-Max-Service-Conns** with **3GPP2-Service-Option-Profile. Max-Service-Links** in the configuration.

For more information see [“Attribute Handling Methods” on page 631](#).

Plug-in Attribute Access

Where a plug-in (SQL or LDAP) allows the definition of a named attribute, a subattribute may be named using the '.' pathname notation.

See [“Single Subattribute Insertion” on page 204](#) for a description of how parent attributes are automatically created when a single subattribute is created.

Example of Configuration and Usage of a Structured Attribute

The following example compares the 3GPP2 data definition and the Steel-Belted Radius Carrier .jdict definition for the same attribute.

3GPP2 Data Definition

[Figure 4 on page 208](#) shows the 3GPP2 data structure for the 3GPP2-Service-Option-Profile attribute.

It encodes a total number of connections (links) available, followed by a list of Service Option types and the maximum number of instances for that service type.

Figure 4: 3GPP2 Data Definition of the Service-Option-Profile Attribute

1								2								3								4															
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7								
Type								Length								Vendor-ID																							
Vendor-ID (cont)								Vendor-Type								Vendor-Length																							
Maximum Service Connections/Link Flows Total																																							
Sub-Type (=1)								Length								Service Option <i>n</i>								Maximum number of service instances of Service Option <i>n</i>															

The first two rows (8 octets) are standard RADIUS VSA headers.

- Type is 26, as it is for all RADIUS VSAs
- Vendor-ID is 5535 the defined identifier for 3GPP2 VSAs
- Vendor-Type is 74, the identifier specified by 3GPP2 for the Service Option Profile attribute
- Length and Vendor-Length depend upon the payload.

The remaining rows are the payload of this attribute. They consist of:

- A mandatory 4-octet integer “Maximum Service Connections/Link Flows Total,” followed by
- Any number of instances of the final line of 4 octets, each of which represents one service option and its associated maximum.

SBR Carrier XML Dictionary Definition

Figure 5 on page 209 shows the definition of the same structure shown in Figure 4 on page 208, but encoded in the Steel-Belted Radius Carrier XML dictionary definition. Color coding shows the correspondence between data here and the 3GPP2 definition in Figure 4 on page 208.

Figure 5: XML Dictionary Definition

```

<dictionary name="3GPP2.dct">
  ....
  <attribute vendor='5535' name="3GPP2-Service-Option-Profile" type='74' format='sequence'>
    <integer name="Max-Service-Links" bitwidth='32' />
    <group name="Service-Options">
      <sequence type='1' name="Service-Option" isMultiple="true">
        <integer name="Value" bitwidth='8' />
        <integer name="Max-Count" bitwidth='8' default='0' />
      </sequence>
    </group>
  </attribute>
  ....
</dictionary>

```

- The attribute is defined as a sequence, because the order of the data is fixed; the Max-Service-Links must appear before the instances of the Service-Option block.
- Max-Service-Links is just a simple 4-octet integer
- A group Service-Options is defined to specify the structure where the Service-Option block can occur multiple times. Typically a group contains many children, any of which may occur. In this case, there is only one child type "Service-Option". This group is purely logical; it does not correspond to any binary data at the physical level.
- The Service-Option sequence is defined with "isMultiple=true", to specify that it can occur multiple times in the Service-Options group.
- Because the Service-Option consists of a type (1), length (4), and two single-octet data values (the value of the Service Option, and its count), it is defined as a sequence of two integers with a type value.

Example Data

Figure 6 on page 210 shows the example data for this structure in reference to the 3GPP2 data definition. It contains Max-Service-Links with value 128 (0x80), and three Service Option values, with Values 30, 10, 20, with Max-Count 60, 20, 40 respectively.

Figure 6: 3GPP2 Data Structure

1								2								3								4							
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Type (=26)								Length (=24)								Vendor-ID (=5535)															
Vendor-ID (=5535 cont)																Vendor-Type (=74)								Vendor-Length (=18)							
Maximum service connections/Link Flows total (=128)																															
Sub-Type (=1)								Length (=4)								Service Option n (=30)								Max number of service instances of Service Option n (=60)							
Sub-Type (=1)								Length (=4)								Service Option n (=10)								Max number of service instances of Service Option n (=20)							
Sub-Type (=1)								Length (=4)								Service Option n (=20)								Max number of service instances of Service Option n (=40)							

In the Steel-Belted Radius Carrier log, with log level set to 2, the instance of the attribute example shown in Figure 7 on page 210 is logged as:

Figure 7: Log of 3GPP2-Service-Option-Profile Attribute

```
06/09/2008 11:40:53 (0092) 3GPP2-Service-Option-Profile : sequence value = 00000080 01041e3c 01040a14 01041428
06/09/2008 11:40:53 (0092) Max-Service-Links : int4 value = 128
06/09/2008 11:40:53 (0092) Service-Options : group value = 01041e3c 01040a14 01041428
06/09/2008 11:40:53 (0092) Service-Option : sequence value = 1e3c
06/09/2008 11:40:53 (0092) Value : int1 value = 30
06/09/2008 11:40:53 (0092) Max-Count : int1 value = 60
06/09/2008 11:40:53 (0092) Service-Option : sequence value = 0a14
06/09/2008 11:40:53 (0092) Value : int1 value = 10
06/09/2008 11:40:53 (0092) Max-Count : int1 value = 20
06/09/2008 11:40:53 (0092) Service-Option : sequence value = 1428
06/09/2008 11:40:53 (0092) Value : int1 value = 20
06/09/2008 11:40:53 (0092) Max-Count : int1 value = 40
```

The log shows both the decoded values, and the binary payload encoding, so the same data appears in multiple forms.

```
1a180000 159f4a12 00000080 01041e3c 01040a14 01041428
```

In the packet, the whole VSA, including its RADIUS header, indicated by the first two lines shown in Figure 6 on page 210, appears as:

LCI Encoding

To include the above data as a return list attribute using the Steel-Belted Radius Carrier LCI, the data format is:

```
<attribute name="3GPP2-Service-Option-Profile">
  <attribute name="Max-Service-Links" value='128'/>
  <attribute name="Service-Options">
    <attribute name="Service-Option">
      <attribute name="Value" value='30'/>
      <attribute name="Max-Count" value='60' />
    </attribute>
    <attribute name="Service-Option">
      <attribute name="Value" value='10'/>
      <attribute name="Max-Count" value='20' />
    </attribute>
    <attribute name="Service-Option">
      <attribute name="Value" value='20'/>
      <attribute name="Max-Count" value='40' />
    </attribute>
  </attribute>
</attribute>
```

The XML must be concatenated onto one line for an LCI command. For example, to add the attribute to the return list for the native user “nativeUserName”, the XML must look like this example:

```
dn: radiuslist=reply,radiusname=nativeUserName,radiusclass=Native-User,o=radius
changetype: add
3GPP2-Service-Option-Profile: <attribute name="3GPP2-Service-Option-
Profile"><attribute name="Max-Service-Links" value='128' /><attribute name="Service-
Options"><attribute name="Service-Option"><attribute name="Value"
value='30' /><attribute name="Max-Count" value='60' /></attribute><attribute
name="Service-Option"><attribute name="Value" value='10' /><attribute name="Max-Count"
value='20' /></attribute><attribute name="Service-Option">      <attribute name="Value"
value='20' /><attribute name="Max-Count" value='40'
/></attribute></attribute></attribute>
```

Because, in the **.jdict** definition, the Max-Count attribute has a default of '0', it is allowable to omit it, if the default is the desired value.

NOTE: When using the LCI command line utilities such as `ldapquery`, XML values for structured attributes are displayed encoded in a non-readable format. This encoding is base64 encoding, which can be decoded with many command line or web-based utilities.

Alternatively, the problem can be avoided by using a graphical LDAP client.

classmap.ini File

The **classmap.ini** initialization file specifies what Steel-Belted Radius Carrier does with RADIUS attributes encoded in one or more Class attributes included in accounting requests it receives.

[AttributeName] Section

The [AttributeName] section ([Table 67 on page 212](#)) of **classmap.ini** specifies whether RADIUS information encapsulated in a Class attribute is appended to an accounting request or replaces a current value in an accounting request. If one attribute is replaced by another, the original attribute can be added to the request with a different identifier.

```
[AttributeName]
<add | replace>= Attribute [,Attribute]
```

Table 67: classmap.ini [AttributeName] Syntax

Parameter	Function
<i>AttributeName</i>	Name of the attribute encoded into the Class attribute by the authenticating server.
<add replace>	Specifies whether the attribute value is added to the accounting request (leaving all other values intact) or whether one value replaces another in the accounting request.
<i>Attribute</i>	Specifies the name of the attribute that is added to the accounting request that contains the original value of the attribute identified by AttributeName .
[<i>.Attribute</i>]	<p>Specifies the name of the attribute in the accounting request that contains the value of the attribute displaced when the value of AttributeName replaces the existing Attribute value.</p> <p>Valid only when the replace keyword is used.</p>

NOTE: The RADIUS Class attribute cannot contain IPv6 attributes and structured attributes.

In the following example, the encapsulated **User-Name** attribute replaces the existing **User-Name** in the accounting request.

```
[User-name]
replace = User-Name
```

In the following example, the encapsulated **User-Name** attribute is placed in the accounting request as **User-Name**, and the original value for **User-Name** is added to the request as **Funk-Full-User-Name**.

```
[User-name]
replace = User-Name, Funk-Full-User-Name
```

In the following example, the encapsulated **User-Name** attribute is added to the accounting request as a new attribute, and the original **User-Name** attribute remains unchanged.

```
[User-name]
add = Funk-Full-User-Name
```

Upon receipt of a subsequent accounting request, SBR Carrier decapsulates and forwards the upstream server's Class attribute. This action can result in two Class attributes being present in the proxied accounting request. In the following example, the encapsulated Class attribute replaces the existing Class attribute in the accounting request to prevent the Class attribute for SBR Carrier from being forwarded.

```
[Class]
replace = Class
```

filter.ini File

The **filter.ini** file lets you set up rules for filtering attributes and structured attributes into and out of RADIUS packets.

NOTE: Use the Web GUI to maintain settings in the **filter.ini** file. Do not edit the **filter.ini** file manually.

Filter Rules

Each filter in the **filter.ini** file consists of the filter name in square brackets ([**name**]) followed by the rules for that filter.

Each rule takes one of the following three forms:

<i>keyword attribute value</i>
<i>keyword attribute</i>
<i>keyword</i>

Table 68 on page 214 lists valid syntax combinations.

Table 68: Filter Syntax

filter.ini Rule Syntax	Function
ALLOW	This keyword by itself specifies that all attributes, regardless of value, are to be allowed in the packet.
ALLOW <i>attribute</i>	This rule specifies that this attribute is allowed in the packet, regardless of its value.
ALLOW <i>attribute value</i>	The rule lists a specific attribute/value pair to allow in the packet.
ALLOW_UNKNOWN <i>vendorID</i>	<p>This rule specifies that all attributes, regardless of whether they are included in the dictionary of the sending NAS, are included when proxying the message to the target (outbound filters) or before returning the proxy response (inbound filters).</p> <p>Optionally, a Vendor Id may accompany the directive. When used with a global EXCLUDE_UNKNOWN, this rule overrides the exclusion of attributes from the specified vendor ID.</p>
EXCLUDE	<p>The keyword by itself specifies that all attributes, regardless of value, are to be excluded from the packet.</p> <p>EXCLUDE is the default action for a filter.</p>
EXCLUDE <i>attribute</i>	The rule specifies that this attribute is excluded from the packet, regardless of its value.
EXCLUDE <i>attribute value</i>	The rule specifies an attribute/value pair to exclude from the packet.

Table 68: Filter Syntax (*continued*)

filter.ini Rule Syntax	Function
EXCLUDE_UNKNOWN <i>vendorID</i>	<p>This rule specifies that all attributes that are not included in the dictionary of the sending NAS are deleted before proxying the message to the target (outbound filters) or before returning the proxy response (inbound filters).</p> <p>Optionally, a Vendor Id may accompany the directive. If included, only attributes from the specified vendor are excluded.</p>
ADD <i>attribute value</i>	The rule lists a specific attribute/value pair to add to the packet. The attribute is added after all other rules are processed.
REPLACE <i>attr1</i> WITH <i>attr2</i>	The rule specifies that any occurrence of attr1 is replaced by attr2 , which retains attr1 's value.
REPLACE <i>attr1</i> WITH <i>attr2</i> v2	The rule specifies that any occurrence of attr1 (regardless of value) is replaced by attr2 whose value is set to v2 .
REPLACE <i>attr1</i> v1 WITH <i>attr2</i>	The rule specifies that any occurrence of attr1 whose value is v1 is replaced by attr2 (which keeps value v1).
REPLACE <i>attr1</i> v1 WITH <i>attr2</i> v2	The rule specifies that any occurrence of attr1 whose value is v1 is replaced by attr2 having a value v2 .

NOTE: You cannot replace a subattribute with a parent attribute, or vice versa.

An attribute is **ADDED** to a packet only if it is legal to do so. Some attributes can appear only once in a RADIUS packet; others can appear multiple times. If an attribute that is the subject of an **ADD** rule is already present in the packet (after processing **ALLOW** and **EXCLUDE** rules) and the attribute can only appear once, the ADD rule is not processed and the second instance of the attribute is not added.

The Steel-Belted Radius Carrier dictionary file **radius.dct** provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute filter rules.

NOTE: Filter rules provide you with tremendous flexibility. However, Steel-Belted Radius Carrier does not prevent you from creating an invalid RADIUS packet. Some attributes are not appropriate for certain types of requests. For example, adding a pooled Framed-Ip-Address attribute to an accounting request can cause a loss of available IP addresses.

Order of Filter Rules

The order of rules is important. General default rules that take no parameters, such as **ALLOW** (allow all attributes unless otherwise specified) or **EXCLUDE** (exclude all attributes unless otherwise specified) must appear as the first rule in the filter. Later rules supersede earlier rules; the last applicable rule “wins.” **ADD** and **REPLACE** rules are applied after the **ALLOW** and **EXCLUDE** rules.

More specific rules with more parameters (**ADD attribute value**) act as exceptions to less specific rules with fewer parameters (**ALLOW attribute**, **EXCLUDE**). For example, you might want to **ALLOW** a certain attribute and **EXCLUDE** one or more specific values for that attribute. Or you might **EXCLUDE** all attributes, **ALLOW** specific attributes, and **ADD** specific attribute/value pairs.

You can use two basic approaches to designing a filter:

- Start the rule list with a default **EXCLUDE** rule (no parameters) and add **ALLOW** rules for any attributes or attribute/value pairs that you want to insert into the packet. **ADD** and **REPLACE** rules may be used.
- Start the rule list with a default **ALLOW** rule (no parameters) and add **EXCLUDE** rules for any attributes or attribute/value pairs that you want to remove from the packet. **ADD** and **REPLACE** rules may be used.

The default action for **filter.ini** is **EXCLUDE**. If a filter does not contain any rules, the filter removes all attributes from a packet when the filter is applied.

Examples

Here are a few examples of how to use the filter rules.

Allow all attributes except any undefined attributes (attributes with no **.dct** definition):

```
[exclude_all_unknown]
ALLOW
EXCLUDE_UNKNOWN
```

Allow all attributes except undefined attributes for vendor 12345:


```
[exclude_specific_unknown]
ALLOW
EXCLUDE_UNKNOWN 12345
```

Allow all known attributes, disallow undefined attributes, but allow undefined attributes for vendor 12345:

```
[exclude_all_unknown_except_specific]
ALLOW
EXCLUDE_UNKNOWN
ALLOW_UNKNOWN 12345
```

Values in Filter Rules

The value of an attribute is interpreted based on the type of the attribute in its attribute dictionary. [Table 69 on page 217](#) lists the meaning of each attribute type.

Table 69: Filter Rule Values

Attribute Type	Function
hexadecimal	A hexadecimal value is specified as a string. Special characters may be included using escape codes.
int1, int4, integer	1- or 4-byte unsigned decimal number (integer is equivalent to int4). NOTE: The Steel-Belted Radius Carrier dictionary file radius.dct provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute filter rules.
ipaddr, ipaddr-pool	An IP address in dotted notation; for example: EXCLUDE NAS-IP-Address 127.0.0.1

Table 69: Filter Rule Values (continued)

Attribute Type	Function
string	<p>String attribute (includes null terminator). A string is specified as text. The text may be enclosed in double-quotes ("). The text is interpreted as a regular expression. Backslash (\) is the escape character. Escape codes are interpreted as:</p> <p>Code Meaning</p> <p>\a 7</p> <p>\b 8</p> <p>\f 12</p> <p>\n 10</p> <p>\r 13</p> <p>\t 9</p> <p>\v 11</p> <p>\nnnnnn is a decimal value between 0 and 255</p> <p>\xnnnn is a hexadecimal value between 00 and FF</p> <p>\c c is a single character, interpreted literally</p> <p>Literal backslashes (\) within a string and double-quotes (") within quoted strings are prefixed with an escape character. For example:</p> <p>ADD Reply-Message Session limit is one hour</p> <p>ADD Reply-Message "Session limit is one hour"</p> <p>ADD Reply-Message "Your username is \"George\""</p>

Table 69: Filter Rule Values (*continued*)

Attribute Type	Function
time	<p>A time value is specified with a string indicating date and time:</p> <p><code>yyyy/mm/dd hh:mm:ss</code></p> <p>The date portion is mandatory; the time portion may be specified to whatever degree of precision is required, or may be omitted entirely. For example:</p> <p><code>2006/4/3 14:00:00</code></p> <p>and</p> <p><code>2006/4/3 14</code></p> <p>both refer to April 3, 2006 at 2:00 p.m.</p> <p>For example:</p> <p>ADD Ascend-PW-Expiration 2006/4/3</p>

Referencing Attribute Filters

Steel-Belted Radius Carrier attribute filtering provides flexibility in packet processing. You can use the same filter for all packets in all realms. You can apply filtering to some realms, and not others. (To disable filtering for a realm, omit filtering parameters from the ***.pro**, ***.dir**, **peapauth.aut**, or **ttlsauth.aut** file.) Filtering is often used only for packets that are routed out to realms (the **FilterOut** parameter).

To reference the filtering rules defined in the **filter.ini** file in proxy or directed realm configurations, you must use the **FilterOut** and **FilterIn** parameters in the [Auth] and [Acct] sections of a RADIUS realm configuration file.

The full syntax used is:

```
[Auth]
FilterIn=name1
FilterOut=name2
```

```
[Auth]
FilterIn=name3
FilterOut=name4
```

where **name1**, **name2**, and so forth provide the names of filters, sections in the filter.ini file called [**name1**], [**name2**], and so forth. The **name** values in this syntax are completely independent of each other. They may be all the same, all different, or some combination of same and different.

When using the FilterIn and FilterOut parameters in the [Auth] and [Acct] sections, be sure to use the filter name without the square brackets ("name", not "[name]").

NOTE: If a [name] section is not found in the **filter.ini** file, it is equivalent to assigning a filter that EXCLUDEs all attributes. In other words, assigning a filter name that cannot be found causes the final packet to be emptied of all attributes.

NOTE: Do not allocate IP addresses from Steel-Belted Radius Carrier IP address pools in accounting filters. These addresses are allocated but never released.

sample.rr File

Attribute value pools allow Steel-Belted Radius Carrier to assign and return attribute sets dynamically when an Authorization Request is processed. This functionality is supported by the use of a vendor-specific attribute (VSA) called **FunkRound-Robin-Group**. The value for this attribute is a string, and is set to the name of a .rr suffix file that defines an attribute value pool. This value can therefore be set for a user or profile by using the Web GUI or LDAP Configuration Interface (LCI) or by any other return list mechanism (such as database retrieval).

Attribute value pooling allows for a dynamic allocation of attribute values sets, so that attributes needed to configure changeable and complex situations do not have to be assigned in static profiles. This functionality is supported by the use of a vendor-specific attribute called **Funk-Round-Robin-Group**. The value for this attribute is a string, and is set to the name of a .rr suffix file that defines an attribute value pool.

A .rr file is defined as:

```
[Sets]
SetName1 = Weight1
SetName2 = Weight2
...
[ SetName1]
AttributeName1.1 = AttributeValue1.1
```

```
Attribute1.2 = AttributeValue1.2
...
```

Steel-Belted Radius Carrier maintains round-robin statistics for each attribute value pool so that weight calculations can be performed properly. When a user who belongs to a profile that has been assigned to a particular attribute value pool logs in, the round-robin values are incremented to determine which Attribute Value set is assigned to the user. This attribute set is added to the return list of the Access-Accept.

Attribute value pooling can be used in several ways. For example, the Acme Company wants off-site employees to be able to establish tunnels to the company network. The Acme Company maintains three tunnel connection endpoints to which end users can create VPNs into the corporate network, each of these with different capacities. The company needs to define an attribute value pool of three attribute sets, each describing how to establish a tunnel with one of these connection points. These attribute sets are weighted according to the capacity of the three connection points. [Figure 8 on page 221](#) illustrates a sample **acme.rr** file.

Figure 8: Sample *.rr file (acme.rr)

```
;acme.rr
[Sets]
VPN1=20
VPN2=12
VPN3=7

[VPN1]
Tunnel-Server-Endpoint = 8.4.2.1
Tunnel-Password = GoodGuess

[VPN2]
Tunnel-Server-Endpoint = 8.4.2.2
Tunnel-Password = BestGuess

[VPN3]
Tunnel-Server-Endpoint = 8.4.2.4
Tunnel-Password = OurSecret
```

To make this attribute value pool visible, the Acme Company defines a **FunkRound-Robin-Group** VSA and assign it to the users (or the profile assigned to these users) and make the value of the VSA point to the **acme.rr** file shown in [Figure 8 on page 221](#).

```
Funk-Round-Robin-Group = acme.rr
```

Refer to the *SBR Carrier Administration and Configuration Guide* for more information about using attribute value pooling.

spi.ini File

The **spi.ini** initialization file defines encryption keys and identifies the servers from which Steel-Belted Radius Carrier processes encrypted Class attributes in accounting requests. The **spi.ini** file allows one Steel-Belted Radius Carrier server to decode accounting requests for sessions that were authenticated on a different Steel-Belted Radius Carrier server. Class attributes received from servers not specified in **spi.ini** are ignored.

NOTE: If you are using the optional SSR (high availability) module and distributing authentication and accounting requests between different SBR Carrier servers sharing the same SSR cluster, you must configure **spi.ini** file.

All Steel-Belted Radius Carrier servers that may receive authentication and accounting requests from a common network access server must be configured with similar **spi.ini** files, which must list the IP addresses of all the servers in that cluster. This allows one server to authenticate a user and generate an encrypted Class attribute that can be decrypted and processed by any other server in the cluster.

[Keys] Section

The [Keys] section ([Table 70 on page 223](#)) of **spi.ini** specifies the list of encryption keys used to encode subattributes encapsulated within Class attributes.

```
[Keys]
CurrentKey = n
1 = value
2 = value
.
.
.
```

Table 70: spi.ini [Keys] Syntax

Parameter	Function
CurrentKey	<p>Specifies the encryption key that is currently active, where <i>n</i> is 0 or the number of a key listed in the [Keys] section:</p> <ul style="list-style-type: none"> • 0—Generate and use a unique random key to encrypt Class attributes. Used only when the Steel-Belted Radius Carrier server does not exchange encrypted Class attributes with other servers. • <i>n</i>—Use the specified key to encrypt Class attributes. <p>Default value is 0.</p>
<i>n</i> = value	Specifies the number and value of the encryption key.

In the following example, the Steel-Belted Radius Carrier server generates a unique random key to encrypt Class attributes.

```
[Keys]
CurrentKey = 0
```

In the following example, the second key (**swordfish**) is currently active and used to encrypt Class attributes. The other keys in this section can be used to decrypt Class attributes received from other servers in the same cluster.

```
[Keys]
CurrentKey = 2
1 = firstkey
2 = swordfish
3 = mypassword
```

[Hosts] Section

The [Hosts] section of **spi.ini** identifies the IP address of servers from which received Class attributes are parsed for encapsulated/encrypted subattributes. Class attributes from servers not identified in the [Hosts] section of **spi.ini** are passed without special processing.

The information in the [Hosts] section is used to compute the server's identifier, which is included in the Class attribute. If one of a host's interfaces is included in the [Hosts] section, that interface is used to compute the server identifier. If more than one interface for a host is listed, the IP address of the last interface listed is used. If no matching address is found, the host's primary IP address is used. Addresses

not corresponding to a host interface are used to configure the collection of other servers whose Class attributes are accepted.

In the following example, three servers are identified as belonging to a cluster.

```
[Hosts]
192.168.15.21
192.168.23.121
192.168.23.205
```

vendor.ini File

The **vendor.ini** initialization file contains information that allows Steel-Belted Radius Carrier to work with the products of other vendors.

[Vendor-Product Identification] Section

The [Vendor-Product Identification] section ([Table 71 on page 224](#)) of **vendor.ini** identifies and provides information about the network access devices that can be used with Steel-Belted Radius Carrier.

Table 71: vendor.ini [Vendor-Product Identification] Syntax

Parameter	Function
vendor-product	Specifies the name of the product. A product name must be unique, cannot include blanks and must consist of 127 or fewer characters. These product names are used only in the Make or Model list in the RADIUS Clients List page. This list is used when adding a new RADIUS client or when selecting a vendor-specific attribute.
dictionary	Specifies the dictionary file to use for this product. The dictionary file must be located in the same directory as the Steel-Belted Radius Carrier daemon or service. You do not need to specify an extension on the dictionary name; Steel-Belted Radius Carrier automatically attaches an extension of .dct to the dictionary names listed in this parameter.
call-filter-attribute	Specifies the attribute used for call filter functions. Used only by Ascend/Lucent network access devices.

Table 71: vendor.ini [Vendor-Product Identification] Syntax (*continued*)

Parameter	Function
challenge-response-attribute	<p>Specifies the attribute number in which a network access server sends responses to challenge sequences.</p> <p>If not specified, the default behavior is to expect responses to be encoded in the User-Password attribute.</p>
Convert-Calling-Station-Id	<ul style="list-style-type: none"> • If set to yes, the Calling-Station-Id is interpreted as a hex string. <p>NOTE: This parameter overrides the ConvertCallingStationId parameter in radius.ini file.</p> <ul style="list-style-type: none"> • If set to no, the Calling-Station-Id is interpreted as ASCII.
data-filter-attribute	<p>Specifies the attribute used for data filter functionality. Used only by Ascend/Lucent network access devices.</p>
discard-after	<p>Used for inbound proxy RADIUS servers that send username information in a decorated format. For example, if a proxy RADIUS server sends usernames of the form username@company, then specifying @ results in the @ delimiter character and all text after the @ delimiter character being discarded for authentication purposes; the string username is used.</p>
discard-before	<p>Used for inbound proxy RADIUS servers that send username information in a decorated format. For example, if a proxy RADIUS server sends usernames of the form company\$username, then specifying \$ results in the \$ delimiter character and all text before the \$ delimiter character being discarded for authentication purposes; the string username is used.</p>
help-id	<p>Help context for the vendor's product in the vendor information help file.</p>
ignore-acct-ss	<p>If set to Yes, the digital signature of accounting packets based on the shared secret is ignored. This accommodates devices that do not properly sign accounting packages.</p> <p>Default value is No.</p>

Table 71: vendor.ini [Vendor-Product Identification] Syntax (*continued*)

Parameter	Function
ignore-ports	<p>Determines whether Steel-Belted Radius Carrier may infer that one user has logged off if the port that was assigned to that user is now being used by another user.</p> <ul style="list-style-type: none"> • If set to No, an inference is made and the previous user is removed from the Active Users list. • If set to Yes, no inference is made and both users are deemed active. <p>Default value is No.</p>
max-eap-fragment	<p>Specifies a maximum EAP fragment length on a make/model basis. The maximum EAP fragment length emitted by TLS or TTLS is the lesser of the maximum specified in their .eap/.aut files and this setting.</p> <p>Default value is 1020. This may be inefficient, however, as the fragment length must be set to a number low enough to work with all of a customer's Access Points.</p>
port-number-usage	<ul style="list-style-type: none"> • If set to per-port-type, entries in the Active List containing duplicate port numbers and port types are deleted. • If set to unique, entries in the Active List containing duplicate port numbers are deleted; port type information is ignored. <p>Default value is per-port-type.</p>
product-scan-acct	<p>Specifies the name of the section in the vendor.ini file that contains rules for dynamically determining the product associated with an accounting request by the contents of the request packet.</p>
product-scan-auth	<p>Specifies the name of the section in the vendor.ini file that contains rules for dynamically determining the product associated with an authentication request by the contents of the request packet.</p>
send-class-attribute	<p>If set to No, the Class attribute is not sent to the client on Access-Accept. (This feature is designed to accommodate devices that do not handle the Class attribute properly.)</p> <p>Default value is Yes.</p>

Table 71: vendor.ini [Vendor-Product Identification] Syntax (*continued*)

Parameter	Function
send-session-timeout-on-challenge	<ul style="list-style-type: none"> • If set to Yes, the Session-Timeout attribute is sent to the client on Access-Challenge responses that include EAP messages. This attribute advises a network access server how long to wait for a user response to the challenge. • If set to No, the Session-Timeout attribute is not sent to the client on Access-Challenge responses that include EAP messages. <p>Default value is Yes.</p>
SuppressSessionAttr	<p>Specifies an attribute name. If an accounting request contains the attribute specified in this parameter for a particular vendor, sessions will not be created in CST. This attribute must be valid for the dictionary used by the Make/Model (vendor-product) entry.</p> <p>NOTE: This setting considers only the specified attribute and does not consider the value of the attribute.</p>
send-extra-attributes-on-auth-only	<ul style="list-style-type: none"> • If set to Yes, attributes in the WiMAX dictionary are included in responses to auth-only (reauthentication) requests. • If set to No, attributes in the WiMAX dictionary are not included in responses to auth-only (reauthentication) requests.
WiMAX-Revision-Number	<p>The WiMAX specification revision number. The WiMAX mobility module changes behavior based on the revision number. Valid values are 1.0 and 1.2.</p> <ul style="list-style-type: none"> • 1.2, include any revision greater than 1.0 up to and including 1.2. <p>The default is 1.2.</p>

Product-Scan Settings

After you define a Vendor-Product entry ([Table 72 on page 228](#)) in **vendor.ini**, the name of this entry can be selected in the RADIUS Clients dialog as a possible value for the **Make/model** field. The **ProductScanAuth** and **ProductScanAcct** settings can be used within a VendorProduct entry to permit dynamic make/model selection to occur. These settings enable Steel-Belted Radius Carrier to examine the incoming packet to determine the make/model of the network access server that originated the packet.

A dynamic Vendor-Product entry might appear as:

```

Vendor-Product = DeviceNameInRASClientsList
Product-Scan-Auth = MakeModelSelect
Product-Scan-Acct = MakeModelForAccounting
[MakeModelForAuthentication]
Product = String
Product = String
.
.
.
Product =
[MakeModelForAccounting]
Product = String
Product = String
.
.
.
Product =

```

Table 72: vendor.ini Product-Scan Syntax

Parameter	Function
Vendor-Product	Creates a label that appears as a selection in the Make/Model list in the RADIUS Clients List page in Web GUI.
Product-Scan-Auth= <i>name</i>	Applies only to authentication servers. <i>name</i> references a section heading that appears elsewhere in vendor.ini .
Product-Scan-Acct= <i>name</i>	Applies only to accounting servers. <i>name</i> references a section heading that appears elsewhere in vendor.ini .
[<i>name</i>]	Provides rules that govern dynamic make/model selection. These rules apply on authentication requests if the value <i>name</i> is assigned to Product-Scan-Auth; they apply on accounting requests if the value <i>name</i> is assigned to Product-Scan-Acct.

Table 72: vendor.ini Product-Scan Syntax (continued)

Parameter	Function
Product= <i>String</i>	Product is a product name. String is a regular expression to match against attributes in the packet. Character by character, Product must match a Vendor-Product value defined elsewhere in the vendor.ini file.
Product=	<p>The default vendor.ini provided with Steel-Belted Radius Carrier includes a number of Vendor-Product values from which you may choose. Each value corresponds to a vendor-specific RADIUS attribute dictionary.</p> <p>The list of product names and strings is tried in order. If the packet does not come from the first device, the next is tried, and so on until the last entry in the list is tried.</p> <p>You can set up a default at the end of the list by making sure the last Product entry in the list has no String assigned. If no match is found earlier in the list, Steel-Belted Radius Carrier assumes that the packet comes from the type of device specified in the final entry.</p>

The following example is appropriate in a configuration where the NAS devices are primarily Ascend devices:

```
Product-Scan-Auth = Bigco Special Scan

[Bigco Special Scan]
Ascend MAX Family = \x2c?
Nortel Versalar Remote Access Concentrator =
\x1a?\x00\x00\x06\x30
US Robotics NETServer = \x1a?\x00\x00\x01\xad
Ascend MAX Family =
```

The preceding example sets up dynamic make/model selection for authentication and states that the identity of the client device is determined by seeking matches in this order:

1. Is the attribute with identifier number 0x2c (**Acct-Session-Id**), with a value of any length (indicated by the question mark character), found in the incoming authentication packet? If so, the originating network access server is a member of the Ascend MAX Family; use that vendor-specific dictionary.
2. Is the vendor-specific attribute with identifier number 0x1a (**Vendor-Id**), with a value of any length (indicated by the question mark character), present in the packet? If so, does it have the value 1584

(0x630) which indicates a Nortel Networks Versalar RAC? If so, use that vendor-specific dictionary (provided with Steel-Belted Radius Carrier).

3. Is the **Vendor-Id** attribute present, with any length, and if so, does it have the value 429 (0x1ad) which indicates a US Robotics NETServer? If so, use that vendor-specific dictionary (provided with Steel-Belted Radius Carrier).
4. If no match can be found using the rules specified in this section, then use the vendor-specific dictionary for the Ascend MAX Family.

NOTE: Include a default entry in this section. When there is no default, if an Access-Request is received with no vendor-specific attributes of any kind, the user may be rejected due to invalid resources, as the RADIUS server cannot associate a valid dictionary with the request. Using the example:
- Standard RADIUS - =
 as the last line in this section is a safe configuration.

Adding NAS Location Information to Access-Request Messages

Steel-Belted Radius Carrier core provides a special attribute handling feature which allows you to add NAS location information to proxied Access-Request messages.

This section describes how this feature works, and the files which must be configured to enable this feature.

NOTE: The terms *network access device* (NAD), *remote access server* (RAS), and *network access server* (NAS) are interchangeable. This guide primarily uses the term NAS.

Service providers might require the location of the mobile device that is requesting access. For example, a service provider might offer weather reports or advertising based on the location of the mobile device.

You can configure an Access-Request to include the location of the NAS through which the proxied request was processed. The NAS is geographically near the mobile device. The location of the NAS closely approximates the location of the mobile device.

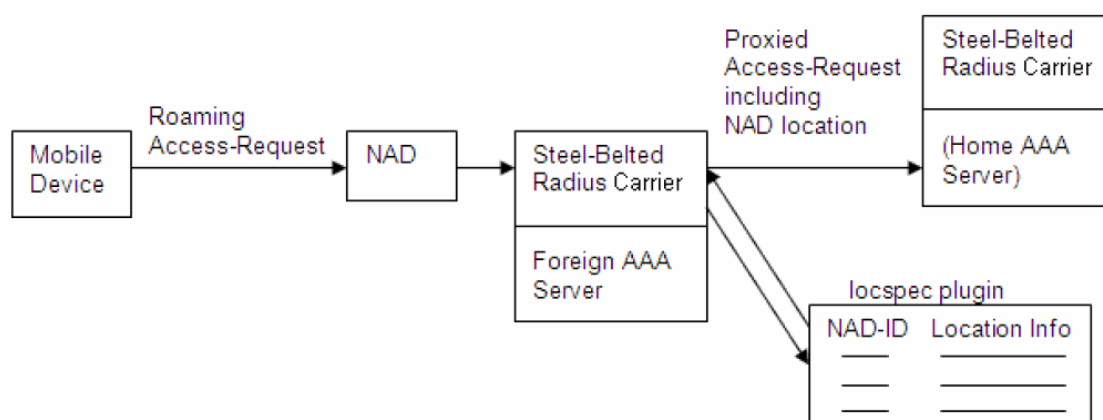
When a mobile device is outside the area of its provider, it roams by sending the request to a local foreign AAA (FAAA) server that is owned by another provider. The FAAA server proxies (forwards) the request to the appropriate home AAA (HAAA) server for the user.

For proxied requests, Steel-Belted Radius Carrier can perform a lookup to find a NAS location based on an attribute (usually NAS-Identifier or NAS-IP-Address). The attribute that is used to look up the NAS location is user-configurable as the **AttributeToIdentifyNAS** in the **locspec.ctrl** file.

Figure 9 on page 231 shows that Steel-Belted Radius Carrier queries the **locspec** plug-in to find the value of the attribute that identifies the NAS location. The NAS location is then added to the Access-Request that is sent to the service provider's home AAA server.

NOTE: Each Steel-Belted Radius Carrier server that might be the target of a proxy request must be set up as a proxy target. Set up proxy targets with the Web GUI. For more information about setting up proxy RADIUS, see the *SBR Carrier Administration and Configuration Guide*.

Figure 9: Addition of NAS Location to Access-Request



Use the following procedure to add NAS location attribute information to Access-Request messages.

Location-Specific Configuration Files

The following files and file sections require configuration to add location attributes to the Access-Request. Figure 11 on page 238 provides an example showing the relationship between all the configuration files.

```

locspec.ctrl file
[Bootstrap] section
[Settings] section
[NAS-LIST] section
[NAS Identifier] section
proxy.ini file
[Realms] section
  
```

```
realm.pro file
[Auth-Outbound-To-Proxy] section
[Acct-Outbound-To-Proxy] section
```

locspec.ctl File

The **locspec.ctl** file calls the **LOCSPEC** control point plug-in, which enables the addition of location-specific information to an Access-Request.

[Table 73 on page 232](#) defines the fields needed in the [Bootstrap] section for adding location-specific attributes to an Access-Request.

Table 73: locspec.ctl [Bootstrap] Fields

Field	Description
LibraryName	Specifies the name of the executable binary. Set to locspec.so
Enable	Set to 1 to enable this file. Set to 0 to disable this file. Set to 1.
InitializationString	Specifies the name of the control point plug-in file that activates location-specific information. Set to LOCSPEC .

Example

```
[Bootstrap]
LibraryName=locspec.so
Enable=1
InitializationString=LOCSPEC
```

[Table 74 on page 233](#) defines the fields needed in the [Settings] section for adding location-specific attributes to an Access-Request.

Table 74: locspec.ctrl [Settings] Fields

Field	Description
AttributeToIdentifyNAS	<p>Attribute to be used to identify the NAS. Typically, this value is set to one of the following:</p> <p>NAS-Identifier</p> <p>NAS-IP-Address</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
ConfigLog	<p>Method for capturing log information.</p> <ul style="list-style-type: none"> • None= Configuration information is not captured. • ConsoleAndLog= Log information is sent to both the console and the log. • Console= Log information is sent to the console only. • Log= Log information is sent to the log file only. <p>Default is ConsoleAndLog.</p>

OperatorNameAttribute = TeliaSonera-Operator-Name
 VisitedOperatorIdAttribute = TeliaSonera-Visited-Operator-ID
 LocationInformation = TeliaSonera-Location-Information
 LocationNameAttribute = TeliaSonera-Location-Name

Example

Table 75 on page 234 defines the fields needed in the [NAS-LIST] section for adding location-specific attributes to an Access-Request.

Table 75: locspec.ctl [NAS-LIST] Fields for Configuration of Location-specific Attributes

Field	Description
NAS designator	<p>List of NAS devices.</p> <p>The [NAS-List] section includes a list of NAS devices that are configured to transmit their location. The attribute used to identify a NAS in this list is configured in the AttributeToIdentifyNAS field within the [Settings] section of the locspec.ctl field. Typically, the NAS-Identifier attribute or the NAS-IP-Address attribute is used to identify a NAS.</p> <p>For example, if AttributeToIdentifyNAS=NAS-IP-Address, then all the NAS devices in this list are identified by their IP Address. If AttributeToIdentifyNAS=NAS-Identifier, then all the NAS devices in this list are identified by their NAS Identifier (name).</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Example

```
[NAS-LIST]
NAS_1
NAS_2
```

For each NAS device listed in the [NAS-LIST] section, there must be a separate section in **locspec.ctl** providing location information about the NAS.

[Table 76 on page 234](#) defines the fields needed in the [NAS Identifier] section that provide location-specific information to an Access-Request. The Access-Request can contain all of these four attributes or a subset.

Table 76: Location Attributes for the NAS Device

Field	Description
GSM-Operator-Name	<p>GSM-Operator-Name = <i>prefix:value</i></p> <p>where</p> <p><i>prefix</i> = either GSM or REALM</p> <p>code =If <i>prefix</i>=GSM, code = any GSMA assigned TADIG code in capital ASCII letters available at http://www.gsmworld.org; If <i>prefix</i>=REALM, code = or any valid domain name string</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 76: Location Attributes for the NAS Device (continued)

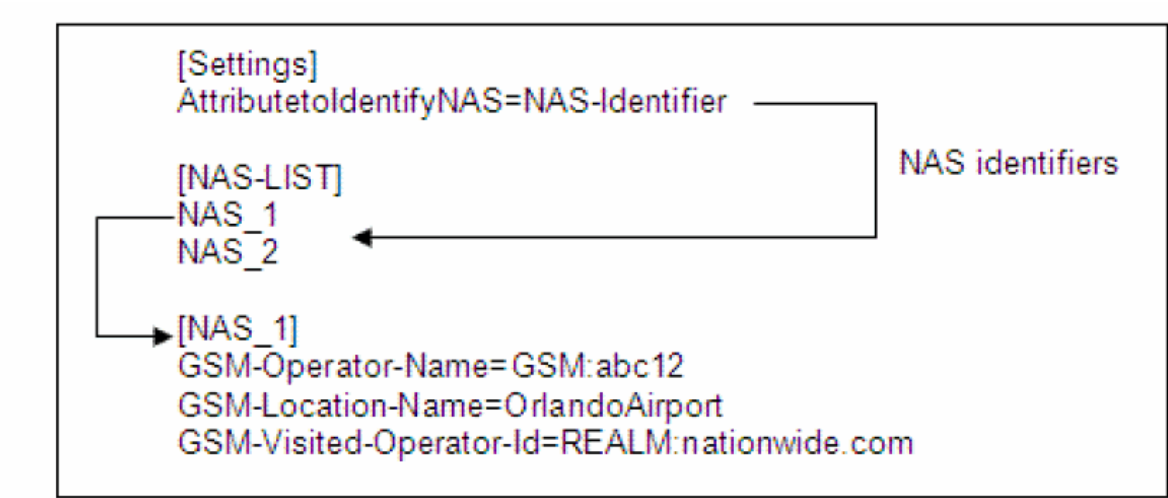
Field	Description
GSM-Location-Information	<p>GSM-Location-Information = country= <i>code</i> [; <i>civic-label</i>=<i>value</i>]</p> <p>where <i>code</i> = ISO 3166 2-letter country code. <i>civic-label</i> = A1, A2, A3, A4, A5, A6, PRD, POD, STS, HNO, HNS, LMK, LOC, NAM, ZIP, PCN, or an integer as defined in draft-ietf-geopriv-dhcp-civil-09.txt available at http://www.ietf.org.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
GSM-Visited-Operator-Id	<p>GSM-Visited-Operator-Id = <i>prefix</i>:<i>value</i></p> <p>where <i>prefix</i> = either TADIG or REALM <i>code</i> = If <i>prefix</i>=GSM, <i>code</i> = any GSMA assigned TADIG code in capital ASCII letters available at http://www.gsmworld.org; If <i>prefix</i>=REALM, <i>code</i> = or any valid domain name string</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
GSM-Location-Name	<p>GSM-Location-Name = <i>value</i></p> <p>where <i>value</i> = textual description of the WLAN Hot Spot (human readable string without mandated format).</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Example

```
[NAS_1]
GSM-Operator-Name=REALM:worldnetwork.com
GSM-Location-Name=BostonNeighborsClub
GSM-Visited-Operator-Id=GSM:USACD
GSM-Location-Information=country=US;A1=MA;A3=Boston;ZIP=02116
```

Figure 10 on page 236 shows the relationship between the **AttributetoIdentifyNAS** setting, the [NAS-LIST] section, and the [NAS identifier] section of the **locspec.ctrl** file.

Figure 10: [Settings], [NAS-LIST], and [NAS identifier] Sections of the locspec.ctrl File



proxy.ini File

The **proxy.ini** file identifies the **.pro** files that are used to specify configuration settings. With respect to adding location information to an Access-Accept, the **.pro** files are needed to invoke the **LOCSPEC** plug-in.

[Table 77 on page 236](#) defines the fields needed in the [Realms] section for adding location-specific attributes to an Access-Request.

Table 77: proxy.ini [Realms] Fields for Configuration of Location-specific Attributes

Field	Description
realm_name	<p>Lists all the realms that can be included in an Access-Request.</p> <p>For every realm_name, there must be an associated realm.pro file. For example, if the [Realms] section contains the lines:</p> <pre>[Realms] CountryNet=countrysnet.com</pre> <p>There must be an associated countrysnet.profile.</p>

Example

```
[Realms]
Realm_Example_1=nationwide.com
Realm_Example_2=peoplesnetwork.com
```

realm.pro File

The **realm.pro** file specifies the control point plug-in that is needed for attaching location-specific information to an Access-Request if the Access-Request is proxied from a foreign AAA server to the home AAA server.

Add the field **LOCSPEC** to both the [Auth-Outbound-To-Proxy] section and the [Acct-Outbound-To-Proxy] section in the **realm.pro** file. These sections call the location-specific control plug-in when an Access-Request is proxied (forwarded) to a home AAA server.

Example realm.pro file:

```
[Auth-Outbound-To-Proxy]
LOCSPEC
[Acct-Outbound-To-Proxy]
LOCSPEC
```

NOTE: The [Auth-Outbound-To-Proxy] section and the [Acct-Outbound-To-Proxy] sections are required in the **realm.pro** files that are related to adding location information to an Access-Request. However, the **realm.pro** files require additional sections that are related to the functionality of Steel-Belted Radius Carrier. See the *SBR Carrier Administration and Configuration Guide* for more information about configuring realm support.

Example Configuration for Adding NAS Location Attributes to Access-Request

[Figure 11 on page 238](#) shows a sample configuration. The purpose of this example configuration is to add NAS location information to Access-Requests for NAS_1.

Example Overview

In this example, an Access-Request is sent for a mobile device through an example NAS identified by the name NAS_1. The example realm associated with the mobile device is nationwide.com. Three location attributes are assigned to NAS_1 and included in the Access-Request that goes to the nationwide.com service provider. These three attributes are GSM-Location-Name, GSM-Operator-Name, and GSM-Visited-Operator-Id.

Example Configuration

The example configuration lines and syntax (shown in [Figure 11 on page 238](#)) associate all the configuration files together to attach NAS location information to an Access-Request.

The example configuration shows that if the realm is **nationwide.com**, then the **.pro** file to be used is **Realm_Example_1.pro**. The file **Realm_Example_1.pro** turns on NAS location information feature with the **LOCSPEC** commands.

Figure 11: Example Configuration for Adding Location Information to an Access-Request

Address Assignment Files

IN THIS CHAPTER

- [dhcp.ini File | 239](#)
- [pool.dhc Files | 242](#)

This chapter describes the usage and settings for the Steel-Belted Radius Carrier initialization (.ini) files that are used to enable, disable, and configure IP address assignment. The following topics are included in this chapter:

dhcp.ini File

The **dhcp.ini** configuration file configures DHCP address pools so that IPv4 addresses can be assigned from a back-end DHCP server, rather than from a standard Steel-Belted Radius Carrier IP address pool.

NOTE: Steel-Belted Radius Carrier does not support DHCP allocation of IPv6 addresses.

[Settings] Section

The [Settings] section of the **dhcp.ini** file ([Table 78 on page 239](#)) controls DHCP address allocation.

Table 78: dhcp.ini [Settings] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">● If set to 1, DHCP address allocation is enabled.● If set to 0, DHCP address allocation is disabled. <p>Default value is 0.</p>

Table 78: dhcp.ini [Settings] Syntax (*continued*)

Parameter	Function
Attempts	<p>Specifies the number of times a DHCP DISCOVER or REQUEST message is sent if no response is received.</p> <p>Default value is 3.</p>
AttemptTimeout	<p>Specifies the waiting period, in seconds, for a response to a DISCOVER or REQUEST message, before resending the message.</p> <p>Default value is 5 seconds.</p> <p>NOTE: If present, the AttemptTimeoutMs parameter overrides this setting.</p>
AttemptTimeoutMs	<p>Specifies the waiting period, in milliseconds, for a response to a DISCOVER or REQUEST message, before resending the message.</p> <p>Default value is 5000 milliseconds (5 seconds).</p> <p>NOTE: If present, this parameter overrides the AttemptTimeout setting.</p>
OverallTimeout	<p>Specifies the number of seconds for acquiring an IP address before DHCP address assignment is presumed to have failed.</p> <p>This timeout applies only to the DISCOVER/REQUEST sequence used to acquire an address initially, not to address renewal or release.</p> <p>Default value is 15 seconds.</p> <p>NOTE: If present, the OverallTimeoutMs parameter overrides this setting.</p>
OverallTimeoutMS	<p>Specifies the number of milliseconds for acquiring an IP address before DHCP address assignment is presumed to have failed.</p> <p>This timeout applies only to the DISCOVER/REQUEST sequence used to acquire an address initially, not to address renewal or release.</p> <p>Default value is 15000 milliseconds. (15 seconds).</p> <p>NOTE: If present, this parameter overrides the OverallTimeout setting.</p>

Table 78: dhcp.ini [Settings] Syntax (*continued*)

Parameter	Function
htype	<p>Specifies the client hardware type (0–255).</p> <p>This parameter is typically omitted, because the value is generated automatically.</p>
Hlen	<p>Specifies the length of the client hardware address (1–16).</p> <p>This parameter is typically omitted, because the value is generated automatically.</p>
Chaddr-prefix	<p>Specifies the string that identifies the initial bytes of the client hardware address (chaddr). This string can include escape codes, including <code>\nnn</code> for decimal values and <code>\xnn</code> for hex values.</p> <p>This parameter is typically omitted, because the value is generated automatically.</p>
ServerPort	<p>Specifies the UDP port number on which the DHCP server(s) listen. This setting is used only for non-standard DHCP configurations.</p> <p>Default value is 67, which is the standard DHCP server port.</p>
LocalPort	<p>Specifies the UDP port number that Steel-Belted Radius Carrier, acting as a relay agent, uses during DHCP communication. This setting is used for only non-standard DHCP configurations.</p> <p>Default value is 67, which is the standard DHCP server port.</p>
Pad	<p>Specifies the minimum number of bytes for a DHCP request message. Messages smaller than this number are padded with 0s.</p> <p>Certain DHCP servers discard messages smaller than a certain value. This option allows interoperability with such servers.</p> <p>Default value is 300.</p>
CheckDuplicate Assignment	<p>Checks for duplicate addresses assigned by the DHCP server.</p> <ul style="list-style-type: none"> • Yes—Enables the checking for duplicate IP addresses assigned by the DHCP server. • No—Disables the checking for duplicate IP addresses assigned by the DHCP server. <p>Default value is Yes.</p>

The following is a sample **dhcp.ini** file:

```
[Settings]
Enable = 1
Attempts = 3
AttemptTimeout = 2
OverallTimeout = 10
```

[Pools] Section

The [Pools] section lists all DHCP pool names (specified in the **pool.dhc** file, which is described on [“pool.dhc Files” on page 242](#)) in the following format:

```
[Pools]
pool 1
pool 2
```

For example:

```
[Pools]
DHCP_SERVER1
DHCP_SERVER_SALES
```

pool.dhc Files

Each pool listed in the [Pools] section of the **dhcp.ini** file must be a corresponding **pool.dhc** file that configures that pool.

[Settings] Section

The [Settings] section of the **pool.dhc** file ([Table 79 on page 242](#)) controls DHCP lease information.

Table 79: pool.dhc [Settings] Syntax

Parameter	Function
LeaseTime	Set to the lease time, in seconds, to request from the DHCP server. Default value is 1 day.

Table 79: pool.dhc [Settings] Syntax (*continued*)

Parameter	Function
MinLeaseTime	<p>Set to the minimum lease time, in seconds. Offers from DHCP servers with lease time less than this minimum are ignored.</p> <p>Default value is the value set for LeaseTime.</p>
TargetAddress	<p>Set to the address to which DISCOVER messages are sent.</p> <p>Default value is 255.255.255.255, the local broadcast address.</p> <p>This entry should normally remain unchanged, to allow DHCP DISCOVER messages to be broadcast.</p>
ExtendOnStart	<p>Specifies whether to extend the DHCP lease time when SBR Carrier receives an Accounting-Start request from the NAD.</p> <ul style="list-style-type: none"> • 1—Extends the lease time when SBR Carrier receives an Accounting-Start request from the NAD. The value set for the LeaseTime parameter is used for extending the lease time. • 0—Does not extend the lease time when SBR Carrier receives an Accounting-Start request from the NAD. <p>Default value is 0.</p>
InitialLeaseTime	<p>Specifies the initial lease time (in seconds) which is used as the lease time when a DHCP server allocates an address during authentication. This lease time is updated to the value set for LeaseTime when SBR Carrier receives an Accounting-Start request from the NAD.</p> <p>This parameter applies only if the ExtendOnStart parameter is set to 1.</p> <p>Default value is 0 seconds.</p>

[Request] Section

The [Request] section allows options in the DHCP DISCOVER and REQUEST messages to be constructed from attributes in the RADIUS Access-Request and from pre-configured literal values in the following way:

```
[Request]
DHCP option = RADIUS attribute or literal value
DCHP option = RADIUS attribute or literal value
.
```

```

.
.

```

The **DHCP option** contains of the following fields (brackets [] indicate optional text). Fields are not separated by spaces.

[vendor-specific] option [offset] format

Table 80: pool.dhc [Request] Syntax

Parameter	Function
vendor-specific	Set to v if this is a vendor-specific option, or omit otherwise.
option	Set to the DHCP option in the format, nnn .
offset	Set to a period followed by the number of bytes into the option where the value is located, or a plus-sign (+) to indicate a list of values in the DHCP option; each to be mapped to an instance of the RADIUS attribute.
format	Set to the format of the DHCP option, which can be one of the following: <ul style="list-style-type: none"> • n32—a 32-bit integer • n16—16-bit integer • n8—8-bit integer • s or string—string • i or ip—IP address

The following are examples of **DCHP option** fields:

- **1ip** (The “Subnet Mask” option as an IP address)
- **3+ip** (The “Router” option as a list of IP address, each to be mapped to an instance of the RADIUS attribute)
- **6.4ip** (The “DNS Server” option as a second IP address in list (each IP address is 4 bytes))
- **12s** (The “Host Name” as a string)

The RADIUS attribute can be set to the name of any attribute defined in any dictionary. A literal value can be specified instead of a RADIUS attribute. This value must be text enclosed in double-quotes (“ ”).

The string is interpreted based on the format of the DHCP option:

- IP addresses must be specified in dotted notation; for example, 127.0.0.1 for IPv4 networks.
- Integers are expressed in decimal format; for example, 100.

- Strings are expressed as any text sequence.

The text can include escape sequences, where the backslash character (\) is the escape character. [Table 81 on page 245](#) lists escape sequences.

Table 81: Escape Code Sequences

Escape Code	Function
\a	7
\b	8
\f	12
\n	10
\r	13
\t	9
\y	11
\nnn	A decimal value between 0 and 255.
\xnn	A hexadecimal value between 00 and FF
\\	A literal backslash \
\"	A double-quote
\char	A single character, interpreted literally

NOTE: You must use an escape character to include a literal backslash (\) or double-quote (") in the string.

An escape sequence can be used to set an option to an arbitrary binary value. This is useful, for example, when setting the Vendor Class Identifier option (60).

The following example sets the DHCP Host Name option to the RADIUS Calling-Station-Id, and sets the DHCP Vendor Class Identifier option to a binary string:

```
[Request]
12s = Calling-Station-Id
60s = "\x01\x02\x03\x04\x05"
```

[Reply] Section

The [Reply] section allows RADIUS Access-Accept attributes to be constructed from options the DHCP server returns in an ACK message, in the following way:

```
[Reply]
RADIUS attribute = DHCP option
RADIUS attribute = DHCP option
.
.
.
```

See the [Request] section for information about how to specify the *RADIUS attribute* and the *DHCP option* values.

NOTE: In contrast to the [Request] section, the left and right sides of the equal sign are reversed to account for the direction in which the data is being set.

The following example returns the RADIUS Framed-IP-Netmask attribute from the DHCP Subnet Mask option and sets the RADIUS Framed-MTU attribute from the DHCP Interface MTU option:

```
[Reply]
Framed-IP-Netmask = 1ip
Framed-MTU = 26n16
```

Reconfiguring Pools

DHCP pool information is loaded at startup from the **dhcp.ini** file and all associated **pool.dhc** files. DHCP pools can be added, deleted, and modified dynamically by doing the following:

1. Modify the **dhcp.ini** file and the **pool.dhc** files as required.
2. Restart the RADIUS process by issuing the SIGHUP (1) signal to the Steel-Belted Radius Carrier process:

```
#./sbrd hup
```

Steel-Belted Radius Carrier reads the modified files and configures its DHCP pools.

Accounting Configuration Files

IN THIS CHAPTER

- [account.ini File | 247](#)
- [acctReport.ini File | 256](#)
- [sessionTable.ini File | 260](#)

This section describes the usage and settings of the **account.ini** initialization file, that enables, disables, and configures accounting features of the server. The initialization file is loaded at startup from the Steel-Belted Radius Carrier directory. The following topics are included in this chapter:

NOTE: Throughout this section, the term *attributes* refers to both standard RADIUS attributes and structured attributes. For information about specifying structured attributes, see [“Structured Attributes” on page 199](#).

account.ini File

The **account.ini** file contains information that controls how RADIUS accounting attributes are logged to a comma-delimited text file by Steel-Belted Radius Carrier. Specifically, the **account.ini** file controls file creation settings, such as file creation frequency, maximum size, and default directory, and file content, such as what information is recorded for each received accounting request.

[Alias/name] Sections

The [Alias/**name**] sections of **account.ini** are used to associate attributes of different names, but identical meaning. For example, one network access server vendor might call an attribute **AcctOctetPkt** and another might call it **AcctOctPackets**, yet the two attributes mean the same thing.

Each [Alias/*name*] section permits you to map one RADIUS accounting attribute that is already being logged by Steel-Belted Radius Carrier to any number of other attributes. You can provide as many [Alias/*name*] sections as you want, using the following syntax for each section:

```
[Alias/name]
VendorSpecificAttribute=
VendorSpecificAttribute=
.
.
.
```

Table 82: account.ini [Alias]name] Syntax

Parameter	Function
name	The preferred attribute name. The name attribute must be one that you are currently logging to a column in the Steel-Belted Radius Carrier accounting log file (.act). Therefore, it must be listed in the [Attributes] section of account.ini .
VendorSpecificAttribute	Each entry is given on one line. An equal sign (=) must immediately follow each VSA name, without any intervening space. Improperly formatted entries are considered invalid and are ignored.

Each **VendorSpecificAttribute** in the list is logged to the *name* column in the accounting log file. Because you are listing these attributes in an [Alias/*name*] section, verify they are not listed in the [Attributes] section; otherwise, they are logged to their own columns as well as the *name* column.

All of the attribute names that you reference in an [Alias/*name*] section must be defined in a dictionary file that is already installed on the Steel-Belted Radius Carrier server. This includes *name* and each **VendorSpecificAttribute** entry.

In the following example, the standard RADIUS attribute **AcctOctetPackets** is mapped to the vendor-specific attributes **AcctOctet-Pkt** and **AcctOctPackets**. Values encountered for all three attributes are logged in the AcctOctetPackets column in the accounting log file:

```
[Alias/Acct-Octet-Packets]
Acct-Octet-Pkt=
Acct-Oct-Packets=
```


[Attributes] Section

The [Attributes] section of the **account.ini** file lists all the attributes logged for each received accounting request in the accounting log file (**.act** file). When you install Steel-Belted Radius Carrier, the **account.ini** file is set up so that all standard RADIUS attributes are listed.

You can change the order of columns in the accounting log file by rearranging the sequence of attributes in the [Attributes] section. You can delete or comment out any attributes that are not relevant to your billing system or which do not apply to the equipment that you are using. This lets you design the content and column order of any spreadsheets that you plan to create based upon the accounting log file.

NOTE: SBR Carrier does not update logging behavior when you make changes to the [Attributes] section. After restart, any changes are logged in a newly created **.act** file.

The syntax is:

```
[Attributes]
AttributeName=
AttributeName=
.
.
.
```

For example:

```
[Attributes]
User-Name=
NAS-Port=
Framed-IP-Address=
Acct-Status-Type=
Acct-Delay-Time=
Acct-Session-Id=
```

The [Attributes] section lists one **AttributeName** on each line. You must ensure that an equal sign (=) immediately follows each **AttributeName**, with no spaces in between. Improperly formatted entries are considered invalid and are ignored.

Each **AttributeName** in the [Attributes] section must be defined in a standard RADIUS dictionary file (**.dct** file), a subattribute dictionary file (**.jdict** file), or a vendor-specific dictionary file on the Steel-Belted Radius Carrier server.

NOTE: The first six attributes in each log file entry (Date, Time, RASClient, RecordType, FullName, and AuthType) are always enabled, and cannot be reordered or deleted. Therefore, these attributes do not appear in the **account.ini** file [Attributes] section.

You can add subattributes to the accounting log file using the following syntax:

AttributeName.Values.SubAttributeName

For example:

```
[Attributes]
User-Name=
Calling-Station-ID=
WiMAX-BS-ID.Values.NAP=
WiMAX-BS-ID.Values.BaseStationID=
```

NOTE: Some structured attributes contain a continuation field. Logging this field can be avoided by logging the subattributes instead.

[Configuration] Section

Table 83: account.ini [Configuration] Syntax

Parameter	Function
LogDir	<p>Sets the destination directory on the local host where accounting log files are stored.</p> <p>Default value is the Steel-Belted Radius Carrier directory.</p> <p>NOTE: You cannot write accounting log files to a linked drive.</p> <p>NOTE: With directed realms, you can maintain separate accounting log locations for each realm.</p>

[Settings] Section

Steel-Belted Radius Carrier writes all accounting data to the current accounting log file (.act) until that log file is closed. After closing the file, Steel-Belted Radius Carrier opens a new one and begins writing accounting data to it. You can configure how often this rollover of the accounting log file occurs.

The naming conventions for accounting log files permit more than one file to be generated during a day. [Table 84 on page 251](#) lists the file naming conventions used for different rollover periods. In the examples below, *y*= year digit, *M*= month digit, *d*= day digit, *h*= hour digit, and *m*= minute digit. When more than one file is generated during a day, the sequence number *_nnnnn* starts at *_00000* each day.

Table 84: Accounting File Rollover

File Generation Method	File Naming Convention
Default (24 hours)	yyyyMMdd.act
Non-24-hour rollover	yyyyMMdd_hhmm.act
Rollover based on size only	yyyyMMdd_nnnnn.act
Rollover based on both time and size	yyyyMMdd_hhmm_nnnnn.act

NOTE: An accounting log file is not created when there is no accounting request or activity from the client.

The [Settings] section of the **account.ini** file ([Table 85 on page 251](#)) controls how entries are written to the accounting log file, and ensures the compatibility of these entries with a variety of database systems.

Table 85: account.ini [Settings] Syntax

Parameter	Function
BufferSize	<p>The size of the buffer used in the accounting logging process, in bytes.</p> <p>Default value is 131072 bytes.</p>
Enable	<ul style="list-style-type: none"> • If set to 1, the accounting log feature is enabled. • If set to 0, no .act files are created on this server. <p>Accounting servers should have Enable set to 1; for efficiency, nonaccounting servers should have Enable set to 0.</p> <p>Default value is 1.</p>

Table 85: account.ini [Settings] Syntax (*continued*)

Parameter	Function
LineSize	<p>Number in the range 1024–32768 that specifies the maximum size of a single accounting log line.</p> <p>Default value is 4096.</p> <p>NOTE: Logging will fail if this value is exceeded.</p>
LogFilePermissions	<p>Specifies the owner and access permission setting for the accounting log file.</p> <p>Enter a value for the LogFilePermissions setting in owner:group permissions format, where:</p> <ul style="list-style-type: none"> • owner specifies the owner of the file in text or numeric format. • group specifies the group setting for the file in text or numeric format. • permissions specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r----- specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p> <p>The default is owner:group mode.</p>
LogHighResolutionTime	<ul style="list-style-type: none"> • If set to 0, the timestamp written to the accounting log file is recorded in the format of <i>MM/DD/YYYY/hh:mm:ss</i> (month/date/year/hour:minutes:seconds). • If set to 1, the timestamp written to the accounting log file is recorded in the format of <i>MM/DD/YYYY/hh:mm:ss.xxx</i>, where <i>xxx</i> represents the number of elapsed milliseconds since the <i>ss</i> value changed. <p>Default value is 0.</p>
MaxSize	<p>The maximum size of an accounting log file, in bytes.</p> <p>If the accounting log file reaches or exceeds this size when it is checked, the log file is closed and a new file started. A value of 0 (the default) means unlimited size.</p>

Table 85: account.ini [Settings] Syntax (*continued*)

Parameter	Function
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>

Table 85: account.ini [Settings] Syntax (*continued*)

Parameter	Function
RollOver	<p>Specifies how often the current accounting log file is closed and a new file opened (a rollover), up to one rollover per minute. Nonzero values indicate the number of minutes until the next rollover.</p> <p>If set to 0, the accounting log file rolls over once every 24 hours, at midnight local time.</p> <p>Default value is 0.</p>
RollOverOnStartup	<ul style="list-style-type: none"> • If set to 1, each time Steel-Belted Radius Carrier is started, it closes the current accounting log file and opens a new one. A sequence number <i>_nnnnn</i> is appended to the log file name, just as when MaxSize is reached. • If set to 0, each time Steel-Belted Radius Carrier is started, it appends entries to the previously open accounting log file. <p>Default value is 0.</p>
Titles	<ul style="list-style-type: none"> • If set to 1, each time a new accounting log file is created, the title line (containing column headings) is written to the file. • If set to 0, the line is not written. <p>Default value is 1.</p>
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>

[TypeNames] Section

Each entry in the [TypeNames] section of **account.ini** maps a possible value of the AcctStatusType attribute to a string. The value of this attribute is written into the fourth column of each accounting log record.

The syntax is:

```
[TypeNames]
TypeID = TypeName
TypeID = TypeName
.
```

```
.  
.
```

Table 86: account.ini [TypeNames] Syntax

Parameter	Function
TypeID	Each TypeID is a numeric value that corresponds to a possible value of the AcctStatusType attribute. This attribute appears in every incoming RADIUS accounting packet to identify the types of data it is likely to contain.
TypeName	Each TypeName value is a string. This string is written to the accounting log to identify the type of packet.

The standard AcctStatusType values 1, 2, 3, 7, and 8 are already listed in the [TypeNames] section of **account.ini** as:

```
[TypeNames]  
1=Start  
2=Stop  
3=Interim  
7=On  
8=Off
```

You can edit the [TypeNames] section to add vendor specific packet types to this list, which makes your accounting log files easier to read and use. For example:

```
[TypeNames]  
1=Start  
2=Stop  
3=Interim  
7=On  
8=Off  
639=AscendType  
28=3ComType
```

If no string is given for a particular AcctStatusType, Steel-Belted Radius Carrier uses the numeric value of the incoming AcctStatusType attribute, formatted as a string.

acctReport.ini File

The **acctReport.ini** file includes information that controls the configuration settings of the accounting log, which provides logging of failed accounting requests. The accounting log records all accounting request failures by logging them in a comma-delimited text file. While viewing the logs, you can sort, search and filter the log records. Each record includes details of who, where, when, and what happened during the accounting request. The accounting log records failures due to:

- Bad shared secret
- Unknown client

You can enable or disable the accounting report logs when needed. The accounting logs are saved in a default folder called **AcctReports** located under the **/opt/JNPRsbr/radius** directory (if not specified). The accounting request failures are saved in two separate files denoted by the failure name and timestamp as follows:

- **badSharedSecret_yyyymmdd.act**
- **unknownClient_yyyymmdd.act**

Each log report is enabled separately.

The following fields are available in both log reports:

- Timestamp (UTC or local time)
- Username (if available)
- Client IP address (May not be accurate due to proxy, NAT)
- Task name—accounting start or stop request and accounting on or off events
- Reason of failure
- NAS name (if known)

[Settings] Section

The [Settings] section of the **acctReport.ini** file ([Table 87 on page 257](#)) defines various global user parameters that affect the log file, such as days to keep and formatting.

If the **MaxMinutesPerFile** parameter is set to 0, the file name of the accounting log report is **badSharedSecret_yyyymmdd.act** or **unknownClient_yyyymmdd.act**, (where **yyymmdd** identifies the date the report was generated.) If the **MaxMinutesPerFile** parameter is set to a value greater than 0, the file name of the report is **badSharedSecret_yyyymmdd_hhmm.act** or **unknownClient_yyyymmdd_hhmm.act** (where **yyymmdd** identifies the date and **hhmm** identifies the time the report was generated.)

Table 87: acctReport.ini [Settings] Syntax

Parameter	Function
LogDir	<p>Specifies the destination directory on the local host where the badSharedSecret_YYYYMMDD.act and unknownClient_YYYYMMDD.act files are stored.</p> <p>Default value is the directory is /opt/JNPRsbr/radius directory (if not specified).</p>
LogFilePermissions	<p>Specifies the owner and access permission settings for the accounting log files.</p> <p>Enter a value for the LogFilePermissions setting in <i>owner:group permissions</i> format, where:</p> <ul style="list-style-type: none"> • <i>owner</i> specifies the owner of the file in text or numeric format. • <i>group</i> specifies the group setting for the file in text or numeric format. • <i>permissions</i> specifies what privileges can be exercised by Owner/Group/Other with respect to the file in text or numeric format. <p>For example, user:1007 rw-r----- specifies that the file owner (user) can read and edit the log file, members of group 1007 can read (but not edit) the log file, and that other users cannot access the log file.</p>
UTC	<ul style="list-style-type: none"> • If set to 1, time and date values are provided according to UTC (GMT). • If set to 0, time and date values reflect local time. <p>Default value is 0.</p>
MaxMinutesPerFile	<p>Specifies how often the current accounting log report is closed and a new file opened.</p> <ul style="list-style-type: none"> • If set to n (where n is a number greater than 0), a new report is generated every n minutes. • If set to 0, a new report is generated once every 24 hours, at midnight local time. <p>Default value is 0.</p> <p>NOTE: The value entered for MaxMinutesPerFile determines the file name of the generated report.</p>

Table 87: acctReport.ini [Settings] Syntax (*continued*)

Parameter	Function
DaysToKeep	<p>Specifies the number of days the Steel-Belted Radius Carrier server retains the accounting log reports.</p> <p>Default value is 1 (one day).</p>
LineSize	<p>The maximum size of a single log line. The allowable range is 1024 to 32768.</p> <p>Default value is 4096.</p> <p>NOTE: Logging fails if this value is exceeded.</p>
QuoteInteger	<ul style="list-style-type: none"> • If set to 1, integer values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteIPAddress	<ul style="list-style-type: none"> • If set to 1, IP addresses written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteText	<ul style="list-style-type: none"> • If set to 1, text strings written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>
QuoteTime	<ul style="list-style-type: none"> • If set to 1, time and date values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>

Table 87: acctReport.ini [Settings] Syntax (continued)

Parameter	Function
QuoteBinary	<ul style="list-style-type: none"> • If set to 1, binary values written to the accounting log file are enclosed in quotes. • If set to 0, quotes are not used. <p>Set this value according to the format expected by the accounting application that processes the entries.</p> <p>Default value is 1.</p>

[UnknownClientReport] Section

The [UnknownClientReport] section ([Table 88 on page 259](#)) of **acctReport.ini** enables or disables generation of the unknown accounting client report, which is an ASCII comma-delimited file produced by the Steel-Belted Radius Carrier server. It identifies accounting requests received from unknown RADIUS clients, or destined to unknown or unreachable destinations. It also defines the file format of the unknown-client reports.

Table 88: acctReport.ini [UnknownClientReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, the unknownClient_yyyymmdd.act is enabled. • If set to 0, the unknownClient_yyyymmdd.act is disabled. <p>Default value is 0.</p>

[BadSharedSecretReport] Section

The [BadSharedSecretReport] section ([Table 89 on page 259](#)) of **acctReport.ini** enables or disables generation of the invalid shared secret report, which is an ASCII comma-delimited file that records information about requests received from known RADIUS clients that used an invalid shared secret. It also defines the file format of the bad shared secret reports.

Table 89: acctReport.ini [BadSharedSecretReport] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1, the badSharedSecret_yyyymmdd.act is enabled. • If set to 0, the badSharedSecret_yyyymmdd.act is disabled. <p>Default value is 0.</p>

[Attributes] Section

The [Attributes] section of **acctReport.ini** lists the attributes you want logged in the accounting reports. For example:

```
[Attributes]
NAS-IP-Address=
User-Name=
```

sessionTable.ini File

The **sessionTable.ini** file allows you to take any attribute in a request and store it in the CST.

[Settings] Section

In the [Settings] section of the **sessionTable.ini** file, the parameter in [Table 90 on page 260](#) ensures unique attributes in the session table.

Table 90: sessionTable.ini [Settings] Syntax

Parameter	Function
GenerateUniqueld	<ul style="list-style-type: none"> • If this is set to default, the CST's UniqueSessionId is created in the regular manner. • If this is set to ipaddr, and if a Framed-IP-Address attribute was received, the CST's UniqueSessionId is set to 4 bytes of IPv4 address, and 12 bytes of 0. This ensures that the Framed-IP-Address attribute is unique in the session table. • If this is set to ipaddr-plus-nas, and if a Framed-IP-Address attribute was received, the CST's UniqueSessionId is set to 4 bytes of IPv4 address, and up to 12 bytes of the NAD Name, with 0s following the NAD Name if less than 12 bytes. This ensures that the Framed-IP-Address attribute is unique in the session table for each NAD. • If this is set to outernai-plus-mac, the CST's UniqueSessionId is a combination of Outer-NAI and MAC address. • If set to acct-session-id-plus-nas, the UniqueSessionId is created always based on NAD name and Acct-Session-Id. <p>NOTE: For accounting records to be matched with authentications with this option, the Access-Request must contain an Acct-Session-Id.</p> <p>Default value is default.</p> <p>NOTE: The ipaddr and ipaddr-plus-nas settings are meant to be used only by accounting servers. The outernai-plus-mac setting is used for WiMAX.</p>

Table 90: sessionTable.ini [Settings] Syntax (continued)

Parameter	Function
UpdateAllOnStart	<p>This parameter is primarily used in the case of GenerateUniqueID with “ipaddr” or similar settings to force overwriting of all session data when a new start comes in for the same unique ID.</p> <p>Default value is “false.”</p>

See *Steel-Belted Radius Carrier Installation Guide* for information about configuring the sessionTable.ini file.

Realm Configuration Files

IN THIS CHAPTER

- Proxy Realm Configuration Files | 263
- Directed Realm Configuration Files | 266
- proxy.ini File | 268
- Proxyrl.ini File | 279
- Proxy RADIUS Configuration (.pro) File | 280
- Directed Realm Configuration (.dir) File | 300
- radius.ini Realm Settings | 307

This chapter describes the configuration files relating to proxy and directed realm administration in Steel-Belted Radius Carrier.

NOTE: Throughout this chapter, the term attributes refers to both standard RADIUS attributes and structured attributes. For information about specifying structured attributes, see [“Structured Attributes” on page 199](#).

[Table 91 on page 262](#) lists the files in the Steel-Belted Radius Carrier directory you must edit to configure realms.

Table 91: Realm Configuration Files

File Name	Purpose
radius.ini	Enables and disables realm features. For more details, see “radius.ini File” on page 24 .
proxy.ini	Specifies the order of realm selection methods, the realm selection rules, and other settings for all realms on the server. For more details, see “proxy.ini File” on page 268 .

Table 91: Realm Configuration Files (*continued*)

File Name	Purpose
<i>RealmName.pro</i>	For each proxy realm that you want to configure on the Steel-Belted Radius Carrier server, you must create a file called <i>RealmName.pro</i> , where <i>RealmName</i> is the name of the realm, and you must register this RealmName by listing it in the [Realms] section of the proxy.ini file.
<i>RealmName.dir</i>	For each directed authentication and accounting realm that you want to configure on the Steel-Belted Radius Carrier server, you must create a file called <i>RealmName.dir</i> , where <i>RealmName</i> is the name of the realm, and you must register this RealmName by listing it in the [Directed] section of proxy.ini .
filter.ini	Stores filters for RADIUS attributes and subattributes; these filters may be referenced from the [Auth] or [Acct] section of a <i>RealmName.pro</i> or <i>RealmName.dir</i> file. For more details, see “filter.ini File” on page 213 . NOTE: Do not edit the filter.ini file manually. Use Web GUI to configure rules for filtering RADIUS attributes and subattributes.

The following topics are included in this chapter:

Proxy Realm Configuration Files

This section describes how to set up the proxy realm configuration files.

Sample proxy.ini Settings

The following **proxy.ini** file registers a proxy realm called **sample.com** and adds that realm to the list of target realms for static proxy accounting.

```
[Realms]
sample.com

[StaticAcct]
7=CustAOnOff
8=CustAOnOff
```

```
[CustAOnOff]
realm=sample.com
```

NOTE: For syntax details, see [“proxy.ini File” on page 268](#).

The following **proxy.ini** file entry specifies that **otto@rtt.other.com** and **carol@3g.other.com** are both mapped to the **other.com** proxy realm.

```
[Realms]
other.com = *.other.com
```

The following **proxy.ini** file specifies that **otto@rtt.other.com** and **carol@3g.other.com** are mapped to the **other.com** proxy realm and that **caitlin@groton.other.com** is mapped to the **groton.other.com** proxy realm.

```
[Realms]
other.com = *.other.com
groton.other.com
```

Sample Proxy Realm (.pro) File

The following complete file must be called **sample.com.pro** for it to work with the sample **proxy.ini** file shown on [“Sample proxy.ini Settings” on page 263](#).

```
[Auth]
Enable = 1
AvailableAsAuthMethod = no
TargetsSection = AuthTargets
RoundRobin = 2
StripRealm = 0
RequestTimeout = 5
NumAttempts = 3
FilterOut = CustAOut
FilterIn = CustAIn
MessageAuthenticator = 0
UseMasterDictionary = yes
UserConcurrency = 0

[Acct]
Enable = 1
```



```

TargetsSection = AcctTargets
RoundRobin = 1
StripRealm = 0
RequestTimeout = 5
NumAttempts = 3
FilterOut = CustAOut
; FilterIn =
RecordLocally = 1
; Block = 1
UseMasterDictionary = yes

[AuthTargets]
bunion=1
desktop=1

[AcctTargets]
desktop

[Called-Station-ID]
8885551212
5551234

[FastFail]
MinFailures = 3
MinSeconds = 3
ResetSeconds = 600

```

NOTE: For syntax details, see [“Proxy RADIUS Configuration \(.pro\) File”](#) on page 280.

This example expects the Steel-Belted Radius Carrier database to contain Proxy entries with target names **Desktop** and **Bunion**. These entries are required to provide the network routing information (IP address, RADIUS shared secret, and UDP ports) that allows forwarded packets to reach the target servers at the customer site.

Sample filter.ini File

The following complete sample **filter.ini** file defines the two attribute filters referenced in the **sample.com.pro** file shown on [“Sample Proxy Realm \(.pro\) File”](#) on page 264:

```

[CustAOut]
ALLOW

```

```

EXCLUDE NAS-IP-Address
ADD NAS-IP-Address 1.2.3.4

[CustAIn]
EXCLUDE
ALLOW Session-Timeout
ALLOW Idle-Timeout
ALLOW Service-Type Framed
ADD Service-Type Framed
ADD Framed-IP-Address CustAPool

```

The **CustAOut** filter in this example is designed to be applied to request packets coming into the Steel-Belted Radius Carrier server that are directed out to the realm. It allows all of the attributes in the packet to go out to the realm, with the exception of the RADIUS client's IP address. It replaces this IP address with the specific dummy address 1.2.3.4. This filter enhances overall security by not publishing routing information to the network when it is not necessary to do so.

The **CustAIn** filter in this example is designed to be applied to response packets returning to the Steel-Belted Radius Carrier server, which are relayed, in turn, to the RADIUS client. Most attributes are excluded; however, if any timeout values are returned, they are allowed through. If the Service-Type attribute is present in the response and it has the value **Framed** (a string alias for the Service-Type integer value 2), it is allowed in the packet. Steel-Belted Radius Carrier adds the Service-Type attribute to the packet if it is not already there, and assigns it the value **Framed** (2).

The **CustAIn** filter in this example expects the Steel-Belted Radius Carrier database to contain an IP address pool entry called CustAPool, which specifies the customer's valid address ranges. If this entry is not present, the **CustAIn** filter fails. CustAPool is referenced in the filter's final entry, which assigns a value to the Framed-IP-Address attribute. As shown in the example, this entry causes Steel-Belted Radius Carrier to (1) add the Framed-IP-Address attribute to the packet; (2) select an available address from CustAPool, and (3) assign this value to the Framed-IP-Address attribute.

Directed Realm Configuration Files

This section discusses how to set up the directed realm configuration files.

Sample proxy.ini File

The following **proxy.ini** file registers the proxy realm called **sample.com** and registers a directed authentication and accounting realm called **sample2.com**. It defines several directed accounting methods, including those we plan to reference from the **sample2.com.pro** realm configuration file.

```
[Realms]
sample.com

[Directed]
sample2.com

[DirectedAcctMethods]
CustBAcctSQL = /opt/JNPRsbr/radius/CustomerB/theirsql.acc
CustCAcctAttributes = /opt/JNPRsbr/radius/CustomerC/account.ini
CustCAcctSQLConfig = /opt/JNPRsbr/radius/sqlacct.acc
CustDAcctSQLConfig3 = /opt/JNPRsbr/radius/CustomerD/mysql.acc
```

NOTE: For syntax details, see [“proxy.ini File” on page 268](#).

NOTE: The user directories must be manually created before restarting or issuing a SIGHUP (1) signal to SBR Carrier.

The following **proxy.ini** file specifies that **otto@rtt.other.com** and **carol@3g.other.com** are both mapped to the **other.com** directed realm.

```
[Directed]
other.com = *.other.com
```

The following **proxy.ini** file specifies that **otto@rtt.other.com** and **carol@3g.other.com** are both mapped to the **other.com** directed realm and that **caitlin@groton.other.com** is mapped to the **groton.other.com** directed realm.

```
[Directed]
other.com = *.other.com
groton.other.com
```

Sample Directed Realm (.dir) File

The following configuration file must be called **sample2.com.dir** for it to work with the **proxy.ini** file (described on [“Sample proxy.ini File” on page 266](#)).

```

[Auth]
Enable = 1
StripRealm = 1
UseMasterDictionary = yes

[Acct]
Enable = 1
RecordLocally = 1
UseMasterDictionary = yes

[AuthMethods]
Native User

[AcctMethods]
CustCAcctAttributes
CustCAcctSQLConfig

[Called-Station-Id]
8885551212
55512340

```

NOTE: For syntax details, see [“Directed Realm Configuration \(.dir\) File”](#) on page 300.

This sample file configures both directed authentication and directed accounting. It also strips realm routing information from the User-Name before authentication.

The [Acct Methods] section of this file lists the two accounting methods for the **sample2.com realm**. These are **CustCAcctAttributes**, which specifies how to log attributes to a **.act** accounting log file on the local server, and **CustCAcctSQLConfig**, which configures accounting to an external SQL database. Both methods are configured in the [DirectedAcctMethods] section of our sample **proxy.ini** file, above.

proxy.ini File

The **proxy.ini** file specifies the order of realm selection methods, the realm selection rules, and other settings for all realms on the server. Settings for a realm are provided in its **RealmName.pro** or **RealmName.dir** file.

After you edit **proxy.ini**, you must apply your changes:

- If you configured any proxy realms, you can load your new realm configuration without stopping and restarting the server.

Issue the **SIGHUP (1)** signal to the Steel-Belted Radius Carrier process.

#./sbrd hup

Steel-Belted Radius Carrier re-reads **proxy.ini**, **filter.ini**, and all ***.pro** and ***.dir** files in the server directory, and resets its realm configuration.

- If you configured any directed realms and if you added or changed:
 - **Any directed accounting methods:** you must stop and restart the server to load your new configuration.
 - **Directed authentication methods in which external database (SQL or LDAP) authentication is used,** you must stop and restart the server to load your new configuration.
 - **Directed authentication methods in which local or pass-through (Native, UNIX, or Host) authentication is used,** you can load your realm configuration by using a SIGHUP (1) signal.

[Configuration] Section

The [Configuration] section ([Table 92 on page 269](#)) of **proxy.ini** permits you to define prefix and suffix conventions for realm name parsing and specifies whether to use the primary RADIUS dictionary to process inbound proxy responses.

You can enable prefix and suffix conventions for realm name parsing if you specify a different delimiter character for each. All prefixed name decorations must use the prefix delimiter, and all suffixed name decorations must use the suffix delimiter.

If you set the prefix and suffix delimiter to the same character, both prefix and suffix conventions are enabled, but (since suffixes are checked first) prefixes may be misinterpreted.

Select different delimiter characters for tunnels, proxies, and realms.

Table 92: proxy.ini [Configuration] Syntax

Parameter	Function
RealmPrefix	<p>Specifies the character used to identify prefixed name decorations; for example, RAS1/RAS2/joeuser.</p> <p>Default value is /.</p> <p>NOTE: Enter \\ to specify the backslash character, since a single backslash in a configuration file indicates a line continuation.</p>

Table 92: proxy.ini [Configuration] Syntax (*continued*)

Parameter	Function
RealmSuffix	<p>Specifies the character used to identify suffixed name decorations; for example, joeuser@RAS1@RAS2.</p> <p>Default value is @.</p> <p>NOTE: Enter \\ to specify the backslash character, since a single backslash in a configuration file indicates a line continuation.</p>
UseMasterDictionary	<ul style="list-style-type: none"> • If set to yes, inbound proxy responses use the primary Steel-Belted Radius Carrier dictionary when attributes are filtered in. • If set to no, proxy responses use the client-specific dictionary when attributes are filtered in. <p>Default value is yes.</p> <p>NOTE: The UseMasterDictionary setting configured in individual .dir or .pro files overrides the global setting configured in the proxy.ini file.</p>
SuppressResponseSelfStatic AcctFails	<p>If set to yes, proxy accounting requests are discarded if static accounting or smart static accounting fails.</p> <p>If set to no, proxy accounting responses are sent even if static accounting or smart static accounting fails.</p> <p>Default value is no.</p>

[Realms] Section

The [Realms] section ([Table 93 on page 271](#)) of **proxy.ini** lists all of the proxy realms known to the server. The syntax is:

```
[Realms]
RealmName
RealmName [= match_rule]
RealmName [= <undecorated>]
.
.
.
```

Table 93: proxy.ini [Realms] Syntax

Parameter	Function
<i>RealmName</i>	Each entry must match the name of a <i>RealmName.pro</i> file in the same directory as proxy.ini .
<i>= match_rule</i>	Optional. Specifies a rule for mapping the domain information in a User-Name to a proxy realm by means of prefix or suffix wildcards.
<i>= <undecorated></i>	Optional. Marker indicating the specified realm is used to process requests containing undecorated User-Name information.

[Directed] Section

The [Directed] section ([Table 94 on page 271](#)) of **proxy.ini** lists the names of all of the directed authentication and accounting realms on the server. The syntax for the [Directed] section is:

```
[Directed]
RealmName
RealmName [= match_rule]
RealmName [=<undecorated>]
.
.
.
```

Table 94: proxy.ini [Directed] Syntax

Parameter	Function
<i>RealmName</i>	Each entry must match the name of a <i>RealmName.dir</i> file in the same directory as proxy.ini .
<i>= match_rule</i>	Optional. Specifies a rule for mapping the domain information in a User-Name to a directed realm by means of prefix or suffix wildcards.
<i>= <undecorated></i>	Optional. Marker indicating the specified realm is used to process requests containing undecorated User-Name information.

[Processing] Section

If the [Processing] section (Table 95 on page 272) is present, it lets you specify which realm selection rules are applied and the order in which they are applied. If no [Processing] section is present, routing continues in its default behavior.

NOTE: If the script keyword appears in the [Processing] section, Steel-Belted Radius Carrier executes the realm selection script first, before trying other built-in methods. For more information about the optional scripting module, see the *SBR Carrier Administration and Configuration Guide*.

```
[Processing]
RealmSelector
.
.
.
```

Table 95: proxy.ini [Processing] Syntax

Parameter	Function
RealmSelector	This can be one of six identifiers: Attribute-Mapping, DNIS, Prefix, Suffix, Undecorated or RealmScript (requires optional scripting license). Only the rules corresponding to the values listed are applied, and they are applied in the order you specify them.

The following example enables undecorated Usernames, suffix delimiters, prefix delimiters, and DNIS rules (in that order).

```
[Processing]
Undecorated
Suffix
Prefix
DNIS
```

[AttributeMap] Sections

The [AuthAttributeMap] and [AcctAttributeMap] sections of **proxy.ini** let you map the presence, absence, or specific value of an attribute or subattribute in the incoming packet to a specific realm. This is referred to as *attribute mapping*.

An [AuthAttributeMap] or [AcctAttributeMap] section consists of one or more **RealmName** entries. Each **RealmName** must match the name of a realm configuration file (**RealmName.pro** or **RealmName.dir**) in the same directory as **proxy.ini**.

NOTE: Attribute and subattribute mapping is supported by proxy realms and directed realms. You cannot use this feature when forwarding packets to a proxy target that is not accessed through a realm.

Each **RealmName** entry is a list of statements that can be true or false regarding the attributes or subattributes in an incoming RADIUS packet; we call these statements rules. Rules found in [AuthAttributeMap] apply to authentication packets; rules found in [AcctAttributeMap] apply to accounting packets. In all other respects, [AuthAttributeMap] or [AcctAttributeMap] are the same. The syntax for individual rules may vary; the following example shows all of the possible syntax variations:

```
[AuthAttributeMap]
RealmName
    Attribute=Value
    Attribute
    ~Attribute=Value
    ~Attribute
.
.
.
[AcctAttributeMap]
.
.
.
```

For example:

```
[AuthAttributeMap]
CustTRealm
    Framed-Protocol=1
    Service-Type=2
CustQRealm
    Framed-Protocol=PPP
    ~Service-Type=Framed
NativeRealm
```

Each attribute or subattribute mapping rule must begin with a space or tab character, followed optionally by a tilde '~', then the name of a standard or vendor-specific RADIUS Attribute or subattribute that is in one of the Steel-Belted Radius Carrier dictionary files. If a Value is present, it is preceded by an equal sign

'=', and must specify a valid possible value for that attribute or subattribute. You can use wildcards ('?' and '*') for values. A '?' wildcard matches any character and a '*' wildcard matches the remainder of the string (but can appear only at the end of a string). For example, entering **Called-Station-ID=800*** indicates any 800 number. The rule is terminated by a carriage return. Tilde '~' indicates that the rule is satisfied only if the attribute or attribute value pair is not present in the packet.

Each **RealmName** entry in an [AuthAttributeMap] or [AcctAttributeMap] section is examined in sequence from top to bottom. Within each **RealmName** entry, each rule is evaluated in sequence from top to bottom. The results are:

- If all of the rules in a **RealmName** entry evaluate to **true**, the packet is routed to the realm called **RealmName** and the remaining entries in the attribute map are ignored.
- If any of the rules in a **RealmName** entry evaluate to **false**, this entry does not result in a mapping. Steel-Belted Radius Carrier evaluates the next entry in the map.
- If Steel-Belted Radius Carrier encounters a **RealmName** entry that contains no rules, the packet is automatically directed to that realm.

[Table 96 on page 274](#) explains how the various types of rules are evaluated.

Table 96: Attribute Mapping Rules

Syntax Variation	Function of the Attribute Mapping Rule
<i>Attribute=Value</i>	<p>If the Attribute is present in the request packet and it has the Value shown, then this rule is true. If the Attribute is not present, or if it is present but does not have the Value shown, then this rule is false.</p> <p>NOTE: The Steel-Belted Radius Carrier dictionary file radius.dct provides string aliases for certain integer values defined in the RADIUS standard. You can use these strings in attribute mapping rules.</p>
<i>Attribute</i>	<p>If the Attribute is present in the request packet, then regardless of its value, this rule is true. If the Attribute is not present, then this rule is false.</p> <p>NOTE: You are likely to use the <i>Attribute</i> rule without a <i>Value</i> infrequently, because most of the RADIUS packets coming into your configuration will contain the same set of RADIUS attributes, but with different values.</p>

Table 96: Attribute Mapping Rules (*continued*)

Syntax Variation	Function of the Attribute Mapping Rule
~Attribute=Value	Note the tilde (~) operator. This rule is looking for a specific attribute that may have any value except the one listed. If Attribute is present in the request packet and it does not have the Value shown, then this rule is true. If Attribute is not present, or if it is present but does have the Value shown, then this rule is false. NOTE: The following is not valid syntax: Attribute=~Value
~Attribute	Note the tilde (~) operator and the absence of a Value. If Attribute is not present in the request packet, then this rule is true. If Attribute is present, then this rule is false.

When setting up [AuthAttributeMap] or [AcctAttributeMap] rules for your configuration, distinguish between the different realms whose requests you are processing. Consider how specific your rules must be to identify each realm uniquely. Is the presence of a particular attribute sufficient (**Ascend-IP-Address**), or must the attribute have a specific value before you can be sure of its source (**NASIPAddr= n.n.n.n**)? Make sure that your logic does not permit a crossing of requests between realms.

If a realm destination has been identified by applying an [AuthAttributeMap] entry to the attributes in a session's authentication request, Steel-Belted Radius Carrier uses the same realm for that session's accounting requests (if the realm is enabled for accounting). Generally, this is the desired behavior for the realm. Provide an [AcctAttributeMap] entry only if there is no [AuthAttributeMap] entry for a realm and you want to map the realm using one or more accounting attributes.

[DirectedAcctMethods] Section

The [DirectedAcctMethods] section of the **proxy.ini** file lists one or more external database accounting configuration files (**.acc**) or local accounting initialization files (**.ini**) on the local server, and assigns each of these files a name by which it may be referenced in a **RealmName.dir** file.

The syntax for the [DirectedAcctMethods] section is:

```
[DirectedAcctMethods]
Description=PathAndFile
Description=PathAndFile
.
.
.
```

where **Description** is the name by which you want to reference the accounting method and **PathAndFile** is the full pathname of a **.acc** or **.ini** file on the local server.

`/usr/lib/extras/acctlib.acc`

`/usr/lib/extras/ouracct.ini`

This is the file that implements the accounting method. The location of this file must not be the Steel-Belted Radius Carrier directory.

- If your **PathAndFile** identifies an **.acc** file, external database accounting is performed as configured in the file. You may reference the Steel-Belted Radius Carrier SQL accounting module in the [Bootstrap] section of this .acc file.
- If your **PathAndFile** identifies an **.ini** file, you may omit the [Bootstrap] section from this file. Normal Steel-Belted Radius Carrier logging is performed, except that:
 - Accounting log entries (for requests that are routed to this accounting method) are written to accounting log files (**.act**) in the specified Path, rather than in the server directory.
 - Logging details (which attributes are logged, and in which order) are controlled by the [Settings] and [Attributes] sections of the **.ini** file listed in **PathAndFile**, rather than the **account.ini** file found in the server directory.

[StaticAcct] Section

Static proxy accounting lets you send duplicate copies of certain types of accounting requests to proxy realms (or any RADIUS-aware device), in addition to the normal routing of the original accounting request. The number of duplicates is not limited.

The [StaticAcct] section of **proxy.ini** maps possible values of the AcctStatusType attribute to a list of proxy realms that receive statically-forwarded, duplicate copies of all accounting packets of that type.

AcctStatusType is a RADIUS standard attribute that identifies the type of accounting request.

[Table 97 on page 276](#) lists the names and meanings assigned to AcctStatusType values 1, 2, 3, 7, and 8. Additional values for Acct-Status-Type have been defined by network access server vendors for use with their equipment; you can also use these values in the [StaticAcct] section.

Table 97: Acct-Status-Type Attribute Values

Acct-Status-Type Value	Name	Meaning
1	Start	A user session has started
2	Stop	A user session has stopped, request contains final statistics
3	Interim	A user session is in progress, request contains current statistics

Table 97: Acct-Status-Type Attribute Values (continued)

Acct-Status-Type Value	Name	Meaning
7	Accounting-On	The network access server has started
8	Accounting-Off	The network access server is about to shut down

The syntax for a [StaticAcct] section is:

```
[StaticAcct]
number=name
number=name
.
.
.
```

where each **number** is a possible value of the Acct-Status-Type attribute, and each **name** identifies a section called [**name**] that appears elsewhere in the **proxy.ini** file.

When it receives an accounting request with an Acct-Status-Type of **number**, Steel-Belted Radius Carrier uses the [StaticAcct] section to match **number** with **name**, and statically forwards a duplicate copy of the packet to all of the proxy realms listed in the [**name**] section.

Each [**name**] section consists of a list name in square brackets ([**name**]) followed by a list of proxy realms. Each of these realms must have a **RealmName.pro** file in the same directory as **proxy.ini**. Directed realms do not support static proxy accounting.

The syntax for a [**name**] section is:

```
[name]
realm1
realm2
...
```

The [**name**] section is used only if its **name** is mapped to a number in the [StaticAcct] section of the **proxy.ini** file.

The following excerpt from a **proxy.ini** file demonstrates some of the flexibility of static proxy forwarding. Copies of all session-related accounting packets (Start, Stop, and Interim) are forwarded by proxy to a realm called **billing**. Copies of all device-related accounting packets (Accounting-On and Accounting-Off) are forwarded by proxy, not only to billing, but also to a realm called **operations**.

```
[Realms]
billing
operations
```

```
[StaticAcct]
1 = SessionObserverList
2 = SessionObserverList
3 = SessionObserverList
7 = RASObserverList
8 = RASObserverList

[SessionObserverList]
realm = billing

[RASObserverList]
realm = billing
realm = operations
```

[Interfaces] Section

If your server has more than one network interface, you can assign the outgoing proxy traffic for a particular realm to a particular interface card:

1. List the IP addresses associated with each network interface card in the [Addresses] section of the **radius.ini** file.
2. Create an [Interfaces] section for the **proxy.ini** file with a list of one or more pairs in the following format:

```
[Interfaces]
InterfaceName = IPAddress
```

where **InterfaceName** is a label you assign to the given **IPAddress**.

3. Extend the existing entries in the [name] sections in **.pro** files for proxy realms with the **InterfaceName** defined in the [Interfaces] section so that they are in the following format:

```
[TargetSection]
Target=NumAttempts,InterfaceName
```

where **InterfaceName** is the name of the interface defined in the [Interfaces] section.

For example:

```
[Targets]
Bert=3,ABCInterface
Ernie=1,XYZInterface
```

NOTE: The **ProxySource** setting in the [Configuration] section of **radius.ini** disables per-realm control of proxy outbound interfaces. If **ProxySource** is not set, sockets are opened and bound for each interface on the server.

Proxyrl.ini File

The **proxyrl.ini** file supports a feature called *smart static accounting*, which lets you specify that the accounting packets for a proxy or directed realm are forwarded to a list of one or more proxy realms. These groups of realms can also be used for static accounting configured in **proxy.ini**.

Parallel proxy accounting can be used to forward accounting packets to multiple target realms simultaneously. To enable parallel proxy accounting, you must set the **Block** parameter in the **RealmName.pro** file to 0 for all the target realms. The arguments on the right **Primary**, **Wait**, and **NoWait** for the realms decide whether SBR Carrier will send Accounting-Response if forwarding to one or more target realms fails.

The **proxyrl.ini** file consists of a number of sections that you name. Each section name is referenced in the **StaticAcctRealms** parameter in the [Acct] section of a **.pro** or **.dir** file. Following the section name, you can list a number of proxy realm names, in the following format:

```
[smartAcctl]
realm1=Primary
realm2=Wait
realm3=NoWait
```

The argument:

- **Primary**—Response from the realm is required before sending an Accounting-Response to the NAD. If there is a timeout, no Accounting-Response will be sent.

NOTE: The response from the main realm will be forwarded to the NAD regardless of the **Primary** setting.

NOTE: To discard the accounting request if one or more target realms marked as “Primary” fail to respond, you must set the **SuppressResponseIfStaticAcctFails** parameter to yes in the [Configuration] section of the **proxy.ini** file.

- **Wait**—Response or timeout from the realm is required by SBR Carrier before sending an Accounting-Response to the NAD.
- **NoWait**—Neither response from the realm nor timeout is required by SBR Carrier before sending an Accounting-Response to the NAD. The default value is **NoWait**.

NOTE: Parallel proxy is not supported for directed realms (defined in **.dir** files).

NOTE: To avoid an infinite loop, the list of static accounting servers must not include realms that use the list. If you include a realm in a list of static accounting servers and specified the same realm in **proxy.ini** as doing static accounting, the realm receives duplicate accounting packets.

NOTE: When you configure the **.pro** file, you must ensure at least one system is referenced in the TargetsSection.

Proxy RADIUS Configuration (.pro) File

For each proxy realm that you want to configure on the Steel-Belted Radius Carrier server, you must create a file called **RealmName.pro**, where **RealmName** is the name of the realm, and you must add this RealmName to the [Realms] section of the **proxy.ini** file.

NOTE: If you create or edit a *RealmName.pro* file, you can apply your configuration changes dynamically, without stopping the server.

Issue the **SIGHUP (1)** signal to the *Steel-Belted Radius Carrier* process.

#./sbrd hup

After you do this, Steel-Belted Radius Carrier re-reads **proxy.ini**, **filter.ini**, and all **.pro** and **.dir** files in the server directory, and resets its realm configuration accordingly.

If you edit **radius.ini** while configuring a realm, you must stop and restart Steel-Belted Radius Carrier to load your new configuration.

[Auth] Section

The [Auth] section ([Table 98 on page 281](#)) of a *RealmName.pro* file configures authentication for the proxy realm. The key parameters in these sections are:

- **TargetsSection**, which names the target selection strategy you want to use.
- **FilterIn** and **FilterOut**, which name the attribute or subattribute filters you want applied to request and response packets, respectively.

Table 98: RealmName.pro [Auth] Syntax

Parameter	Function
AvailableAsAuthMethod	<ul style="list-style-type: none"> • If set to yes, allows the proxy realm to be used as an authentication method for a directed realm. • If set to no, the proxy realm cannot be used as an authentication method for a directed realm. <p>The default is no.</p>
Enable	<ul style="list-style-type: none"> • If set to 1, enables forwarding of authentication packets to the realm called <i>RealmName</i>. • If set to 0, the realm called <i>RealmName</i> is disabled for authentication. <p>Default value is 0.</p>

Table 98: RealmName.pro [Auth] Syntax (*continued*)

Parameter	Function
FilterOut=name	<p>The FilterOut=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [<i>name</i>] section of filter.ini. These rules are applied while Steel-Belted Radius Carrier is processing the <i>incoming</i> RADIUS request packet, and <i>before</i> it directs the packet <i>out</i> to the destination realm. You may also think of this as filtering various attributes and values <i>out</i> of the request before directing it to the realm.</p> <p>NOTE: The FilterOut setting will be applied only to the realm for which it is configured.</p>
FilterIn=name	<p>The FilterIn=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [<i>name</i>] section of filter.ini. These rules are applied <i>after</i> Steel-Belted Radius Carrier has received a response <i>in</i> from the destination realm, and while it is preparing the RADIUS response packet for its client. You may also think of this as filtering various attributes and values <i>in</i> to the response before returning it to the client.</p>
MessageAuthenticator	<p>If set to 1, a Message-Authenticator is inserted into each request forwarded to any target server in the realm.</p> <p>Default value is 0.</p> <p>NOTE: Both the proxy and the target RADIUS server require this functionality.</p>
NumAttempts	<p>The number of times a timeout may occur when attempting to contact servers within the realm, before a failure is declared and the attempts to forward the request are stopped.</p> <p>Default value is 3.</p>

Table 98: RealmName.pro [Auth] Syntax (continued)

Parameter	Function
RequestTimeout=x, y, z	<p>A list of times, in seconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on.</p> <p>The number of items in the list should not exceed the NumAttempts setting. If NumAttempts is greater, the last number listed is reused for subsequent timeouts.</p> <p>Default value is 5.</p> <p>NOTE: You can specify RequestTimeout or RequestTimeoutMills, but not both.</p>
RequestTimeoutMills=x, y, z	<p>A list of times, in milliseconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on.</p> <p>The number of items in the list should not exceed the NumAttempts setting. If NumAttempts is greater, the last number listed is reused for subsequent timeouts.</p> <p>NOTE: You can specify RequestTimeout or RequestTimeoutMills, but not both.</p>
RoundRobin	<p>Specifies the number of target servers that are participating in round-robin load balancing. The count begins from the top of the list in the [name] section identified by TargetsSection. Other listed targets are used only after the round-robin targets fail for a particular request.</p> <p>Default value is 0.</p>
StripRealm	<ul style="list-style-type: none"> • If set to 1, strip the realm name from the username before forwarding. • If set to 0, name stripping is disabled. <p>NOTE: For proxy realms, realm name stripping is disabled (StripRealm = 0) by default. If you want to enable it, you must explicitly set StripRealm to 1. The default value is different for directed realm.</p>

Table 98: RealmName.pro [Auth] Syntax (continued)

Parameter	Function
TargetsSection= <i>name</i>	<p><i>name</i> identifies a section called [<i>name</i>] that appears elsewhere in the .pro file. This section lists all the targets in a proxy realm. When it receives a request for this proxy realm, Steel-Belted Radius Carrier selects a target from this list.</p> <p>Having the TargetsSection setting available in the [Auth] and [Acct] sections permits you to name different target selection parameters for proxy RADIUS authentication and accounting.</p> <p>Default value of <i>name</i> is AuthTargets, indicating the name of the section is [AuthTargets].</p> <p>NOTE: When you configure the .pro file, you must ensure at least one system is referenced in the TargetsSection.</p>
UseMasterDictionary	<ul style="list-style-type: none"> ● If set to yes, inbound proxy responses for this realm use the primary Steel-Belted Radius Carrier dictionary when authentication attributes are filtered in. ● If set to no, proxy responses for this realm use the client-specific dictionary when authentication attributes are filtered in. <p>Default value is yes. The default value is the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p> <p>NOTE: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p>
UserConcurrency	<p>Specifies the maximum number of concurrent connections that can be maintained by users who are authenticated by this realm. Connection attempts above the limit are rejected regardless of the response from the target.</p> <p>Default value is 0, which means that users authenticated by this realm can have unlimited concurrent connections.</p>

[Acct] Section

The [Acct] section (Table 99 on page 285) of a **RealmName.pro** file configures accounting. The key parameters in these sections are:

- **TargetsSection**, which names the target selection strategy you want to use.
- **FilterIn** and **FilterOut**, which name the attribute or subattribute filters you want applied to request and response packets, respectively.

Table 99: RealmName.pro [Acct] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">• If set to 1, enables forwarding of accounting packets to the realm called RealmName.• If set to 0, the realm called RealmName is disabled for accounting. <p>Default value is 0.</p>
Block	<ul style="list-style-type: none">• If set to 0, the Steel-Belted Radius Carrier server sends an accounting acknowledgement immediately (for example, after Steel-Belted Radius Carrier records an accounting message).• If set to 1, the Steel-Belted Radius Carrier server waits for a response from the target realm before sending an accounting acknowledgement. <p>Default value is 1.</p> <p>NOTE: Set the Block parameter to 0 if your network access server is not able to deal with long acknowledgment delays to accounting requests gracefully.</p> <p>To enable parallel proxy for this realm, set the Block parameter to 0.</p>
FilterOut=name	<p>The FilterOut=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [name] section of filter.ini. These rules are applied while Steel-Belted Radius Carrier is processing the <i>incoming</i> RADIUS request packet, and <i>before</i> it directs the packet <i>out</i> to the destination realm. You may also think of this as filtering various attributes and values <i>out</i> of the request before directing it to the realm.</p> <p>NOTE: The FilterOut setting will be applied only to the realm for which it is configured.</p>

Table 99: RealmName.pro [Acct] Syntax (*continued*)

Parameter	Function
FilterIn=name	<p>The FilterIn=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [<i>name</i>] section of filter.ini. These rules are applied <i>after</i> Steel-Belted Radius Carrier has <i>received</i> a response in from the destination realm, and while it is preparing the RADIUS <i>response</i> packet for its client. You may also think of this as filtering various attributes and values <i>in</i> to the response before returning it to the client.</p>
NumAttempts	<p>Specifies the number of times a timeout may occur when attempting to contact servers within the realm, before a failure is declared and the attempts are stopped.</p> <p>Default value is 3.</p>
RecordLocally	<ul style="list-style-type: none"> • If set to 1, log the packet locally before forwarding. • If set to 0, forward the packet and do not log locally. <p>Default value is 0.</p>
RequestTimeout=x, y, z	<p>A list of times, in seconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on.</p> <p>The number of items in the list should not exceed the NumAttempts setting. If NumAttempts is greater, the last number listed is reused for subsequent timeouts.</p> <p>Default is 5 seconds.</p> <p>NOTE: You can specify RequestTimeout or RequestTimeoutMills, but not both.</p>
RequestTimeoutMills=x, y, z	<p>A list of times, in milliseconds, to wait when attempting to contact a target server before timing out. The first value is the time to wait before the first timeout, and so on.</p> <p>The number of items in the list should not exceed the NumAttempts setting. If NumAttempts is greater, the last number listed is reused for subsequent timeouts.</p> <p>NOTE: You can specify RequestTimeout or RequestTimeoutMills, but not both.</p>

Table 99: RealmName.pro [Acct] Syntax (continued)

Parameter	Function
RoundRobin	<p>Specifies the number of target servers that are participating in load balancing. The count begins from the top of the list in the [name] section identified by TargetsSection. Other listed targets are only used after the <i>round-robin</i> targets fail for a particular request.</p> <p>Default value is 0.</p>
SendAckOnProxyFailure	<p>Specifies whether or not the Steel-Belted Radius Carrier server sends an accounting acknowledgement to the NAD when a proxy accounting request is not acknowledged.</p> <ul style="list-style-type: none"> • If set to 1, the Steel-Belted Radius Carrier server sends an accounting acknowledgement to the NAD even if the proxy accounting request is not acknowledged. • If set to 0, the Steel-Belted Radius Carrier server does not send an accounting acknowledgement to the NAD if the proxy accounting request is not acknowledged. <p>The default value is 1.</p> <p>NOTE:</p> <ul style="list-style-type: none"> • If the Block parameter is set to 1 and the RecordLocally parameter is set to 0, the Steel-Belted Radius Carrier server does not send an accounting acknowledgement to the NAD regardless of the SendAckOnProxyFailure setting. • If the Block parameter is set to 0, the Steel-Belted Radius Carrier server sends an accounting acknowledgement immediately, regardless of the SendAckOnProxyFailure setting.
StaticAcctRealms	<p>If a setting is supplied for this parameter, accounting packets are forwarded to a list of realms. The setting given must be a section name defined in the proxyrl.ini file that lists the realms to which the accounting packets are forwarded.</p> <p>See “Proxyrl.ini File” on page 279.</p>

Table 99: RealmName.pro [Acct] Syntax (*continued*)

Parameter	Function
StripRealm= <i>n</i>	<ul style="list-style-type: none"> • If set to 1, strip the realm name from the username before forwarding. • If set to 0, name stripping is disabled. <p>Default value is 0.</p> <p>NOTE: For proxy realms, realm name stripping is disabled (StripRealm = 0) by default. If you want to enable it, you must explicitly set StripRealm to 1.</p>
SuppressAuxConnectionAcct	<ul style="list-style-type: none"> • Set to 1 to suppress the proxying of per-flow (auxiliary connection) accounting on a per-realm basis. • Set to 0 to enable the proxying of per-flow (auxiliary connection) accounting on a per-realm basis. <p>NOTE: Use caution when enabling this option as per-flow accounting messages may alter the information in the current sessions table (CST).</p>
TargetsSection= <i>name</i>	<p><i>name</i> identifies a section called [<i>name</i>] that appears elsewhere in the .pro file. This section lists all the targets in a proxy realm. When it receives a request for this proxy realm, Steel-Belted Radius Carrier selects a target from this list.</p> <p>Having the TargetsSection parameter available in the [Auth] and [Acct] sections permits you to name different target selection parameters for proxy RADIUS authentication and accounting.</p> <p>The default value of name is AcctTargets; in which case the name of the section is [AcctTargets].</p> <p>NOTE: When you configure the .pro file, you must ensure at least one system is referenced in the TargetsSection.</p>

Table 99: RealmName.pro [Acct] Syntax (*continued*)

Parameter	Function
UseMasterDictionary	<ul style="list-style-type: none"> • If set to yes, inbound proxy responses for this realm use the primary Steel-Belted Radius Carrier dictionary when accounting attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when accounting attributes are filtered in. <p>Default value is yes. The default value is the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p> <p>NOTE: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p>

[AutoStop] Section

The [AutoStop] section ([Table 100 on page 290](#)) of a realm configuration file permits you to activate the Proxy AutoStop feature. When this feature is enabled, an AutoStop request is automatically recorded and associated with the session in the current sessions database when the initial Accounting-Start message is received. This AutoStop message may be used later to simulate an Accounting-Stop message which is fed back into the request processing engine, causing it to be forwarded to the appropriate realms and for the normal processes of ending the user session to be enacted.

NOTE: As the AutoStop record is generated when the session begins, it is simply a duplicate of the original Start request and does not have access to information about the lifetime of the user's actual activity.

NOTE: AutoStop records are not saved on persistent storage: this means that if Steel-Belted Radius Carrier is restarted, this information is lost and hence Accounting-Stop messages cannot be simulated for these user sessions.

Table 100: RealmName.pro [AutoStop] Syntax

Parameter	Function
Enable	<p>Set to 0 to disable AutoStop for the current realm.</p> <p>Set to 1 to enable AutoStop for the current realm.</p> <p>Default value is 0.</p>

Table 101 on page 290 lists the parameters in other configuration files you must enable (set to 1) for AutoStop to operate.

Table 101: AutoStop Configuration Requirements

File	Section	Parameter
<i>RealmName.pro</i>	[Acct]	Enable
<i>RealmName.pro</i>	[Acct]	RecordLocally
radius.ini	[Configuration]	AcctAutoStopEnable

[Called-Station-ID] Section

The [Called-Station-ID] section of a *RealmName.pro* file allows the target realm to be selected based on DNIS. The [CalledStationID] section lists each DNIS string that identifies the realm. If this string is found in the CalledStationId attribute of an incoming RADIUS request, the request is assumed to be addressed to this realm.

The syntax is:

```
[Called-Station-ID]
String
String
.
.
.
```

where *String* is a DNIS string.

For example:

```
[Called-Station-ID]
8005551212
8005551213
6175551212
```

You can also use wildcards, as in the following example:

```
[Called-Station-ID]
800*
508*
```

[DynAuth] Section

The [DynAuth] section in each **realm.pro** file controls the dynamic authorization proxy configuration for each proxy realm.

Table 102: [DynAuth] Section

Parameter	Function
FilterOut	<p>If set to a valid attribute filter name, the FilterOut parameter causes SBRC to apply that filter (and optional JavaScript) when forwarding a dynamic authorization (CoA/DM) request to a NAS client.</p> <p>NOTE: The FilterOut setting will be applied only to the realm for which it is configured.</p>
IncludeDeviceModel	<p>The IncludeDeviceModel parameter can be set to yes or no.</p> <p>If set to yes, the Funk-Device-Model attribute with the appropriate device name is added to every forwarded proxy request sent to this realm.</p>
RequireMessageAuthenticator	<p>If set to a non-zero value, SBRC requires a Message-Authenticator attribute in the incoming CoA/DM requests. If this attribute is not present, then the CoA/DM request is discarded.</p> <p>Default value is 0.</p>

Target Selection Rules

Each **[name]** section of a **RealmName.pro** file specifies a set of rules that Steel-Belted Radius Carrier can use to select a target for proxy-forwarding within the proxy realm. Each **[name]** section consists of a list of target servers. For any particular request, if the first listed server fails to respond (or is presumed down),

the other servers are tried in the order listed. A **[name]** section is activated by referencing it from the **[Auth]** and **[Acct]** sections ([Table 103 on page 292](#)).

Table 103: Proxy Realm Target Selection

To activate	Use
a [name] section for authentication	TargetName=name in the [Auth] section
the same [name] section for accounting	TargetName=name in the [Acct] section
some [other] section for accounting	TargetName=other in the [Acct] section

The full syntax is:

```
[Auth]
TargetsSection=nameB
[Acct]
TargetsSection=nameA
[nameA]
Server = n
Server = n
.
.
.
[nameB]
```

where **Server** is the name of a server configured as a target for standard proxy RADIUS forwarding, and **n** is explained in the next section.

Server must match a Proxy entry in the Steel-Belted Radius Carrier database. This Proxy entry provides the address and shared secret for the target server. All other settings in the Proxy entry (retry policy, proxy accounting) are overridden by the settings that you configure in the **RealmName.pro** file.

NOTE: If your server has multiple interface cards, you may add a parameter referring to the interface to each line to order the outgoing proxy traffic for the realm through a particular interface. See “[Interfaces] Section” on page 278.

Round-Robin Load Balancing

If you have multiple target servers in a realm, you can select whether to use them in round-robin fashion (load balancing), primary/backup fashion, or a combination of both. The value of the **RoundRobin** entry in the **[Auth]** or **[Acct]** section indicates the number of targets that are to be used in round-robin fashion.

The count begins from the top of list in the **[name]** section. Other listed targets are used only if the round-robin targets fail for a particular request. If **RoundRobin** is 0 or 1, all requests are routed to the first target in the **[name]** list, assuming that it is up, the others are tried in the order listed.

If **RoundRobin** is 2 or greater (say, *n*), each request is routed to a different target server, in rotation among the first *n* listed targets. Requests are then load- balanced evenly among those targets. For any particular request, if one target fails to respond, other targets are attempted. The round-robin targets are tried first; if they all fail to respond, any additional targets are then tried in the order in which they appear in the list.

In the following example, **RoundRobin** is 3. Under normal circumstances, requests are balanced in round-robin fashion among the first three targets. The first request goes to **Bert**; the next goes to **Ernie**; the next to **George**; the next to **Bert**; the next to **Ernie**; the next to **George**; and so on. If any of these servers go down at some point, the other two are tried, in list order. The fourth target (**Mary**) receives requests only when other targets are down.

```
[Auth]
RoundRobin=3
NumAttempts=8
TargetsSection=Targets

[Targets]
Bert=1
Ernie=1
George=1
Mary=5
```

Selecting a Backup Server

If **RoundRobin** is set to 0, Steel-Belted Radius Carrier makes a selection from the other servers in the list only if the primary server is down.

For example:

```
[Auth]
RoundRobin=0
NumAttempts=8
TargetsSection=Targets

[Targets]
Bert=1
Ernie=1
```

In this case, Bert is used until there is a problem; then Ernie becomes the server of second choice.

Realm Retry Policy

Each target selection rule in the **[name]** section permits you to name a target and assign it a numeric value:

```
[name]
Server = n
Server = n
.
.
.
```

The **n** setting indicates the number of times to retry requests to this target server when no response is received within the amount of time set by **RequestTimeout** in the [Auth] or [Acct] section.

The number of attempts to all servers within the entire realm is given by the **NumAttempts** value in the [Auth] or [Acct] section. For example, the **NumAttempts** is 8 and there are three target servers, each with **n** set to 3:

```
[Auth]
NumAttempts=8
TargetsSection=Targets

[Targets]
Bert=3
Ernie=3
George=3
```

All three servers are down when a request comes into the realm. The first target (**Bert**) is tried 3 times; then the second target (**Ernie**) is tried 3 times; and the third target (**George**) is tried 2 times. At this point, the number of tries to all servers in the realm is 8, which equals **NumAttempts**. Steel-Belted Radius Carrier returns a failure response from the realm.

NOTE: A third attempt to **George** will not be made unless you edited the **RealmName.pro** file, increased **NumAttempts** to 9, and reloaded Steel-Belted Radius Carrier.

[FastFail] Section

The [FastFail] section ([Table 104 on page 295](#)) of a realm configuration file permits you to fine-tune retry policies for individual realms, and for specific targets within a realm. If you provide a [FastFail] section, the **ProxyFastFail** parameter in the **radius.ini** [Configuration] section is ignored.

Table 104: RealmName.pro [FastFail] Syntax

Parameter	Function
MinFailures=x MinSeconds=y	<p>These parameters define a tolerance level for failures to reach a target server within a realm. Failures are judged according to the NumAttempts and RequestTimeout settings that you defined in the [Auth] or [Acct] sections.</p> <p>A target is presumed <i>down</i> and is assigned that statue after x consecutive failures have occurred and at least y seconds have elapsed.</p> <p>After a target is presumed <i>down</i>, Steel-Belted Radius Carrier directs proxy requests to another target in the same realm, if available. It does not wait for responses from the failed target.</p> <p>However, it sends strobe requests periodically to the failed target to detect when that server comes back up. After a response is received to one of these strobe requests, that server is no longer presumed down. You can configure the interval for sending these strobe requests by setting the StrobeInterval parameter. In addition, you can enable or disable the sending of these strobes on a per-realm basis by setting the StrobeEnable parameter.</p>
ResetSeconds=z	<p>After the realm's tolerance level is exceeded, this parameter specifies how long a target may be presumed down.</p> <p>The ResetSeconds value indicates the maximum number of seconds during which a server can be presumed down in the absence of strobe requests. If z seconds elapse with no strobe requests sent to the down server, the server is reset to <i>up</i>.</p> <p>The status of a target that is presumed down is reset to <i>up</i> when one of the following occurs:</p> <ul style="list-style-type: none"> • A response to a strobe request is received from the server. • There has been no request sent to the server for z seconds. <p>Default value is 600 seconds.</p>

Table 104: RealmName.pro [FastFail] Syntax (*continued*)

Parameter	Function
SendStatusServer	<p>Specifies the type of strobe request sent to verify the status of the target server.</p> <ul style="list-style-type: none"> • If set to 1, a Status-Server type message is sent as a strobe request. • If set to 0, current authentication or accounting packet is sent as a strobe request based on the type of original request received. <p>Default value is 0.</p> <p>When you use the RetryCount parameter to time out spooled packets, you need to send Status-Server packets as strobe requests because some RADIUS servers may not acknowledge duplicate packets.</p>
StrobeEnable	<p>Enables strobe requests on a per-realm basis. If strobes are disabled, the target is taken out of fast-fail after ResetSeconds has elapsed.</p> <ul style="list-style-type: none"> • 0=Disabled • 1=enabled (default)
StrobeInterval	<p>Specifies the interval (in seconds) in which strobe requests are sent.</p> <p>The default is 1 second.</p>

[ModifyUser] Section

The [ModifyUser] section ([Table 105 on page 297](#)) of a realm configuration file permits you to decorate a realm, where the realm is determined by other means, such as DNIS or attribute mapping.

This is used mainly to enhance directed realms. For example, the following two users are in the database: **george@gm** and **george@ford**. Either user can log in as **george**, because Steel-Belted Radius Carrier determines the realm, for example, by DNIS. Based on the realm, Steel-Belted Radius Carrier appends either **@gm** or **@ford** to the username, and then uses the Native User directed method to authenticate.

This methodology can also be used in a double-proxy situation. The first proxy uses DNIS to determine a realm, then decorates the name and forwards it to the next hop server. This second proxy (which may be a legacy RADIUS server that does not understand DNIS) can then handle realms based on the name decoration.

Table 105: RealmName.pro [ModifyUser] Syntax

Parameter	Function
AddPrefix= <i>prefix</i>	These parameters define the User-Name prefix and suffix.
AddSuffix= <i>suffix</i>	

[SpooledAccounting] Section

Proxy spooling is configured within the [SpooledAccounting] section ([Table 106 on page 297](#)) of a **RealmName.pro** file.

```
[SpooledAccounting]
Enable=1
RolloverSeconds=600
RolloverSize=1048576
Directory=./all_acct_data
RetryInterval=60
ShutdownDelay=20
RetryCount=0
```

Table 106: RealmName.pro [SpooledAccounting] Syntax

Parameter	Function
Directory	<p>Specifies the directory where the spool (.psf) files are stored. The directory must be manually created in the RADIUS service directory.</p> <p>Default is ./RealmName</p> <p>NOTE: Each realm must have its own directory for spool files. Otherwise, packets for multiple realms are interspersed and a problem in one realm can prevent subsequent packets to other realms from being forwarded.</p>
Enable	<ul style="list-style-type: none"> • If set to 1, proxy spooling is enabled. • If set to 0, proxy spooling is disabled. <p>Default value is 0.</p>

Table 106: RealmName.pro [SpooledAccounting] Syntax (*continued*)

Parameter	Function
RetryCount	<p>Specifies the number of times a proxy request is retransmitted if an acknowledgment from the target system is not received after a successful strobe attempt. If the number of retries is exhausted, the current spooled proxy request is abandoned and the next spooled packet is sent.</p> <p>Default value is 0.</p> <p>Setting RetryCount to 0 disables the timing out of unacknowledged spooled requests.</p> <p>NOTE: The RetryCount parameter takes effect only when the StrobeEnable parameter is set to 1.</p>
RetryInterval	<p>Specifies the interval in seconds before retrying a proxy request if the target system (the downstream server where accounting data for this realm is sent) is down.</p> <p>Default value is 60.</p>
RolloverSeconds	<p>Specifies the rollover interval in seconds. After the interval elapses, the current spool file is closed and a new one is created.</p> <p>Default value is 600 (10 minutes.)</p>
RolloverSize	<p>Specifies the rollover file size limit in bytes. After the file size exceeds this limit, the current spool file is closed and a new one is created.</p> <p>If both RolloverSeconds and RolloverSize are set, the first parameter that exceeds its limit initiates rollover.</p> <p>Default value is 1,048,576 bytes (1 megabyte).</p>

Table 106: RealmName.pro [SpooledAccounting] Syntax (*continued*)

Parameter	Function
ShutdownDelay	<p>Specifies the amount of time (given as the number of seconds) before the execution of a shutdown request during which the final undelivered spooled packets in the spool file can be sent to their target. This value should reflect the amount of accounting data normally received for this realm and other network conditions that may have an impact on the delay.</p> <p>If the target system is down when Steel-Belted Radius Carrier shuts down, this setting is not applied, and unspooling terminates immediately (and Steel-Belted Radius Carrier shuts down immediately). Upon restart, unspooling of accounting data restarts from the beginning of the oldest spool file.</p> <p>Default value is 20.</p>

NOTE: A new log file, in the **yyyymmdd_Failed.act** format, containing information about records that are abandoned after the configured **RetryCount** number of retries, is created. This file includes information such as packet code, ID, length, target name, and port.

NOTE: Do not enable proxy spooling for realms that are not enabled for accounting.

Account Spooling guarantees sequential delivery of proxied accounting packets through a single-threaded mechanism. However, the use of Account Spooling can adversely affect performance in systems that may sustain heavy load and high latency, or both of proxy targets.

Retry Sequence

If Steel-Belted Radius Carrier receives an accounting packet for a realm, and the target system is down, Steel-Belted Radius Carrier implements the **RealmName.pro** retry configuration, as in the following example:

```
[Acct]
RequestTimeout=5, 3, 5
NumAttempts=3
```

In this example, Steel-Belted Radius Carrier attempts to proxy forward the accounting packet to the target IP address, as it does in a non-SpooledAccounting scenario. Three attempts are made; the first waits for 5 seconds before timing out, the second 3 seconds, and the third 5 seconds.

If there is still no response from the target after three attempts, the **RetryInterval** in the [SpooledAccounting] section is applied. If **RetryInterval** equals 60, then 5 seconds after the last unsuccessful **NumAttempts** is completed, Steel-Belted Radius Carrier waits another 60 seconds and then attempts the entire retry policy again.

NOTE: Because Account Spooling guarantees sequential delivery of proxied accounting packets through a single-threaded mechanism, its use can adversely affect performance in systems that may sustain heavy load.

Directed Realm Configuration (.dir) File

A *directed realm* specifies target methods for directed authentication and directed accounting. Its realm configuration file is called **RealmName.dir**. By default, a sample .dir file (**example.dir**) is installed with Steel-Belted Radius Carrier.

The *directed authentication* feature permits the server to bypass its Authentication Methods list and map an incoming RADIUS request to one or more specific authentication methods. Steel-Belted Radius Carrier chooses the destination method based on routing information found in the request packet. The destination methods may be any authentication methods already configured on the local Steel-Belted Radius Carrier server, regardless of how they were configured; for example, a method may have been configured using Web GUI, the LDAP configuration interface, or an .aut configuration file.

If no directed authentication method is configured, every request percolates through the same Authentication Methods list, as defined in the authentication methods listed in the **Authentication Methods** page in Web GUI. This behavior may or may not be ideal for every customer. Directed authentication lets you tailor an Authentication Methods list to a customer's needs.

Directed accounting is also possible. The destination accounting method may be the Steel-Belted Radius Carrier accounting log, an external database configured using an .acc file, or a distinct accounting log file that contains entries only for this customer.

To activate these features, you must create **RealmName.dir** files, place them in the Steel-Belted Radius Carrier directory, and list them in the [Directed] section of **proxy.ini**. Subsequently, any requests that arrive addressed to one of these realm names are processed on the local server using the instructions you provided in **proxy.ini** and the corresponding **RealmName.dir** file.

After you edit a **RealmName.dir** file, you must apply your changes. If you have added or changed:

- Any directed accounting methods, you must stop and restart the server to load your new configuration.

- Directed authentication methods in which external database (SQL or LDAP) authentication is used, you must stop and restart the server to load your new configuration.
- Directed authentication methods in which local or pass-through (Native, UNIX, or Host) authentication is used, you can apply your configuration changes dynamically, without stopping the server.

Issue the **SIGHUP (1)** signal to the Steel-Belted Radius Carrier process.

#./sbrd hup

Steel-Belted Radius Carrier re-reads **proxy.ini**, **filter.ini**, and all **.pro** and **.dir** files in the server directory, and resets its realm configuration accordingly.

NOTE: If you edit **radius.ini** while configuring a realm, you must restart Steel-Belted Radius Carrier to load your new configuration.

[Auth] Section

Directed authentication is enabled in a realm by setting the Enable parameter in the [Auth] section (Table 107 on page 301) of the corresponding **RealmName.dir** file, where *RealmName* is the name of the realm. The syntax is:

```
[Auth]
Enable = 1
StripRealm = 1
UseMasterDictionary = yes
FilterOut = name
FilterIn = name
ServerCertificate =
```

Table 107: RealmName.dir [Auth] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 1 in the [Auth] section of a RealmName.dir file, the directed authentication realm called RealmName is enabled. • If set to 0, the realm is disabled. <p>By enabling a directed authentication realm, you make it possible for Steel-Belted Radius Carrier to override the Authentication Methods list on the local server by providing an alternate list - for requests addressed to this realm only. Details of this list are provided in the [AuthMethods] section of the same RealmName.dir file.</p>

Table 107: RealmName.dir [Auth] Syntax (continued)

Parameter	Function
ServerCertificate	<p>Specifies the name of the server certificate (as mentioned under the Name column of the Server Certificates List page in Web GUI) that must be used for EAP requests received from the directed realm. The certificate specified in this parameter should have been added through the Web GUI; otherwise, EAP requests will be rejected.</p> <p>If this parameter is left blank, the default certificate configured through the Web GUI will be used for EAP authentication protocols.</p> <p>NOTE: A server certificate can be mapped to one or more directed realms.</p>
StripRealm	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier strips the realm name from the username before attempting to authenticate the user's request. • If set to 0, realm name stripping is disabled. <p>NOTE: For directed realms, realm name is enabled (StripRealm = 1) by default. If you want to disable it, you must explicitly set StripRealm to 0.</p>
UseMasterDictionary	<ul style="list-style-type: none"> • If set to yes, inbound proxy responses for this realm use the primary Steel-Belted Radius Carrier dictionary when authentication attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when authentication attributes are filtered in. <p>Default value is yes. The default value is the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p> <p>NOTE: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p>

Table 107: RealmName.dir [Auth] Syntax (continued)

Parameter	Function
FilterOut = <i>name</i>	<p>The FilterOut=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [name] section of filter.ini. These rules are applied while Steel-Belted Radius Carrier is processing the incoming RADIUS request packet, and before it directs the packet out to the destination realm. You may also think of this as filtering various attributes and values out of the request before directing it to the realm.</p> <p>NOTE: The FilterOut setting will be applied only to the realm for which it is configured.</p>
FilterIn = <i>name</i>	<p>The FilterIn=name parameter causes Steel-Belted Radius Carrier to apply the filtering rules found in the [name] section of filter.ini. These rules are applied after Steel-Belted Radius Carrier has received a response in from the destination realm, and while it is preparing the RADIUS response packet for its client. You may also think of this as filtering various attributes and values in to the response before returning it to the client.</p>

[AuthMethods] Section

If directed authentication is enabled, the [AuthMethods] section of a **RealmName.dir** file lists one or more authentication methods to be used.

The syntax is:

```
[AuthMethods]
Description
Description
.
.
.
```

where **Description** is the official name of an authentication method configured on the Steel-Belted Radius Carrier server. For example:

```
[AuthMethods]
Native User
UNIX User
UNIX Group
```

```
<InitializationString=SQL>
<InitializationString=LDAP>
```

If you want your [AuthMethods] section to reference an external authentication method, a *Description* string must match the names of that method. If you want your [AuthMethods] section to reference an external database, enter the InitializationString value from the [Bootstrap] section of the corresponding .aut file.

NOTE: There is no interaction between the settings in the **Authentication Methods** page and in **RealmName.dir** files, or between different **RealmName.dir** files. For example, if you disable the UNIX User method in the **Authentication Methods** page while it is enabled in a **RealmName.dir** file, it remains enabled in **RealmName.dir**.

[Acct] Section

Directed accounting is enabled in a realm by setting the Enable parameter in the [Acct] section (Table 108 on page 304) of the corresponding *RealmName.dir* file, where *RealmName* is the name of the realm. The syntax is:

```
[Acct]
Enable = 1
StripRealm = 0
RecordLocally = 0
UseMasterDictionary = yes
```

Table 108: RealmName.dir [Acct] Syntax

Parameter	Function
Enable	<ul style="list-style-type: none">• If set to 1 in the [Acct] section of a <i>RealmName.dir</i> file, the directed accounting realm called <i>RealmName</i> is enabled.• If set to 0, the realm is disabled. <p>By enabling a directed accounting realm, you make it possible for Steel-Belted Radius Carrier to override the normally configured accounting methods on the local server by providing an alternate list - for requests addressed to this realm only. Details of this list are provided in the [AcctMethods] section of the same <i>RealmName.dir</i> file.</p> <p>Default value is 0.</p>

Table 108: RealmName.dir [Acct] Syntax (*continued*)

Parameter	Function
RecordLocally	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier writes accounting records to its main accounting log file in addition to the accounting destinations specified in [AcctMethods]. • If set to 0, this feature is disabled. <p>Default value is 0.</p>
StaticAcctRealms	<p>If a value is supplied for this parameter, accounting packets are forwarded to a list of realms. The setting given must be a section name defined in the proxyrl.ini file that lists the realms to which the accounting packets are forwarded.</p> <p>See “Proxyrl.ini File” on page 279.</p>
StripRealm	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier strips the realm name from the username before attempting to authenticate the user's request. • If set to 0, realm name stripping is disabled. <p>NOTE: For directed realms, username stripping is enabled (StripRealm = 1) by default. If you want to disable it, you must explicitly set StripRealm to 0.</p>
UseMasterDictionary	<ul style="list-style-type: none"> • If set to yes, inbound proxy responses for this realm use the primary Steel-Belted Radius Carrier dictionary when accounting attributes are filtered in. • If set to no, proxy responses for this realm use the client-specific dictionary when accounting attributes are filtered in. <p>Default value is yes. The default value is the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p> <p>NOTE: This value overrides the global setting configured in the UseMasterDictionary parameter in the proxy.ini file.</p>

[AcctMethods] Section

If directed accounting is enabled, the [AcctMethods] section of a **RealmName.dir** file lists one or more accounting methods to be used. The syntax is:

```
[AcctMethods]
```

```
  Description
```

```
  Description
```

```
  .
```

```
  .
```

```
  .
```

where **Description** is the official name of a directed accounting method configured in the **proxy.ini** file.

[Called-Station-ID] Section

The [Called-Station-ID] section of a **RealmName.dir** file allows Steel-Belted Radius Carrier to select a realm to be used for directed authentication and accounting based on DNIS information supplied in an incoming RADIUS packet. The [CalledStationID] section lists each DNIS string that identifies the realm. If this string is found in the **CalledStationId** attribute of an incoming request, the directed authentication and accounting rules found in the corresponding **RealmName.dir** file are applied to the request.

The syntax is:

```
[Called-Station-ID]
```

```
  String
```

```
  .String
```

```
  .
```

```
  .
```

```
  .
```

where **String** is a DNIS string.

[ModifyUser] Section

The [ModifyUser] section ([Table 109 on page 307](#)) of a realm directed file permits you to decorate a realm, where the realm is determined by other means, such as DNIS or attribute mapping.

This is used mainly to enhance directed realms. For example, the following two users are in the database: **george@gm** and **george@ford**. Either user can log in as **george**, because Steel-Belted Radius Carrier determines the realm, for example, by DNIS. Based on the realm, Steel-Belted Radius Carrier appends either **@gm** or **@ford** to the username, and then uses the Native User directed method to authenticate.

Table 109: RealmName.dir [ModifyUser] Syntax

Parameter	Function
AddPrefix= <i>prefix</i>	These parameters define the User-Name prefix and suffix.
AddSuffix= <i>suffix</i>	

radius.ini Realm Settings

The [Self] section of **radius.ini** lets you list all of the realm names that are handled by this Steel-Belted Radius Carrier server, rather than being proxied to other targets.

The [Configuration] section of **radius.ini** provides parameter that you can use to enable or disable realm features for the Steel-Belted Radius Carrier server: AttributeEdit. This parameter is enabled (set to 1) by default. You can disable the feature by setting the parameter to 0.

NOTE: If you edit **radius.ini** while configuring a realm, you must stop and restart the Steel-Belted Radius Carrier server to load your new realm configuration.

For more details about the **radius.ini** file sections, see [“\[Self\] Section” on page 95](#) and [“\[Configuration\] Section” on page 29](#).

EAP Configuration Files

IN THIS CHAPTER

- [eap.ini File | 308](#)
- [peapauth.aut File | 312](#)
- [tlsauth.aut File | 322](#)
- [tlsauth.eap File | 334](#)
- [ttlsauth.aut File | 348](#)

This chapter describes the EAP configuration and helper files, which specify options for automatic EAP helper methods. These files are loaded at startup time and reside in the Steel-Belted Radius Carrier (SBRC) directory.

NOTE:

The SIM plug-ins uses EAP, but not the configuration files explained here. For more information, see [“SIM Authentication Module” on page 491](#).

The following topics are included in this chapter:

All files are in the Steel-Belted Radius Carrier directory and are loaded at startup.

eap.ini File

The **eap.ini** configuration file controls the sequence in which EAP authentication types are tried when authenticating users by means of the different Steel-Belted Radius Carrier authentication methods.

NOTE: Use the Web GUI to maintain settings in the **eap.ini** file. Do not edit the **eap.ini** file manually.

Each authentication method that you want EAP authentication to be performed against must be configured within this **eap.ini** file.

This file must contain one section for each authentication method that you use, and the title of the section must identify the authentication method:

- | | |
|---|---|
| <ul style="list-style-type: none">• Native User• LDAP• SQL• SQL-ORACLE | <ul style="list-style-type: none">• EAP-TLS• EAP-TTLS• EAP-PEAP• EAP-MD5-Challenge• EAP-MS-CHAP-V2• defaultMethods |
|---|---|

```
[Native-User]
EAP-Only = 0
First-Handle-Via-Auto-EAP = 0
EAP-Type = TTLS, MD5-Challenge
Available-EAP-Types=MD5-Challenge,MS-CHAP-V2,TLS
Available-EAP-Only-Values=0,1
Available-Auto-EAP-Values=1
```

NOTE: Steel-Belted Radius Carrier is configured with an **eap.ini** file that works for most environments.

Table 110 on page 310 lists the parameters in each section.

Table 110: eap.ini Syntax

Parameter	Function
EAP-Only	<ul style="list-style-type: none"> • If set to 0, the authentication method accepts all types of user credentials. • If set to 1, the authentication method is given only EAP credentials or acts only as a back-end server to an automatic EAP protocol method. <p>For authentication methods expected to handle EAP-TTLS inner authentications, this parameter must be set to 0 or 1 depending on the type of credentials used in the inner authentication.</p> <p>NOTE: If you are using a third party authentication service with PEAP, set this value to 0. Since the PEAP plug-in converts the inner EAP credentials to PAP for security reasons, setting this value to 1 causes third party authentication processing to be skipped when using EAP, ultimately leading to the user being rejected.</p>
EAP-Type	<p>A comma-separated list of the EAP protocols to support for this authentication method. The first protocol in the list is the primary protocol. Protocols that appear later in the list are used with this authentication method only if the client responds with an EAP NAK and specifies such a protocol or if another authentication method triggers the use of the protocol but cannot complete the request.</p> <p>Valid values include the following:</p> <ul style="list-style-type: none"> • MD5-Challenge • TTLS • TLS • MS-CHAP-V2 <p>Leave the EAP-Type list empty to disable EAP for this authentication method.</p>

Table 110: eap.ini Syntax (continued)

Parameter	Function
First-Handle-Via-Auto-EAP	<ul style="list-style-type: none"> • If set to 1 and the user credentials are EAP, an appropriate automatic EAP helper method is called before the authentication method. The purpose of calling the automatic EAP helper method is to convert the user's EAP credentials into a format acceptable to the authentication method. • If set to 0, the authentication method itself handles the request directly, before any automatic helper methods. <p>Default varies based on type of user. Refer to the comments in the eap.ini file for more information.</p>
Available-EAP-Types	<p>A comma-separated list of the EAP protocols that can be selected when configuring the Steel-Belted Radius Carrier server by means of the Web GUI.</p> <p>Valid values include the following:</p> <ul style="list-style-type: none"> • TTLS • TLS • MS-CHAP-V2 • MD5-Challenge
Available-EAP-Only-Values	<p>Controls whether the Use EAP authentication only check box in the EAP Setup dialog (accessed through the Authentication Methods page in Web GUI) is enabled. Network administrators can use this parameter to control whether Web GUI users can select EAP authentication options.</p> <ul style="list-style-type: none"> • If set to 0,1, users can enable or disable the Use EAP Authentication Only check box. • If set to 0, the Use EAP Authentication Only option is disabled and the check box is inactive. • If set to 1, the Use EAP Authentication Only option is enabled and the check box is inactive. <p>Default varies based on type of user. Refer to the comments in the eap.ini file for more information.</p>

Table 110: eap.ini Syntax (*continued*)

Parameter	Function
Available-Auto-EAP-Values	<p>Controls whether the Handle via Auto-EAP First check box in the EAP Setup window (accessed through the Authentication Methods page in Web GUI) is enabled. Network administrators can use this parameter to control whether Web GUI users can select auto-EAP options.</p> <ul style="list-style-type: none"> • If set to 0,1, users can enable or disable the Handle via Auto-EAP First check box. • If set to 0, the Handle via Auto-EAP First option is disabled and the check box is inactive. • If set to 1, the Handle via Auto-EAP First option is enabled and the check box is inactive. <p>Default varies based on type of user. Refer to the comments in the eap.ini file for more information.</p>

peapauth.aut File

Settings for the EAP-PEAP plug-in are stored in the **peapauth.aut** file. The **peapauth.aut** configuration file is read each time the Steel-Belted Radius Carrier server receives a SIGHUP (1) signal.

NOTE: Use the Web GUI to maintain settings in the **peapauth.aut** file. Do not edit the **peapauth.aut** file manually.

[Bootstrap] Section

The [Bootstrap] section of the **peapauth.aut** file ([Table 111 on page 312](#)) specifies information that Steel-Belted Radius Carrier uses to load the EAP-PEAP authentication method.

Table 111: peapauth.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the executable binary that implements the EAP-PEAP module. Default value is peapauth.so .

Table 111: peapauth.aut [Bootstrap] Syntax (*continued*)

Parameter	Function
Enable	<p>Specifies whether the EAP-PEAP authentication module is enabled.</p> <ul style="list-style-type: none"> • If set to 0, EAP-PEAP is disabled. • If set to 1, EAP-PEAP is enabled. <p>Default value is 0.</p>
InitializationString	<p>Specifies the name of the EAP-PEAP.</p> <p>The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, the InitializationString value in each file must specify a unique method name.</p> <p>Default value is EAP-PEAP.</p>

[Server_Settings] Section

The [Server_Settings] section ([Table 113 on page 315](#)) lets you configure the basic operation of the EAP-PEAP plug-in.

Cipher_Suites Parameter

The **Cipher_Suites** parameter defined in the **peapauth.aut** [Server_Settings] section, specifies the cipher suites (in order of preference) that the server uses for EAP-PEAP. [Table 112 on page 313](#) lists the tested cipher suites and their TLS protocol versions.

Table 112: Tested Cipher Suites

Tested Cipher Suites	TLS Protocol Version
0xC013	TLS 1.0
0xC014	TLS 1.0
0x003C	TLS 1.2
0x003D	TLS 1.2
0x0067	TLS 1.2
0x006B	TLS 1.2
0x009C	TLS 1.2

Table 112: Tested Cipher Suites (*continued*)

Tested Cipher Suites	TLS Protocol Version
0x009D	TLS 1.2
0x009E	TLS 1.2
0x009F	TLS 1.2
0xC027	TLS 1.2
0xC028	TLS 1.2
0xC02F	TLS 1.2
0xC030	TLS 1.2

NOTE: SBR Carrier does not provide support for TLSv1.3.

SBR Carrier supports the following weak cipher suites: 0x002F, 0x0033, 0x0035, 0x0039, 0x003C, and 0x003D. These weak ciphers are not supported by default and need to be defined in the **Cipher_Suites** parameter.

When SBR Carrier receives a PEAP message, it compares the cipher suites in the client message to the cipher suites defined in this parameter. A match is selected based on both type (for example DSS) and order of preference defined in the client cipher suite list. If no match is found, SBR Carrier returns a handshake failure alert and closes the connection. Following are several examples of the cipher suite selection process:

Example 1

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

**0xC00A,0xC014,0xC019,0xC009,0xC013,
0x00AF,0x00B9,0xC035,0xC09A**

Client cipher suite list:

**0x0040,0x0033,0x0032,0x0016,0x0013,0x0066,
0x0035,0x002f,0x0015,0x0012,0x000a,0x0005,0xC014**

Match found: **0xC014**

In this example SBR Carrier selects 0xC014 because it is the first algorithm listed in the client cipher suite list that is also listed in the SBR Carrier cipher suite list, and because the type is also a match.

Example 2

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

0xC00A,0xC014,0xC019

Client cipher suite list:

**0x0039,0x0033,0x0032,0x0016,0x0013,0x0066,0x0035,
0x002f,0x0015,0x0012,0x000a,0x0005**

Match found: No match found, results in handshake failure.

Table 113: peapauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	<p>Set to the maximum size TLS message length that may be generated during each iteration of the TLS exchange.</p> <p>Some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers).</p> <p>The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting.</p> <p>Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips.</p> <p>The minimum value is 500.</p>
Return_MPPE_Keys	<p>Setting this attribute to 1 causes the module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption.</p> <p>If the Access Point is authenticating only end users and WEP is not being used, this attribute may be set to 0.</p> <p>Default value is 1.</p>

Table 113: peapauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Challenge_Timeout	<p>This parameter defines the timeout (in seconds) for a particular challenge request.</p> <p>Minimum value for the parameter is 1 second.</p> <p>Maximum value should be less than or equal to the value specified in the Max_Transactions_Seconds parameter.</p> <p>Default value is 30.</p>
Max_Transaction_Seconds	<p>This parameter defines the maximum timeout (in seconds) for a transaction.</p> <p>Minimum value for the parameter is 1 second.</p> <p>Maximum value for the parameter is 3600 seconds.</p> <p>Default value is 120.</p>
TLS_Protocol_Version	<p>Specifies the TLS protocol version on which the server expects the client to initiate the handshake process. The value can be one of the following:</p> <ul style="list-style-type: none"> • 31—TLS protocol version 1.0 • 32—TLS protocol version 1.1 • 33—TLS protocol version 1.2 <p>Default value is 31.</p> <p>If you set a value other than 31, 32, or 33, then the default TLS protocol version 1.0 (31) is considered.</p>
DH_Prime_Bits	<p>Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation.</p> <p>Valid values are 512, 1024, 1536, 2048, 3072, and 4096.</p> <p>Default value is 1024.</p>

Table 113: peapauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Cipher_Suites	<p>Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, <i>The TLS Protocol Version 1</i>, RFC 4346, <i>The TLS Protocol Version 1.1</i>, and RFC 5246, <i>The TLS Protocol Version 1.2</i>.</p> <p>Default value is: 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013.</p> <p>See Table 112 on page 313 for the list of tested cipher suites and their TLS protocol versions.</p> <p>For more information see “Cipher_Suites Parameter” on page 313.</p>
PEAP_Min_Version	<p>Specifies the minimum version of the PEAP protocol that the server negotiates:</p> <ul style="list-style-type: none"> • If set to 0, the server negotiates version 0. • If set to 1, the server negotiates version 1. <p>Default value is 0.</p> <p>NOTE: The value entered in this setting must be less than or equal to the value entered for the PEAP_Max_Version setting.</p>
PEAP_Max_Version	<p>Specifies the maximum version of the PEAP protocol that the server negotiates:</p> <ul style="list-style-type: none"> • If set to 0, the server negotiates version 0. • If set to 1, the server negotiates version 1. <p>Default value is 1.</p> <p>NOTE: The value entered in this parameter must be equal to or greater than the value entered for PEAP_Min_Version.</p>

[Inner_Authentication] Section

The [Inner_Authentication] section ([Table 114 on page 318](#)) lets you specify the way in which the inner authentication step is to operate.

Table 114: peapauth.aut [Inner_Authentication] Syntax

Parameter	Function
Directed_Realm	<p>Omitting this setting causes the inner authentication request to be handled like any other request received from a RAS.</p> <p>Specifying the name of a directed realm causes the request to be routed based on the methods listed in the directed realm.</p> <p>Default is to process the inner authentication through standard request processing.</p>

NOTE: The filters named in these settings must be defined in the **filter.ini** file.

[Request Filters] Section

Request filters ([Table 115 on page 318](#)) affect the attributes of inner authentication requests.

Table 115: peapauth.aut [Request Filters] Syntax

Parameter	Function
Transfer_Outer_Attrbs_to_New	<p>This filter affects only a new inner authentication request (rather than continuations of previous requests).</p> <p>If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request.</p> <p>If this filter is not specified, no attributes from the outer request are transferred to the inner request.</p>

Table 115: peapauth.aut [Request Filters] Syntax (*continued*)

Parameter	Function
Transfer_Outer_Attribs_to_Continue	<p>This filter affects only a continued inner authentication request (rather than the first inner authentication request).</p> <p>If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request.</p> <p>If this filter is not specified, no attributes from the outer request are transferred to the inner request.</p>
Edit_New	<p>This filter affects only a new inner authentication request (rather than continuations of previous requests).</p> <p>If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_New in this table) and attributes included in the inner authentication request sent through the tunnel by the client.</p> <p>If this filter is not specified, the request remains unaltered.</p>
Edit_Continue	<p>This filter affects only a continued inner authentication request (rather than a new inner authentication request).</p> <p>If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_Continue in this table) and attributes included in the inner authentication request sent through the tunnel by the client.</p> <p>If this filter is not specified, the request remains unaltered.</p>

NOTE: The filters named in these settings must be defined in the **filter.ini** file.

[Response Filters] Section

Response filters ([Table 116 on page 320](#)) affect the attributes in the responses returned to authentication requests

Table 116: peapauth.aut [Response Filters] Syntax

Parameter	Function
Transfer_Inner_Attribs_To_Accept	<p>This filter affects only an outer Access-Accept response that is sent back to a network access server.</p> <p>If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response.</p> <p>If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.</p>
Transfer_Inner_Attribs_To_Reject	<p>This filter affects only an outer Access-Reject response that is sent back to a network access server.</p> <p>If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response.</p> <p>If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.</p>

NOTE: The filters named in these settings must be defined in the **filter.ini** file.

[Session_Resumption] Section

The [Session_Resumption] section ([Table 117 on page 321](#)) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

NOTE: For session resumption to work, the network access server must be configured to handle the Session-Timeout return list attribute, because the network access server must be able to tell the client to reauthenticate after the session timer has expired.

Table 117: peapauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	<p>Set this attribute to the maximum number of seconds you want the client to remain connected to the network access server before having to reauthenticate.</p> <ul style="list-style-type: none"> • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. Default value is 0. <p>Entering a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. You can configure the resumption limit to make most reauthentications fast and computationally cheap.</p>
Termination_Action	<p>Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are:</p> <ul style="list-style-type: none"> • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. <p>Default value is -1. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.</p>
Resumption_Limit	<p>Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature.</p> <p>This type of reauthentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature.</p> <p>Default value is 0.</p>

tlsauth.aut File

NOTE: Use the Web GUI to maintain settings in the **tlsauth.aut** file. Do not edit the **tlsauth.aut** file manually.

Settings for the EAP-TLS authentication method are stored in the **tlsauth.aut** file. The **tlsauth.aut** configuration file is read each time the Steel-Belted Radius Carrier server receives a SIGHUP (1) signal.

[Server_Settings] Section

The [Server_Settings] section contains the settings that control the basic operation of the EAP-TLS authentication method.

Cipher_Suites Parameter

The **Cipher_Suites** parameter defined in the **tlsauth.aut** [Server_Settings] section, specifies the cipher suites (in order of preference) that the server uses for EAP-TLS. When SBR Carrier receives a TLS message, it compares the cipher suites in the client message to the cipher suites defined in this parameter. A match is selected based on both type (for example DSS) and order of preference defined in the client cipher suite list. If no match is found, SBR Carrier returns a handshake failure alert and closes the connection. Following are several examples of the cipher suite selection process:

Example 1

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

**0xC00A,0xC014,0xC019,0xC009,0xC013,
0x00AF,0x00B9,0xC035,0xC09A**

Client cipher suite list:

**0x0040,0x0033,0x0032,0x0016,0x0013,0x0066,
0x0035,0x002f,0x0015,0x0012,0x000a,0x0005,0xC014**

Match found: **0xC014**

In this example SBR Carrier selects 0xC014 because it is the first algorithm listed in the client cipher suite list that is also listed in the SBR Carrier cipher suite list, and because the type is also a match.

Example 2

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

0xC00A,0xC014,0xC019

Client cipher suite list:

**0x0039,0x0033,0x0032,0x0016,0x0013,0x0066,0x0035,
0x002f,0x0015,0x0012,0x000a,0x0005**

Match found: No match found, results in handshake failure.

Table 118: tlsauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	<p>Maximum TLS message length that may be generated during each iteration of the TLS exchange. Anecdotal evidence suggests that some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers).</p> <p>The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting.</p> <p>Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips.</p> <p>The minimum value is 500.</p>

Table 118: tlsauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Verify_User_Name_Is_Principal_Name	<p>Certificates issued by Microsoft's Windows 2000 Certificate Server typically include a Subject Alternative Name/Other Name attribute, where Principal Name is set to something like user@certtest.acme.com.</p> <p>The Windows XP client that supports EAP-TLS in conjunction with 802.1X extracts this attribute value from the client's certificate and uses it to respond to the Access Point's EAP Identity Request. The Access Point, in turn, packages up this value as the RADIUS User-Name attribute in requests it sends to a RADIUS server.</p> <ul style="list-style-type: none"> • If set to 1, the EAP-TLS module verifies that the contents of the RADIUS User-Name attribute match the 'Principal Name' of the certificate used to authenticate the user. • If set to 0, no such check is performed. Set the value to 0 if the certificates do not include a 'Principal Name' or if the client being used does not report the contents of 'Principal Name' as the user's identity in response to an EAP Identity Request. <p>Default value is 0.</p>
Return_MPPE_Keys	<p>Setting this attribute to 1 causes the EAP-TLS module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end users and WEP is not being used, this attribute may be set to 0.</p> <p>Default value is 1.</p>

Table 118: `tlsauth.aut` [Server_Settings] Syntax (*continued*)

Parameter	Function
DH_Prime_Bits	<p>Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation.</p> <p>Valid values are 512, 1024, 1536, 2048, 3072, and 4096.</p> <p>Default value is 1024.</p>
TLS_Protocol_Version	<p>Specifies the TLS protocol version on which the server expects the client to initiate the handshake process. The value can be one of the following:</p> <ul style="list-style-type: none"> • 31—TLS protocol version 1.0 • 32—TLS protocol version 1.1 • 33—TLS protocol version 1.2 <p>Default value is 31.</p> <p>If you set a value other than 31, 32, or 33, then the default TLS protocol version 1.0 (31) is considered.</p>
Challenge_Timeout	<p>This parameter defines the timeout (in seconds) for a particular challenge request.</p> <p>Minimum value for the parameter is 1 second.</p> <p>Maximum value should be less than or equal to the value specified in the Max_Transactions_Seconds parameter.</p> <p>Default value is 30.</p>
Max_Transaction_Seconds	<p>This parameter defines the maximum timeout (in seconds) for a transaction.</p> <p>Minimum value for the parameter is 1 second.</p> <p>Maximum value for the parameter is 3600 seconds.</p> <p>Default value is 120.</p>

Table 118: tlsauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Cipher_Suites	<p>Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, <i>The TLS Protocol Version 1</i>, RFC 4346, <i>The TLS Protocol Version 1.1</i>, and RFC 5246, <i>The TLS Protocol Version 1.2</i>.</p> <p>Default value is: 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013.</p> <p>See Table 112 on page 313 for the list of tested cipher suites and their TLS protocol versions.</p> <p>For more information see “Cipher_Suites Parameter” on page 322.</p>
Profile	<p>Specifies a profile that is to be used to select attributes sent back on an Access-Accept.</p> <p>By default, additional attributes are not sent back.</p>
Verify_Client_Certificate_Published	<p>Specifies that the EAP-TLS module checks that the client certificate is published in Active Directory for account users.</p> <p>Default value is 0 (disabled).</p>

[CRL_Checking] Section

The [CRL_Checking] section ([Table 119 on page 326](#)) lets you specify settings that control how Steel-Belted Radius Carrier performs certificate revocation list (CRL) checking.

Table 119: tlsauth.aut [CRL_Checking] Syntax

Parameter	Function
Enable	<p>Specifies whether CRL checking is enabled.</p> <p>Default value is 0 (disabled).</p>

Table 119: tlsauth.aut [CRL_Checking] Syntax (*continued*)

Parameter	Function
Retrieval_Timeout	<p>Specifies the time (in seconds) that EAP-TLS waits for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request is rejected.</p> <p>Default value is 5 seconds.</p>
Expiration_Grace_Period	<p>Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TLS always attempts to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache.</p> <ul style="list-style-type: none"> • If set to 0 (strict expiration mode), EAP-TLS does not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TLS considers the expired CRL an acceptable stand-in from the time the CRL expires to the time the grace period ends. <p>Default value is 0 (strict expiration mode).</p>
Allow_Missing_CDP_Attribute	<p>Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS does not know how to retrieve a CRL and cannot perform a revocation check on the certificate.</p> <ul style="list-style-type: none"> • If set to false, EAP-TLS does not accept a CRL with a missing CDP attribute. • If set to true, EAP-TLS allows such certificates and skips CRL checking for them. <p>Default value is true.</p>

Table 119: `tlsauth.aut` [CRL_Checking] Syntax (continued)

Parameter	Function
Default_LDAP_Server_Name	<p>Specifies what LDAP server name to use if the CDP contains a value that begins with the string <code>//ldap:\\</code>. This style of CDP (generated by some CAs) does not include the identity of the LDAP server.</p> <p>Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you do not specify a server name and such certificates are encountered, the CRL retrieval fails.</p>
Enable_CRL_Cache_Timeout	<p>Specifies whether CRL cache timeout is enabled. Valid values are:</p> <ul style="list-style-type: none"> • If set to 0, the CRL is refreshed whenever it expires. • If set to 1, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in the CRL_Cache_Timeout_period parameter or when the scheduled CRL expiration time occurs, whichever comes first. <p>After a CRL has expired (because its scheduled expiration time has passed or because the CRL cache has timed out), Steel-Belted Radius Carrier uses the expiration grace period to determine whether to use the current CRL.</p>

Table 119: `tlsauth.aut` [CRL_Checking] Syntax (*continued*)

Parameter	Function
<code>CRL_Cache_Timeout_Period</code>	<p>Specifies the maximum age, in hours, that a CRL can exist in the cache before it begins to expire.</p> <ul style="list-style-type: none"> • If you enter 0, Steel-Belted Radius Carrier always regards the CRL in the cache as expired and downloads a new CRL every time it receives a client certificate request. • If you enter a number greater than 0, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in this parameter or when the scheduled CRL expiration time occurs, whichever comes first. <p>NOTE: You must set <code>Enable_CRL_Cache_Timeout</code> to 1 or the <code>CRL_Cache_Timeout_Period</code> parameter is ignored.</p>
<code>LDAP_Bind_Version</code>	<p>Enables the selection of the LDAP protocol when binding to an LDAP server (2 or 3)</p> <p>The default is 2 (LDAP version 2)</p>

[Session_Resumption] Section

The [Session_Resumption] section ([Table 120 on page 330](#)) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

NOTE: For session resumption to work, the network access server must be configured to handle the Session-Timeout return list attribute, because the network access server must be able to tell the client to reauthenticate after the session timer has expired.

Table 120: tlsauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	<p>Set this attribute to the maximum number of seconds you want the client to remain connected to the network access server before having to reauthenticate.</p> <ul style="list-style-type: none"> • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. <p>Default value is 0.</p> <p>Entering a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. You can configure the resumption limit to make most reauthentications fast and computationally cheap.</p>
Termination_Action	<p>Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are:</p> <ul style="list-style-type: none"> • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. <p>Default value is -1. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.</p>
Resumption_Limit	<p>Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature.</p> <p>This type of reauthentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature.</p> <p>Default value is 0.</p>

Sample tlsauth.aut File

```

You must set Enable_CRL_Cache_Timeout to 1
or the CRL_Cache_Timeout_Period parameter is
ignored.

[Server_Settings]
; Note that all trusted root certificates
; must have a .der file extension and
; must be placed in the ROOT directory
; immediately below the directory
; containing the SBR 'radius' daemon and
; the radius.ini file.

; Indicates the maximum TLS Message fragment
; length EAP-TLS handles. If not
; specified, this parameter defaults to 1020.
; It can be set as high as 4096,
; but sizes over 1400 bytes are likely to cause
; fragmentation of the UDP packet
; carrying the message and some RADIUS client
; may be incapable of dealing with
; this fragmentation.
;TLS_Message_Fragment_Length = 1020

; Indicates whether or not the EAP-TLS module
; it to check whether the User Name
; provided in the RADIUS request matches the
; principal name in the client's
; certificate. The default is not to perform
; this check.
;Verify_User_Name_Is_Principal_Name = 0

; Indicates whether or not the EAP-TLS module
; should return the
; MS-MPPE-Send-Key and MS-MPPE-Recv-Key
; attribute upon successfully
; authenticating the user. The default is
; to return these attributes.
;Return_MPPE_Keys = 1

; Specifies the size of the prime to use
; for DH modular exponentiation. The
; choices are 512, 1024, 1536, 2048, 3072

```

```

; and 4096. The default is 1024 bits.
;DH_Prime_Bits = 1024

; Specifies the TLS cipher suites that the server is to use. These cipher suites
; are documented in RFC 2246 and other TLS related RFCs or draft RFCs.
;Cipher_Suites = 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013

; Specifies the TLS Protocol Version on which the server expects client to
; initiate the handshake process. Allowed values are 31, 32 and 33.
;TLS_Protocol_Version = 31

; Specifies a profile that is to be used
; to select attributes sent back on an
; Access-Accept. The default is not to send
; any additional attributes.
; Profile =<profile-name>

[CRL_Checking]
; Specifies whether CRL checking is to be enabled.
; The default is to disable CRL checking.
; Enable = 0

; Specifies the time (in seconds) that EAP-TLS
; waits for a CRL checking
; transaction to complete when the CRL check
; involves a CRL retrieval. When
; CRL retrieval takes longer than the
; specified time, the user's authentication
; request results in a reject. The
; default value is 5 seconds.
; Retrieval_Timeout = 5

; Specifies the time (in seconds) after
; expiration during which a CRL is
; still considered acceptable. EAP-TLS
; always attempts to retrieve a
; new CRL when it is presented with a
; certificate chain and it finds an
; expired CRL in its cache. EAP-TLS
; considers the expired CRL as an
; acceptable stand-in from the time the
; CRL expires to the time the grace
; period ends.
; Expiration_Grace_Period = 0

```

```

; Specifies whether the omission of a
; CDP attribute in a non-root certificate
; is acceptable. Without a CDP attribute,
; EAP-TLS does not know where to
; retrieve a CRL from and is not
; able to perform a revocation check on
; the certificate. The default is allow
; such certificates and to skip CRL
; checking for them.
; Allow_Missing_CDP_Attribute = 1

; Specifies what LDAP server name to
; use if the CDP contains a value that
; begins with the string "//ldap:\\\\".
; This style of CDP (generated by some
; CAs does not include the identity of
; the LDAP server. Specify the name of
; the LDAP that contains the CRLs if you
; expect to encounter certificates
; with this style CDP. If you don't specify
; a server name and such certificates
; are encountered, the CRL retrieval fails.
; Default_LDAP_Server_Name =

[Session_Resumption]
; Specifies the maximum length of time (in seconds)
; the RAS/AP is
; instructed to allow the session to persist
; before the client is asked
; to reauthenticate. Specifying a 0
; causes the Session-Timeout attribute
; not to be generated by the plug-in. The default is 0.
;Session_Timeout = 0

; Specifies the value to return for the
; Termination-Action attribute
; sent in an accepted client. If omitted in
; this file, the Termination-Action
; attribute is not sent.
Termination_Action = 0

; Specifies the length of time (in seconds)
; during which an authentication

```

```

; request that seeks to resume a previous TLS
; session is considered
; acceptable. Specifying 0 causes session
; resumption support to be
; disabled. The default is 0.
Resumption_Limit = 3600

```

tlsauth.eap File

NOTE: Use the Web GUI to maintain settings in the **tlsauth.eap** file. Do not edit the **tlsauth.eap** file manually.

Settings for the EAP-TLS automatic EAP helper are stored in the **tlsauth.eap** file. The **tlsauth.eap** configuration file is read each time the Steel-Belted Radius Carrier server receives a SIGHUP (1) signal.

[Server_Settings] Section

The [Server_Settings] section ([Table 121 on page 335](#)) contains the settings that control the basic operation of the EAP-TLS authentication process.

Cipher_Suites Parameter

The **Cipher_Suites** parameter defined in the **tlsauth.eap** [Server_Settings] section, specifies the cipher suites (in order of preference) that the server uses for the EAP-TLS automatic EAP helper. When SBR Carrier receives a message for the EAP-TLS automatic EAP helper, it compares the cipher suites in the client message to the cipher suites defined in this parameter. A match is selected based on both type (for example DSS) and order of preference defined in the client cipher suite list. If no match is found, SBR Carrier returns a handshake failure alert and closes the connection. Following are several examples of the cipher suite selection process:

Example 1

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

**0xC00A,0xC014,0xC019,0xC009,0xC013,
0x00AF,0x00B9,0xC035,0xC09A**

Client cipher suite list:

**0x0040,0x0033,0x0032,0x0016,0x0013,0x0066,
0x0035,0x002f,0x0015,0x0012,0x000a,0x0005,0xC014**

Match found: **0xC014**

In this example SBR Carrier selects 0xC014 because it is the first algorithm listed in the client cipher suite list that is also listed in the SBR Carrier cipher suite list, and because the type is also a match.

Example 2

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

0xC00A,0xC014,0xC019

Client cipher suite list:

0x0039,0x0033,0x0032,0x0016,0x0013,0x0066,0x0035,
0x002f,0x0015,0x0012,0x000a,0x0005

Match found: No match found, results in handshake failure.

Table 121: tlsauth.eap [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	<p>Maximum TLS message length that may be generated during each iteration of the TLS exchange. Anecdotal evidence suggests that some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers).</p> <p>The default value (1020) prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame. This is likely to be the safest setting.</p> <p>Setting a smaller value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. While a value of 1400 may result in 6 round-trips, a value of 500 may result in 15 round-trips.</p> <p>The minimum value is 500.</p>

Table 121: tlsauth.eap [Server_Settings] Syntax (*continued*)

Parameter	Function
Verify_User_Name_Is_Principal_Name	<p>Certificates issued by Microsoft's Windows 2000 Certificate Server usually include a Subject Alternative Name/Other Name attribute, where Principal Name set to something like user@certtest.acme.com.</p> <p>The MS Windows XP client that supports EAP-TLS in conjunction with 802.1X extracts this attribute value from the client's certificate and uses it to respond to the Access Point's EAP Identity Request. The Access Point, in turn, packages up this value as the RADIUS User-Name attribute in requests it sends to a RADIUS server.</p> <ul style="list-style-type: none"> • If set to 1, the EAP-TLS module verifies that the contents of the RADIUS User-Name attribute match the 'Principal Name' of the certificate used to authenticate the user. • If set to 0, no such check is performed. The value is set to 0 if the certificates used do not include a 'Principal Name' or if the client being used does not report the contents of 'Principal Name' as the user's identity in response to an EAP Identity Request. <p>Default value is 0.</p>
Return_MPPE_Keys	<p>Setting this attribute to 1 causes the EAP-TLS module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption. If the Access Point is authenticating only end users and WEP is not being used, this attribute may be set to 0.</p> <p>Default value is 1.</p>

Table 121: tlsauth.eap [Server_Settings] Syntax (*continued*)

Parameter	Function
TLS_Protocol_Version	<p>Specifies the TLS protocol version on which the server expects the client to initiate the handshake process. The value can be one of the following:</p> <ul style="list-style-type: none"> • 31—TLS protocol version 1.0 • 32—TLS protocol version 1.1 • 33—TLS protocol version 1.2 <p>Default value is 31.</p> <p>If you set a value other than 31, 32, or 33, then the default TLS protocol version 1.0 (31) is considered.</p>
DH_Prime_Bits	<p>Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation.</p> <p>Valid values are 512, 1024, 1536, 2048, 3072, and 4096.</p> <p>Default value is 1024.</p>
Cipher_Suites	<p>Specifies the TLS cipher suites (in order of preference) that the server is to use. These cipher suites are documented in RFC 2246, <i>The TLS Protocol Version 1</i>, RFC 4346, <i>The TLS Protocol Version 1.1</i>, and RFC 5246, <i>The TLS Protocol Version 1.2</i>.</p> <p>Default value is: 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013.</p> <p>See Table 112 on page 313 for the list of tested cipher suites and their TLS protocol versions.</p> <p>For more information see, “Cipher_Suites Parameter” on page 334.</p>

[Secondary_Authorization] Section

The [Secondary_Authorization] section lets you specify whether secondary authorization is performed and, if it is, what information is used in the secondary authorization request.

Table 122: tlsauth.eap [Secondary_Authorization] Syntax

Parameter	Function
Enable	<p>Specifies whether secondary authorization checking is enabled.</p> <ul style="list-style-type: none"> • If set to 0, this feature is disabled and the EAP-TLS plug-in accepts the user upon proof of ownership of a private key that matches a valid certificate. If this setting is 0, no other settings in this section are applicable to the plug-in's operation. • If set to 1, a secondary authorization check against a traditional authentication method such as an SQL plug-in is performed. <p>Default value is 1.</p>
UseSubjectCNAsUserName	<p>Once the EAP-TLS module has concluded its processing, it may still defer to a traditional authentication method (core or plug-in) for final authorization. To do so, it must provide a username and password to the traditional authentication method.</p> <p>If set to 1, the EAP-TLS module parses the Subject attribute of the client's certificate for the least significant 'CN=' and takes the value of this attribute (for example, 'George Washington') as the username being passed to the traditional authentication method.</p> <p>Important: At any given instance, you can set only one of these parameters (UseSubjectCNAsUserName, UsePrincipalNameAsUserName, UseUserNameAttributeAsUserName, UseCallingStationIdAsUserName) to 1.</p> <p>Default value is 1.</p>

Table 122: tlsauth.eap [Secondary_Authorization] Syntax (*continued*)

Parameter	Function
UsePrincipalNameAsUserName	<p>Once the EAP-TLS module has concluded its processing, it may still defer to a traditional authentication method (core or plug-in) for final authorization. To do so, it must provide a username and password to the traditional authentication method.</p> <ul style="list-style-type: none"> • If set to 0, the username passed to the traditional authentication method is the username retrieved from the Subject field of the client certificate (see description of UseSubjectCNAsUserName above). • If set to 1, the EAP-TLS module uses the principal name (Subject Alternate Name or Other Name) from the client certificate (for example, 'joe@acme.com') as the username being passed to the traditional authentication method. <p>Default value is 0.</p> <p>Important: At any given instance, you can set only one of these parameters (UseSubjectCNAsUserName, UsePrincipalNameAsUserName, UseUserNameAttributeAsUserName, UseCallingStationIdAsUserName) to 1.</p>
UseUserNameAttributeAsUserName	<p>Indicates whether or not the plug-in should substitute the User Name for the RADIUS username before attempting to perform an inner authentication check. The default is not to make this substitution.</p> <p>Default value is 0.</p> <p>Important: At any given instance, you can set only one of these parameters (UseSubjectCNAsUserName, UsePrincipalNameAsUserName, UseUserNameAttributeAsUserName, UseCallingStationIdAsUserName) to 1.</p>

Table 122: tlsauth.eap [Secondary_Authorization] Syntax (*continued*)

Parameter	Function
UseCallingStationIdAsUserName	<p>Indicates whether or not the plug-in should substitute the Calling Station Id for the RADIUS User Name before attempting to perform an inner authentication check. The default is not to make this substitution.</p> <p>Default value is 0.</p> <p>Important: At any given instance, you can set only one of these parameters (UseSubjectCNAsUserName, UsePrincipalNameAsUserName, UseUserNameAttributeAsUserName, UseCallingStationIdAsUserName) to 1.</p>
UseInnerRadius	<p>Indicates whether or not inner authentication is to be performed.</p> <p>Set 1 to enable an inner authentication and 0 to disable inner authentication.</p> <p>Default value is 0.</p>
FixedPassword	<p>By default, the secondary authorization check includes a username but no other user credentials, because no password or similar credential for the client is available at the conclusion of the TLS handshake. Some authentication methods (Native User, LDAP, and SQL) can be configured to not require user credentials.</p> <p>If you plan to use secondary authorization against an authentication method (for example, LDAP) that cannot be configured to ignore the lack of user credentials, you may specify a fixed password that the plug-in uses on all secondary authorization checks.</p> <p>Default is to perform the check without user credentials.</p>

Table 122: tlsauth.eap [Secondary_Authorization] Syntax (*continued*)

Parameter	Function
Include_Certificate_Info	<p>If set to 1, the EAP-TLS plug-in adds four attributes to the request before the secondary authorization check is performed:</p> <ul style="list-style-type: none"> • The Funk-Peer-Cert-Subject attribute contains the value of the Subject attribute in the client certificate. • The Funk-Peer-Cert-Principal attribute contains the value of the principal name (Subject Alternate Name or Other Name) attribute of the client certificate. • The Funk-Peer-Cert-Issuer attribute contains the value of the Issuer attribute in the client certificate. • The Funk-Peer-Cert-Hash attribute contains a hexadecimal ASCII representation of the SHA1 hash of the client certificate. <p>These attributes are ignored if the authentication method that performs the authentication check does not use them.</p> <p>Default value is 0.</p>
RequestFilter	<p>Indicates the filter to be used to edit the attributes used in the inner authentication request. The filter can be used to modify attributes to influence routing of the inner authentication through attribute editing realm selection.</p> <p>Filter is not applied by default.</p>
ResponseFilter	<p>Indicates the filter to be used to edit attributes in the authentication response.</p> <p>Filter is not applied by default.</p>

Table 122: tlsauth.eap [Secondary_Authorization] Syntax (*continued*)

Parameter	Function
ProfileAttribute	<p>Indicates response attribute from the inner authentication method can contain the name of a profile to apply to the Access-Accept message.</p> <p>The profile name will be present in the attribute returned from the response. If the profile name is not available in the SBR, an Access-Reject message is sent.</p> <p>Profile is not applied by default.</p>
Realm	<p>Indicates directed or proxy realm to which inner authentication requests will be sent.</p> <p>If a realm name is configured in the SBR, all inner authentications will be forwarded to the realm. If a realm name is not configured, then standard authentication takes place as defined in the proxy.ini file</p> <p>The default is standard authentication.</p>

[CRL_Checking] Section

The [CRL_Checking] section ([Table 123 on page 342](#)) lets you specify settings that control how Steel-Belted Radius Carrier performs certificate revocation list (CRL) checking.

Table 123: tlsauth.eap [CRL_Checking] Syntax

Parameter	Function
Enable	<p>Specifies whether CRL checking is enabled.</p> <p>Default value is 0 (disabled).</p>
Retrieval_Timeout	<p>Specifies the time (in seconds) that EAP-TLS waits for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request results in a reject.</p> <p>Default value is 5 seconds.</p>

Table 123: `tlsauth.eap` [CRL_Checking] Syntax (*continued*)

Parameter	Function
Expiration_Grace_Period	<p>Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TLS always attempts to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache.</p> <ul style="list-style-type: none"> • If set to 0 (strict expiration mode), EAP-TLS does not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TLS considers the expired CRL as an acceptable stand-in from the time the CRL expires to the time the grace period ends. <p>Default value is 0 (strict expiration mode).</p>
Allow_Missing_CDP_Attribute	<p>Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS does not know how to retrieve a CRL and cannot perform a revocation check on the certificate.</p> <ul style="list-style-type: none"> • If set to false, EAP-TLS does not accept a CRL with a missing CDP attribute. • If set to true, EAP-TLS allows such certificates and skip CRL checking for them. <p>Default value is true.</p>
Default_LDAP_Server_Name	<p>Specifies what LDAP server name to use if the CDP contains a value that begins with the string <code>//ldap:\\</code>. This style of CDP (generated by some CAs) does not include the identity of the LDAP server.</p> <p>Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you do not specify a server name and such certificates are encountered, the CRL retrieval fails.</p>
LDAP_Bind_Version	<p>Enables the selection of the LDAP protocol when binding to an LDAP server (2 or 3)</p> <p>The default is 2 (LDAP version 2)</p>

[Session_Resumption] Section

The [Session_Resumption] section lets you specify whether session resumption is permitted and under what conditions session resumption is performed. The [Session_Resumption] section consists of the parameters listed in [Table 124 on page 344](#).

NOTE: For session resumption to work, the network access server must be configured to handle the Session-Timeout return list attribute, because the network access server must be able to tell the client to reauthenticate after the session timer has expired.

Table 124: tlsauth.eap [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	<p>Set this attribute to the maximum number of seconds you want the client to remain connected to the network access server before having to reauthenticate.</p> <ul style="list-style-type: none"> • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the RADIUS client on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. <p>Default value is 0.</p> <p>Entering a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. You can configure the resumption limit to make most reauthentications fast and computationally cheap.</p>
Termination_Action	<p>Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are:</p> <ul style="list-style-type: none"> • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. <p>Default value is -1. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.</p>

Table 124: tlsauth.eap [Session_Resumption] Syntax (continued)

Parameter	Function
Resumption_Limit	<p>Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature.</p> <p>This type of reauthentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature.</p> <p>Default value is 0.</p>

Sample tlsauth.eap File

```
[Bootstrap]
LibraryName=tlsauth.so
Enable=1

; Maximum TLS Message fragment length
TLS_Message_Fragment_Length = 1020

; Indicates whether the EAP-TLS module is to check
; whether the User Name provided in the RADIUS request
; matches the principal name in the client's certificate.
Verify_User_Name_Is_Principal_Name = 1

; Indicates whether the EAP-TLS module should return
; the MS-MPPE-Send-Key and MS-MPPE-Recv-Key attribute upon
; successfully authenticating the user.
Return_MPPE_Keys = 1

; Specifies the size of the prime to use for DH modular
; exponentiation.
DH_Prime_Bits = 1536

[Secondary_Authorization]
; Indicates whether secondary authorization is to be
; performed. Set to 1 to require a secondary authorization
; check against traditional authentication method
; (for example, SQL plug-in)
```

```

Enable = 1

; Indicates whether the plug-in should substitute the CN
; contained in the client certificate for the RADIUS User
; Name before the secondary authorization check
Convert_User_Name_To_Subject_CN = 1

; Indicates whether the plug-in should substitute the
; principal name contained in the Subject Alternate Name
; (Other Name) field of the client certificate for the
; RADIUS User Name before secondary authorization check.
Convert_User_Name_To_Principal_Name = 0

; Indicates whether the secondary authorization check
; should use no user credentials or a fixed password.
FixedPassword = test

; Indicates whether attributes containing information
; about the client certificate should be added to the
; request before secondary authorization is performed.
; The attributes include Funk-Peer-Cert-Subject,
; Funk-Peer-Cert-Principal, Funk-Peer-Cert-Issuer, and
; Funk-Peer-Cert-Hash. The default is not to include
; these attributes.
Include_Certificate_Info = 0

[Session_Resumption]
; Maximum length of time (in seconds) the RAS/AP
; allows the session to persist before the client is asked
; to reauthenticate.
Session_Timeout = 600

; The value to return for the Termination-Action attribute
; sent in an accepted client.
Termination_Action = 0

; The length of time (in seconds) during which an
; authentication request that seeks to resume a previous
; TLS session is considered acceptable.
Resumption_Limit = 3600

```

Configuring Secondary Authorization

The EAP-TLS plug-in may be configured to perform a secondary authorization check that typically requires a traditional authentication method that can be configured to authenticate users without the presence of credentials.

Examples for the Oracle SQL plug-in and the LDAP plug-in authentication are provided below.

SQL Authentication

The `.aut` file below shows an example of how the Oracle SQL plug-in can be configured so that password information is not required as input or output.

To configure these two plug-ins to cooperate, no password has been given in the **SQL= string** entry in the [Settings] section, and the **Password=** entry in the [Results] section has been similarly left empty.

```
[Settings]
SQL=SELECT FullName FROM orasqlauth WHERE username = %Name/50s

[Results]
; Empty definition of Password= indicates password to be ignored,
; since EAP-TLS is assumed to have already authenticated the user.
Password=
FullName=1/255s
;Profile=2/48
;Alias=3/48
```

For more information, see [“SQL Authentication” on page 403](#).

If the SQL authentication method used for secondary authorization is intended to be used only in conjunction with EAP-TLS, use Web GUI to set **EAP-Only=1** and **EAP-Type=TLS** in the appropriate section of the `eap.ini` file to prevent unintended use of this SQL authentication method for traditional authentication requests.

LDAP Authentication

The `.aut` file below shows an example of how the LDAP plug-in can be configured so that password information is not required as input or output.

To configure the EAP-TLS and LDAP plug-ins to cooperate properly, the **BindName=** option has been utilized in the [Settings] section to log into the LDAP server and no **%password=** setting has been specified in the [Response] section.

```
[Settings]
BindName=uid=admin,ou=administrators,o=bigco.com
BindPassword=adminPassword
```

```

Search=DoLdapSearch

[Request]
%Username=User-Name

[Response]
%profile=TheUserProfile

[Search/DoLdapSearch]
Base=ou=Special Users,o=bigco.com
Scope=2
Filter=(uid=<User-Name>)
Attributes=AttrList
Timeout=20
%DN=dn

```

For more information, see [“LDAP Authentication” on page 444](#).

If the LDAP authentication method used for secondary authorization is intended to be used only in conjunction with EAP-TLS, use Web GUI to set **EAP-Only=1** and **EAP-Type=TLS** in the appropriate section of the **eap.ini** file to prevent unintended use of this LDAP authentication method for traditional authentication requests.

ttlsauth.aut File

NOTE: Use the Web GUI to maintain settings in the **ttlsauth.aut** file. Do not edit the **ttlsauth.aut** file manually.

Settings for the EAP-TTLS authentication method are stored in the **ttlsauth.aut** file. The **ttlsauth.aut** configuration file is read each time the Steel-Belted Radius Carrier server receives a SIGHUP (1) signal.

[Bootstrap] Section

The [Bootstrap] section of the **ttlsauth.aut** file ([Table 125 on page 349](#)) specifies information that Steel-Belted Radius Carrier uses to load the EAP-TTLS authentication method.

Table 125: ttlsauth.aut [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the executable binary that implements the EAP-TTLS method. Default value is ttlsauth.so .
Enable	<p>Specifies whether the EAP-TTLS authentication module is enabled.</p> <ul style="list-style-type: none"> • If set to 0, EAP-TTLS is disabled. • If set to 1, EAP-TTLS is enabled. <p>Default value is 0.</p>
InitializationString	<p>Specifies the name of the EAP-TTLS authentication method.</p> <p>The name of each authentication method must be unique. If you create additional .aut files to implement authentication against multiple databases, the InitializationString value in each file must specify a unique method name.</p> <p>Default value is EAP-TTLS.</p>

[Server_Settings] Section

The [Server_Settings] section ([Table 126 on page 350](#)) lets you configure the basic operation of the EAP-TTLS plug-in.

Cipher_Suites Parameter

The **Cipher_Suites** parameter defined in the **ttlsauth.aut** [Server_Settings] section, specifies the TLS cipher suites (in order of preference) that the server uses for TTLS. When SBR Carrier receives a TTLS message, it compares the cipher suites in the client message to the cipher suites defined in this parameter. A match is selected based on both type (for example DSS) and order of preference defined in the client cipher suite list. If no match is found, SBR Carrier returns a handshake failure alert and closes the connection. Following are several examples of the cipher suite selection process:

Example 1

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

**0xC00A,0xC014,0xC019,0xC009,0xC013,
0x00AF,0x00B9,0xC035,0xC09A**

Client cipher suite list:

**0x0040,0x0033,0x0032,0x0016,0x0013,0x0066,0x0035,
0x002f,0x0015,0x0012,0x000a,0x0005,0xC014**

Match found: **0xC014**

In this example SBR Carrier selects 0xC014 because it is the first algorithm listed in the client cipher suite list that is also listed in the SBR Carrier cipher suite list, and because the type is also a match.

Example 2

SBR Carrier cipher suite list defined in **Cipher_Suites** parameter:

0xC00A,0xC014,0xC019

Client cipher suite list:

0x0039,0x0033,0x0032,0x0016,0x0013,0x0066,
0x0035,0x002f,0x0015,0x0012,0x000a,0x0005

Match found: No match found, results in handshake failure.

Table 126: ttlsauth.aut [Server_Settings] Syntax

Parameter	Function
TLS_Message_Fragment_Length	<p>Specifies the maximum size TTLS message length that may be generated during each iteration of the TTLS exchange. This value affects the number of RADIUS challenge/response round-trips required to conclude the TLS exchange. A value of 1400 may result in 6 round-trips, while a value of 500 may result in 15 round-trips.</p> <p>Some Access Points may have problems with RADIUS responses or EAP messages that exceed the size of one Ethernet frame (1500 bytes including IP/UDP headers).</p> <p>Minimum value is 500.</p> <p>Maximum value is 4096.</p> <p>Default value is 1020, which prevents the RADIUS challenge response (carried in a UDP packet) from exceeding one Ethernet frame.</p>

Table 126: ttlsauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Return_MPPE_Keys	<p>Setting this attribute to 1 causes the module to include RADIUS MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes in the final RADIUS Accept response sent to the Access Point. This is necessary for the Access Point to key the WEP encryption.</p> <p>If the Access Point is authenticating only end users and WEP is not being used, this attribute may be set to 0.</p> <p>For the optional WiMAX mobility module, set this to 0.</p> <p>Default value is 1.</p>
TLS_Protocol_Version	<p>Specifies the TLS protocol version on which the server expects the client to initiate the handshake process. The value can be one of the following:</p> <ul style="list-style-type: none"> • 31—TLS protocol version 1.0 • 32—TLS protocol version 1.1 • 33—TLS protocol version 1.2 <p>Default value is 31.</p> <p>If you set a value other than 31, 32, or 33, then the default TLS protocol version 1.0 (31) is considered.</p>
DH_Prime_Bits	<p>Specifies the size of the prime number that the module uses for Diffie-Hellman exponentiation. Selecting a larger prime number makes the system less susceptible to certain types of attacks but requires more CPU processing to compute the Diffie-Hellman key agreement operation.</p> <p>Valid values are 512, 1024, 1536, 2048, 3072, and 4096.</p> <p>Default value is 1024.</p>

Table 126: ttlsauth.aut [Server_Settings] Syntax (*continued*)

Parameter	Function
Cipher_Suites	<p>Specifies the TLS cipher suites (in order of preference) that the server uses. These cipher suites are documented in RFC 2246, <i>The TLS Protocol Version 1</i>, RFC 4346, <i>The TLS Protocol Version 1.1</i>, and RFC 5246, <i>The TLS Protocol Version 1.2</i>.</p> <p>For more information see “Cipher_Suites Parameter” on page 349.</p> <p>Default value is: 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013.</p> <p>See Table 112 on page 313 for the list of tested cipher suites and their TLS protocol versions.</p>
Require_Client_Certificate	<ul style="list-style-type: none"> • If set to 1, specifies that the client must provide a certificate as part of the TTLS exchange. • If set to 0, no client certificate is required. <p>Default value is 0.</p>

[Inner_Authentication] Section

The [Inner_Authentication] section ([Table 127 on page 352](#)) lets you specify the way in which the inner authentication step is to operate.

Table 127: ttlsauth.aut [Inner_Authentication] Syntax

Parameter	Function
Directed_Realm	<p>Omitting this setting causes the inner authentication request to be handled like any other request received from a RAS.</p> <p>Specifying the name of a directed realm causes the request to be routed based on the methods listed in the directed realm.</p> <p>Default is to process the inner authentication through standard request processing.</p>

[Request_Filters] Section

Request filters ([Table 128 on page 353](#)) affect the attributes of inner authentication requests.

NOTE: The filters named in these settings must be defined in the **filter.ini** file.

Table 128: ttlsauth.aut [Request_Filters] Syntax

Parameter	Function
Transfer_Outer_Attribs_to_New	<p>This filter affects only a new inner authentication request (rather than continuations of previous requests).</p> <p>If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request.</p> <p>If this filter is not specified, no attributes from the outer request are transferred to the inner request.</p>
Transfer_Outer_Attribs_to_Continue	<p>This filter affects only a continued inner authentication request (rather than the first inner authentication request).</p> <p>If this filter is specified, all attributes from the outer request are transferred to the inner request and this filter is applied. The transfer occurs and the filter is applied before any attributes specified in the inner authentication are added to the request.</p> <p>If this filter is not specified, no attributes from the outer request are transferred to the inner request.</p>
Edit_New	<p>This filter affects only a new inner authentication request (rather than continuations of previous requests).</p> <p>If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_New in this table) and attributes included in the inner authentication request sent through the tunnel by the client.</p> <p>If this filter is not specified, the request remains unaltered.</p>

Table 128: ttlsauth.aut [Request_Filters] Syntax (continued)

Parameter	Function
Edit_Continue	<p>This filter affects only a continued inner authentication request (rather than a new inner authentication request).</p> <p>If this filter is specified, it is applied to the inner request that is the cumulative result of attributes transferred from the outer request (see Transfer_Outer_Attribs_To_Continue in this table) and attributes included in the inner authentication request sent through the tunnel by the client.</p> <p>If this filter is not specified, the request remains unaltered.</p>

[Response_Filters] Section

Response filters ([Table 129 on page 354](#)) affect the attributes in the responses returned to authentication requests

NOTE: The filters named in these settings must be defined in the **filter.ini** file.

Table 129: ttlsauth.aut [Response_Filters] Syntax

Parameter	Function
Transfer_Inner_Attribs_To_Accept	<p>This filter affects only an outer Access-Accept response that is sent back to a network access server.</p> <p>If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response.</p> <p>If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.</p>

Table 129: ttlsauth.aut [Response_Filters] Syntax (*continued*)

Parameter	Function
Transfer_Inner_Attribs_To_Reject	<p>This filter affects only an outer Access-Reject response that is sent back to a network access server.</p> <p>If this filter is specified, the filter is applied to the inner authentication response and all resulting attributes are transferred to the outer authentication response.</p> <p>If this filter is not specified, no inner authentication response attributes are transferred to the outer authentication response.</p>

[CRL_Checking] Section

The [CRL_Checking] section ([Table 130 on page 355](#)) lets you specify settings that control how Steel-Belted Radius Carrier performs certificate revocation list (CRL) checking.

Table 130: ttlsauth.aut [CRL_Checking] Syntax

Parameter	Function
Enable	<p>If set to 1, the CRL checking is enabled for EAP-TTLS.</p> <p>Default value is 0.</p>
Retrieval_Timeout	<p>Specifies the time (in seconds) that EAP-TTLS waits for a CRL checking transaction to complete when the CRL check involves a CRL retrieval. When CRL retrieval takes longer than the specified time, the user's authentication request is rejected.</p> <p>Default value is 5 seconds.</p>

Table 130: ttlsauth.aut [CRL_Checking] Syntax (*continued*)

Parameter	Function
Expiration_Grace_Period	<p>Specifies the time (in seconds) after expiration during which a CRL is still considered acceptable. EAP-TTLS always attempts to retrieve a new CRL when it is presented with a certificate chain and it finds an expired CRL in its cache.</p> <ul style="list-style-type: none"> • If set to 0 (strict expiration mode), EAP-TTLS does not accept a CRL that has expired. • If set to a value greater than 0 (lax expiration mode), EAP-TTLS considers the expired CRL as an acceptable stand-in from the time the CRL expires to the time the grace period ends. <p>Default value is 0 (strict expiration mode).</p>
Allow_Missing_CDP_Attribute	<p>Specifies whether the omission of a CDP attribute in a non-root certificate is acceptable. Without a CDP attribute, EAP-TLS does not know how to retrieve a CRL and cannot perform a revocation check on the certificate.</p> <ul style="list-style-type: none"> • If set to 0, EAP-TLS does not accept a CRL with a missing CDP attribute. • If set to 1, EAP-TLS allows such certificates and skips CRL checking for them. <p>Default value is 1.</p>
Enable_CRL_Cache_Timeout	<p>Specifies whether CRL cache timeout is enabled. Valid values are:</p> <ul style="list-style-type: none"> • If set to 0, the CRL is refreshed whenever the CRL in the cache expires. • If set to 1, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in the CRL_Cache_Timeout_period parameter or when the scheduled CRL expiration time occurs, whichever comes first. <p>Default value is 0.</p> <p>After a CRL has expired (because its scheduled expiration time has passed or because the CRL cache has timed out), Steel-Belted Radius Carrier uses the expiration grace period to determine whether to use the current CRL.</p>

Table 130: ttlsauth.aut [CRL_Checking] Syntax (*continued*)

Parameter	Function
CRL_Cache_Timeout_Period	<p>Specifies the maximum time period (in hours) that a CRL can exist in the cache before it begins to expire.</p> <ul style="list-style-type: none"> • If you enter 0, Steel-Belted Radius Carrier always regards the CRL in the cache as expired and downloads a new CRL every time it receives a client certificate request. • If you enter a number greater than 0, the CRL begins to expire when the age of the CRL in the cache exceeds the number of hours specified in this parameter or when the scheduled CRL expiration time occurs, whichever comes first. <p>Default value is 168 hours.</p> <p>NOTE: You must set Enable_CRL_Cache_Timeout to 1 or the CRL_Cache_Timeout_Period parameter is ignored.</p>
Default_LDAP_Server_Name	<p>Specifies what LDAP server name to use if the CDP contains a value that begins with the string <code>//ldap:\\</code>. This style of CDP (generated by some CAs) does not include the identity of the LDAP server.</p> <p>Specify the name of the LDAP that contains the CRLs if you expect to encounter certificates with this style CDP. If you do not specify a server name and such certificates are encountered, CRL retrieval fails.</p>
LDAP_Bind_Version	<p>Enables the selection of the LDAP protocol when binding to an LDAP server (2 or 3)</p> <p>The default is 2 (LDAP version 2)</p>

[Session_Resumption] Section

The [Session_Resumption] section ([Table 131 on page 358](#)) lets you specify whether session resumption is permitted and under what conditions session resumption is performed.

NOTE: For session resumption to work, the network access server must be configured to handle the Session-Timeout return list attribute, because the network access server must be able to tell the client to reauthenticate after the session timer has expired.

Table 131: ttlsauth.aut [Session_Resumption] Syntax

Parameter	Function
Session_Timeout	<p>Set this attribute to the maximum number of seconds you want the client to remain connected to the network access server before having to reauthenticate.</p> <ul style="list-style-type: none"> • If set to a number greater than 0, the lesser of this value and the remaining resumption limit (see description below) is sent in a Session-Limit attribute to the network access server on the RADIUS Access Accept response. • If set to 0, no Session-Limit attribute is generated by the plug-in. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute. <p>Default value is 0.</p> <p>Entering a value such as 600 (10 minutes) does not necessarily cause a full reauthentication to occur every 10 minutes. You can configure the resumption limit to make most reauthentications fast and computationally cheap.</p>
Termination_Action	<p>Specifies the value to return for the Termination-Action attribute sent for an accepted client. This is a standard attribute supported by most Access Points and determines what happens when the session timeout is reached. Valid values are:</p> <ul style="list-style-type: none"> • -1: Do not send the attribute. • 0: Send the Termination-Action attribute with a value of 0. • 1: Send the Termination-Action attribute with a value of 1. <p>Default value is -1. This does not prevent the authentication methods performing secondary authorization from providing a value for this attribute.</p>
Resumption_Limit	<p>Set this attribute to the maximum number of seconds you want the client to be able to reauthenticate using the TLS session resumption feature.</p> <p>This type of reauthentication is fast and computationally cheap. It does, however, depend on previous authentications and may not be considered as secure as a complete (computationally expensive) authentication. Specifying a value of 0 disables the session resumption feature.</p> <p>Default value is 0.</p>

Sample ttlsauth.aut File

```
[Bootstrap]
LibraryName=ttlsauth.so
Enable=1
InitializationString=EAP-TTLS

; Maximum TLS Message fragment length EAP-TLS handles.
TLS_Message_Fragment_Length = 1020

; Indicates whether the EAP-TLS module should return the
; MS-MPPE-Send-Key and MS-MPPE-Recv-Key attribute upon successful
; authentication of user.
Return_MPPE_Keys = 1

; Size of the prime to use for DH modular exponentiation.
DH_Prime_Bits = 1536
; TLS cipher suites (in order of preference)
; that the server is to use.
Cipher_Suites = 0x0067,0x006B,0xC030,0xC028,0xC014,0xC013
; Specifies the TLS Protocol Version on which the server expects client to initiate
the handshake process.
TLS_Protocol_Version = 31

[Inner_Authentication]
; Specifies how inner authentication routing operates.
Directed_Realm = ttls_realm

[Request_Filters]
Transfer_Outer_Attribs_to_New = My_Xfer_Out_New_Filter
Transfer_Outer_Attribs_to_Continue = My_Xfer_Out_Con_Filter
Edit_New = My_Edit_New_Filter
Edit_Continue = My_Continue_Filter

[Response_Filters]
Transfer_Inner_Attribs_To_Accept = My_Xfer_Acc_Filter
Transfer_Inner_Attribs_To_Reject = My_Xfer_Rej_Filter

[Session_Resumption]
; Maximum length of time (in seconds) the NAD/AP allows
; the session to persist before the client is asked
; to reauthenticate.
Session_Timeout = 600
```

```
; Value to return for the Termination-Action attribute sent  
; sent in an accepted client.  
Termination_Action = 0  
  
; Maximum length of time (in seconds) during which an authentication  
; request that seeks to resume a previous TLS session is  
; considered acceptable.  
Resumption_Limit = 3600
```

For this to work, you must also provide the following settings in the [EAP-TTLS] section of the **eap.ini** file:

```
First-Handle-Via-Auto-EAP = 0  
EAP-Type = TTLS
```


Session State Register (SSR) Configuration Files

IN THIS CHAPTER

- [Configuring the config.ini File | 361](#)
- [Configuring the dbclusterndb.gen File | 368](#)
- [Using the georedSess.ses File to Configure the Geo-Redundancy Feature | 379](#)

Configuring the config.ini File

The **config.ini** file contains information about each node involved in the SSR cluster. This includes configuration parameters for data nodes and connections between them. Use the **config.ini** file to distribute this information to all processes participating in the cluster. The **config.ini** file is stored in the **/opt/JNPRhadm** directory.

For more information about the recommended settings of the **config.ini** file, consult the database vendor documentation.

[tcp default] Section

The [tcp default] section ([Table 132 on page 361](#)) of **config.ini** specifies the buffer size required to send and receive data between data nodes.

Table 132: config.ini [tcp default] Fields

Parameter	Function
ReceiveBufferMemory	<p>Specifies the size of the buffer used when receiving data from the TCP/IP socket.</p> <p>You can set a value from 64 KB through 4 GB.</p> <p>Default value is 2 MB.</p>
SendBufferMemory	<p>Specifies the size of the buffer used when sending data to the TCP/IP socket.</p> <p>You can set a value from 64 KB through 4 GB.</p> <p>Default value is 2 MB.</p>

[ndbd default] Section

The [ndbd default] section ([Table 133 on page 362](#)) of **config.ini** specifies parameters required to configure the behavior of data nodes.

Table 133: config.ini [ndbd default] Fields

Parameter	Function
ConnectCheckIntervalDelay	<p>Specifies the interval for checking the connection between data nodes.</p> <p>A data node that fails to respond within an interval of ConnectCheckIntervalDelay seconds is considered suspect, and is considered dead after two such intervals.</p> <p>You can set a value from 0 through 4,000,000,000 milliseconds.</p> <p>Default value is 1500 milliseconds.</p> <p>NOTE: The ConnectCheckIntervalDelay parameter helps protect against intermittent spikes of latency, which could cause nodes to fail due to heartbeat timeouts. This feature checks all nodes immediately after the first heartbeat timeout occurs and provides more efficient behavior in detecting the failed nodes.</p>
DataMemory	<p>Defines the amount of space, in bytes, available for storing database records. The entire amount specified by this value is allocated in memory, so it is extremely important that the machine have sufficient physical memory to accommodate it.</p> <p>The minimum space is 1 MB and there is no maximum size. However, the maximum size has to be adapted so that the process does not start swapping when the limit is reached. This limit is determined by the amount of physical RAM available on the machine and by the amount of memory that the operating system may commit to any one process.</p> <p>Default value is 80 MB.</p>
Diskless	<p>Enables or disables the diskless feature. When enabled, the tables are not check pointed to disk and no logging occurs. The cluster online backup is disabled. In addition, a partial start of the cluster is not possible.</p> <p>The diskless feature is disabled by default.</p>

Table 133: config.ini [ndbd default] Fields (*continued*)

Parameter	Function
HeartbeatIntervalDbApi	<p>Specifies the interval between the heartbeat signals sent to the MySQL server.</p> <p>Each data node sends heartbeat signals to each MySQL server to ensure that they remain in contact. If a MySQL server fails to send a heartbeat signal within this interval, it is declared dead. In this case, all ongoing transactions are completed and all resources are released.</p> <p>You can set a value from 100 through 4,000,000,000 milliseconds.</p> <p>Default value is 1500 milliseconds.</p>
HeartbeatIntervalDbDb	<p>Specifies how often heartbeat signals are sent and how often to expect to receive them.</p> <p>You can set a value from 10 through 4,000,000,000 milliseconds.</p> <p>Default value is 1500 milliseconds.</p>
LockExecuteThreadToCPU	<p>Specifies the ID of the CPU assigned to handle the NDBCLUSTER execution thread.</p> <p>You can set a value from 0 through 65,535.</p> <p>Default value is 65,535.</p>
LockPagesInMainMemory	<p>Specifies whether to lock a process into memory and avoid any swapping to disk. This can be used to guarantee real-time characteristics of the cluster.</p> <ul style="list-style-type: none"> • If set to 0, disables locking • If set to 1, performs locking after allocating memory for the process • If set to 2, performs locking before allocating memory for the process <p>Default value is 0.</p>
MaxNoOfConcurrentOperations	<p>Specifies the number of records that can be in an update phase or locked simultaneously.</p> <p>You can set a value from 32 through 4,000,000,000.</p> <p>Default value is 32,768.</p>

Table 133: config.ini [ndbd default] Fields (*continued*)

Parameter	Function
MaxNoOfConcurrentTransactions	<p>Specifies the number of parallel transactions possible in a node.</p> <p>You can set a value from 32 through 4,000,000,000.</p> <p>Default value is 4096.</p>
MaxNoOfExecutionThreads	<p>Specifies the number of local query handler (LQH) threads spawned by a multithreaded version of ndbd.</p> <p>You can set a value from 2 through 8.</p> <p>Default value is 2.</p>
MaxNoOfLocalOperations	<p>Specifies the number of records that can be in an update phase.</p> <p>You can set a value from 32 through 4,000,000,000.</p> <p>By default, this parameter is calculated as follows: 1.1 x MaxNoOfConcurrentOperations.</p>
MaxStartFailRetries	<p>Specifies the limit to the number of restart attempts by the data node in the event of a failure on startup.</p> <p>You can set a value from 0 through 4,000,000,000.</p> <p>Default value is 3.</p>
NoOfFragmentLogFiles	<p>Specifies the number of REDO log files for the node and the amount of space allocated to REDO logging.</p> <p>You can set a value from 3 through 4,000,000,000.</p> <p>Default value is 16.</p> <p>NOTE: NoOfFragmentLogFiles must be set to 300 or higher to provide sufficient space for REDO logs.</p>
NoOfReplicas	<p>Defines the number of replicas for each table stored in the cluster, and specifies the size of node groups.</p> <p>A node group is a set of nodes storing the same information.</p> <p>You can set a value from 1 through 4.</p> <p>Default value is 2.</p>

Table 133: config.ini [ndbd default] Fields (*continued*)

Parameter	Function
RealTimeScheduler	<p>Enables or disables real-time scheduling of NDBCLUSTER threads.</p> <p>The RealTimeScheduler feature is disabled by default.</p>
RedoBuffer	<p>Sets the size of the buffer in which the REDO log is written.</p> <p>You can set a value from 1 MB through 4 GB.</p> <p>Default value is 32 MB.</p>
StartFailRetryDelay	<p>Specifies the number of seconds between restart attempts by the data node in the event of a failure on startup.</p> <p>Default value is 0 (no delay).</p> <p>NOTE: This parameter is ignored if StopOnError is disabled.</p>
StartPartialTimeout	<p>Specifies how long the cluster waits for all data nodes to come up before the cluster is initialized. This parameter is used to avoid a partial cluster startup, whenever possible.</p> <p>You can set a value from 0 through 4,000,000,000 milliseconds.</p> <p>Default value is 30,000 milliseconds.</p> <p>NOTE: If StartPartialTimeout is set to 0, the cluster starts only if all the nodes are available.</p> <p>This parameter is overridden during an initial start or restart of the cluster.</p>
StartPartitionedTimeout	<p>Specifies the wait time, in seconds, for the cluster to start after waiting for StartPartialTimeout milliseconds and the cluster is still in a partitioned state.</p> <p>You can set a value from 0 through 4,000,000,000 milliseconds.</p> <p>Default value is 60,000 milliseconds.</p> <p>NOTE: If StartPartitionedTimeout is set to 0, the cluster waits indefinitely.</p> <p>This parameter is overridden during an initial start or restart of the cluster.</p>

Table 133: config.ini [ndbd default] Fields (*continued*)

Parameter	Function
StopOnError	<p>Specifies whether an ndbd process should exit or perform an automatic restart when an error condition is encountered.</p> <ul style="list-style-type: none"> • If set to 0, the ndbd process tries to start the data node after an exit due to an error condition. • If set to 1, the ndbd process exits when there is an error condition and must be restarted manually. <p>Default value is 0.</p>
TimeBetweenGlobalCheckpoints	<p>Specifies the interval between global checkpoints.</p> <p>All transactions taking place within a given interval are put into a global checkpoint, which can be assumed as a set of committed transactions that has been flushed to disk.</p> <p>You can set a value from 10 through 32,000 milliseconds.</p> <p>Default value is 2000 milliseconds.</p>
TimeBetweenLocalCheckpoints	<p>Specifies the interval between local checkpoints.</p> <p>Ensures that local checkpoints are not performed in a cluster where relatively few updates are taking place. In most clusters with high update rates, it is likely that a new local checkpoint is started immediately after the previous one is completed.</p> <p>This parameter is specified as the base-2 logarithm of the number of 4-byte words.</p> <p>You can set a value from 0 through 31.</p> <p>Default value is 20.</p>
TimeBetweenWatchDogCheck	<p>Specifies the number of milliseconds between watchdog checks.</p> <p>To prevent the main thread from getting stuck in an endless loop at some point, a watchdog thread checks the main thread. If the process remains in the same state after three checks, the watchdog thread terminates it.</p> <p>You can set a value from 70 through 4,000,000,000 milliseconds.</p> <p>Default value is 6000 milliseconds.</p>

Table 133: config.ini [ndbd default] Fields (*continued*)

Parameter	Function
TotalSendBufferMemory	<p>Specifies the total amount of memory to allocate to each node for which it is set for use among all configured transporters.</p> <p>To be backward-compatible with existing configurations, this parameter takes as its default value the sum of the maximum send buffer sizes of all configured transporters, plus an additional 32 KB (one page) per transporter.</p> <p>You can set a value from 256 KB through 429 MB.</p>
TransactionDeadlockDetectionTimeout	<p>Specifies the time that the transaction coordinator waits for another node to execute a query before the coordinator terminates the transaction. This parameter is important for handling node failures and detecting deadlocks.</p> <p>You can set a value from 50 through 4,000,000,000 milliseconds.</p> <p>Default value is 1200 milliseconds.</p>

[ndbd] Section

ndbd is the process that is used to handle all data in the tables by using the NDB Cluster storage engine. The [ndbd] section ([Table 134 on page 367](#)) of **config.ini** specifies information that empowers a data node to handle the distribution of transactions, recover lost nodes, checkpoint committed transactions to disk, perform an online backup of transactions in real time, and perform other related tasks.

```
[ndbd]
Hostname = 10.13.20.13
NodeId = 41
# HeartBeatOrder =
```

The preceding fields in the **config.ini** file are generated by the configure script.

Table 134: config.ini [ndbd] Fields

Parameter	Function
Hostname	Specifies the name of the host on which the data node is to reside.
NodeId	<p>Specifies a unique node ID that is used as the address of the node to which or from which all internal messages of the cluster are sent or received. For data nodes, each data node in the cluster must have a unique identifier.</p> <p>You can set a value from 1 through 48.</p>

Table 134: config.ini [ndbd] Fields (*continued*)

Parameter	Function
HeartBeatOrder	<p>Specifies the order of heartbeat transmissions between data nodes.</p> <p>NOTE: In a four-way cluster containing two pairs of mirrored NDB nodes in different data centers, HeartBeatOrder must be set appropriate to the installation, as described in the config.ini file, and the M nodes and D nodes must be restarted in succession.</p> <p>A new setting for HeartBeatOrder may alleviate certain issues. This is not set by default; it must be configured manually.</p>

Configuring the dbclusterndb.gen File

Use the **dbclusterndb.gen** file to configure the database settings used by each SBR Carrier node to access the SSR database. The **dbclusterndb.gen** file is stored in the *radiusdir* directory, usually **/opt/JNPRsbr/radius**.

[Bootstrap] Section

The [Bootstrap] section ([Table 135 on page 368](#)) of **dbclusterndb.gen** specifies information that SBR Carrier nodes use to load SSR functions.

```
[Bootstrap]
LibraryName = dbclusterndb
Enable = 1
ManagementMode = 0
```

Table 135: dbclusterndb.gen [Bootstrap] Fields

Parameter	Function
LibraryName	<p>Specifies the name of the cluster database module. Default value is dbclusterndb.</p> <p>Do not change this unless you are advised to do so by Juniper Networks Technical Support.</p>

Table 135: dbclusterndb.gen [Bootstrap] Fields (continued)

Parameter	Function
Enable	<ul style="list-style-type: none"> • If set to 0, the high availability functionality is disabled. • If set to 1, the high availability functionality is enabled. <p>Default value is 1 in the file provided with SSR. If this setting is removed from the dbclusterndb.gen file, default value switches to 0.</p>
ManagementMode For information about management mode, see the section on <i>Session State Register Administration</i> in the <i>SBR Carrier Administration and Configuration Guide</i> .	<ul style="list-style-type: none"> • If set to 0, the SBR Carrier server operates in standard high-availability mode. • If set to 1, the SBR Carrier server operates in management mode. <p>Default value is 0.</p>

[NDB] Section

The [NDB] section (Table 136 on page 370) of **dbclusterndb.gen** identifies how SBR Carrier nodes access the SSR database.

```
[NDB]
ManagementNode = 127.0.0.1:5235;nodeid=30
ConnectRetries = 3
DelayBetweenConnectRetriesSec = 5
TimeoutForFirstAliveSec = 10
WaitForAllNodesAlive = 0
TimeoutAfterFirstAliveSec = 10
NdbHandles = 32
NdbHandlesAlert = 1
NDBHardErrorThreshold = 10
```

Table 136: dbclusterndb.gen [NDB] Fields

Parameter	Function
ManagementNode	<p>Specifies the NDB connect-string value, made up of the IP address of the management node hosts, the port the management node uses for connection requests, and the node ID (NDB connect-string) of the local SBR Carrier node.</p> <p>This information is created by the installation script and should only be changed by that script or under direction of JTAC.</p>
ConnectRetries	<p>Specifies how many times SBR Carrier tries to connect to the management nodes.</p> <p>Default value is 3.</p>
DelayBetweenConnectRetriesSec	<p>Specifies how many seconds SBR Carrier waits between retries when trying to connect to the management nodes.</p> <p>Default value is 5 seconds.</p>
TimeoutForFirstAliveSec	<p>Specifies how many seconds SBR Carrier waits for the first NDBD server to confirm that it can communicate before communicating with the database cluster.</p> <p>Default value is 10 seconds.</p>
WaitForAllNodesAlive	<ul style="list-style-type: none"> • If set to 0, SBR Carrier does not wait for confirmation that all NDB nodes are alive before communicating with the database cluster. • If set to 1, SBR Carrier waits for confirmation that all NDB nodes are alive before communicating with the database cluster. <p>Default value is 0.</p>
TimeoutAfterFirstAliveSec	<p>Specifies how many seconds SBR Carrier waits after the first NDBD server alive indicator.</p> <p>Default value is 10 seconds.</p>

Table 136: dbclusterndb.gen [NDB] Fields *(continued)*

Parameter	Function
NdbHandles	<p>Specifies the number of NDB handles used for parallel database transactions. One NDB handle is needed for each ongoing database operation. Therefore, the number of available handles poses an upper limit to the number of concurrent operations a SBR Carrier server can make to NDB. Performance increases with concurrency up to a point, where thread overhead overwhelms the benefits of concurrency. The best balance depends on your environment.</p> <p>NOTE: Each NDB handle uses more than 32K of memory. Increasing the value of NdbHandles increases the start time for SBR Carrier very slightly.</p> <p>Enter a value in the range 1–128. Note that one handle is attached permanently to each cache thread, and one handle is reserved by the system for special usage when in ManagementMode.</p> <p>NOTE: The NdbHandles setting, and its associated alert message, only counts operational handles, i.e., not those reserved for MgmtMode or for caching threads.</p> <p>Default value is 32.</p>
NdbHandlesAlert	<ul style="list-style-type: none"> • If set to 0, the SBR Carrier server does not record NDB handle messages in its log file. • If set to 1, the SBR Carrier server records messages identifying the maximum number of concurrent NDB handles that have been used since the server was restarted in its log file. The messages take the form: <p style="text-align: center;">Max concurrent NDB handles = N</p> <p>Default value is 0.</p>

Table 136: dbclusterndb.gen [NDB] Fields (continued)

Parameter	Function
NDBHardErrorThreshold	<p>Specifies the threshold value for NDB hard errors. If the number of hard errors exceeds the threshold value, SBR Carrier starts to monitor the cluster health and determines whether to persist sessions in the local file on the SBR Carrier server on the basis of the value set in the FallbackLocal parameter in the radius.ini file.</p> <p>You can enter the value in the range from 1 through 100. Default value is 10.</p> <p>This parameter is reloaded every time when SBR Carrier receives a SIGHUP (1) signal.</p>

[Database] Section

The [Database] section ([Table 137 on page 372](#)) of **dbclusterdb.gen** controls how SBR Carrier front ends accesses the SSR database.

```
[Database]
Database = SteelBeltedRadius
Retries = 6
DelayBetweenRetriesMillisec = 50
RetryAlertThreshold = 4294967295
```

Table 137: dbclusterndb.gen [Database] Fields

Parameter	Function
Database	<p>Specifies the name of the database used by SBR Carrier.</p> <p>Default value is SteelBeltedRadius.</p>
ReconnectOnHUP	<p>Controls whether the database cluster is disconnected and reconnected after a SIGHUP (1) signal is sent to the dbcluster plug-in.</p> <ul style="list-style-type: none"> • If set to 1, the database reconnects after receiving a SIGHUP (1) signal. • If set to 0, the database does not reconnect after receiving a SIGHUP (1) signal. <p>Default value is 0.</p>

Table 137: dbclusterndb.gen [Database] Fields (continued)

Parameter	Function
Retries	<p>Specifies how many times SBR Carrier tries to connect to the management nodes before giving up.</p> <p>Default value is 6.</p>
DelayBetweenRetriesMillisec	<p>Specifies the base number of milliseconds SBR Carrier waits before retrying a database operation. The first retry delay is 1 x the value specified for DelayBetweenRetries, the second retry delay is 2 x the value specified for DelayBetweenRetries, and the nth retry delay is n x the value specified for DelayBetweenRetries.</p> <p>Default value is 50 milliseconds.</p>
RetryAlertThreshold	<p>Specifies the threshold for recording log messages when a retry that exceeds the threshold is attempted. The log message identifies why the retry was attempted (that is, why the previous attempt failed).</p> <ul style="list-style-type: none"> • If RetryAlertThreshold is set to 0, log messages are written before every database retry. • If RetryAlertThreshold is set to the value of (Retries -1), log messages are written before the last retry and after the last retry (if it fails). • If RetryAlertThreshold is set to the value of Retries, log messages are written only after the last retry fails (that is, when the last retry failed and no further attempt will be made). • If RetryAlertThreshold is set to a value of greater than the value of Retries, log messages are not recorded. <p>Default value is 4294967295 (0xffffffff).</p>
UseConnectionManager	<p>UseConnectionManager=<bool>. This parameter should always be set to true unless a change is recommended by Juniper Networks Technical Support. This might have a minor performance impact on throughput, bandwidth-bound installations. Contact your sales engineer or Juniper Networks Technical Support for more information.</p> <p>The default is enabled.</p>

[IpAddressPools] Section

The [IpAddressPools] section (Table 138 on page 374) of `dbclusterndb.gen` specifies aging and caching parameters for IP address pools. Some of these settings are system-wide and cannot be overridden. Other settings establish system defaults, which can be overridden for specific IP address pools.

```
[IpAddressPools]
MinUnusedAgeSec = 300
MaxAgeRetries = 2
AgePercent = 50
AgeRetryAlertThreshold = 4294967295
NumCacheThreads = 2
StartCachingAtBootTime = 1
CacheThreadSleepMin = 1000
CacheThreadSleepMax = 2000
CacheLowWater = 100
CacheHighWater = 250
CacheChunkSize = 50
EmergencyChunkSize = 1
CacheAlertThreshold = 0
CacheThreadVerbose = 0
```

Table 138: `dbclusterndb.gen` [IpAddressPools] Fields

Parameter	Function
MinUnusedAgeSec	<p>Specifies how many seconds an IP address can remain unused before SBR Carrier reassigns it.</p> <p>Default value is 300 seconds.</p>
MaxAgeRetries	<p>Specifies the number of times SBR Carrier attempts to retrieve acceptably aged IP addresses before retrieving any available address.</p> <p>If MaxAgeRetries is 0, SBR Carrier makes one attempt to look for addresses that have been idle for at least MinUnusedAgeSec seconds, then no retries if enough addresses are not found.</p> <p>Default value is 2.</p>

Table 138: dbclusterndb.gen [IPAddressPools] Fields (continued)

Parameter	Function
AgePercent	<p>Specifies the percentage (a number in the range 0–100) that SBR Carrier uses as the multiplier for MinUnusedAgeSec when enough IP addresses that have been idle for MinUnusedAgeSec cannot be found. For example, if MinUnusedAgeSec is 400 and AgePercent is 75, SBR Carrier would look for addresses idle for at least 400 seconds, then look for addresses that have been idle for at least 300 ($400 * 0.75$) seconds, then look for addresses that have been idle for at least 225 ($300 * 0.75$) seconds.</p> <p>If AgePercent is 100, only the original MinUnusedAgeSec is used.</p> <p>If AgePercent is 0, then age is disregarded.</p> <p>Default value is 50.</p>
AgeRetryAlertThreshold	<p>Specifies the threshold (0–4294967295) for recording log messages when an aged-based retry that exceeds the threshold is attempted. The log message identifies why the retry was attempted (that is, why the previous attempt failed).</p> <ul style="list-style-type: none"> • If AgeRetryAlertThreshold is set to 0, log messages are written before every age-based retry. • If AgeRetryAlertThreshold is set to the value of (MaxAgeRetries - 1), log messages are written before the last age-based retry and after the last retry (if it fails). • If AgeRetryAlertThreshold is set to the value of MaxAgeRetries, log messages are written only after the last retry fails (that is, when no addresses have been found and no further attempt will be made). • If AgeRetryAlertThreshold is set to a value of greater than the value of MaxAgeRetries, log messages are not recorded. <p>Default value is 4294967295 (0xffffffff).</p>
NumCacheThreads	<p>Specifies how many parallel threads SBR Carrier uses to cache IP addresses.</p> <p>Default value is 2.</p>

Table 138: dbclusterndb.gen [IPAddressPools] Fields (continued)

Parameter	Function
StartCachingAtBootTime	<ul style="list-style-type: none"> • If set to 1, SBR Carrier fills its IP address pool cache immediately when it is rebooted. • If set to 0, SBR Carrier fills its IP address pool cache when it receives an address request. <p>Default value is 1.</p>
CacheThreadSleepMin	<p>Specifies the minimum range of time (0–4294967295) a cache-filling thread waits before it goes to the database to get another cache of IP addresses. Set this parameter to better manage your caching threads.</p> <p>Default value is 1000 milliseconds.</p>
CacheThreadSleepMax	<p>Specifies the maximum range of time (CacheThreadSleepMin–4294967295) a cache-filling thread waits before it goes to the database to get another cache of IP addresses. Set this parameter to better manage your caching threads.</p> <p>Default value is 2*CacheThreadSleepMin milliseconds.</p>
CacheLowWater	<p>Specifies the minimum number of addresses that must be available in the address cache for an IP address pool. When the number of addresses in a server's cache falls below the CacheLowWater value, the server begins requesting blocks of IP addresses</p> <p>Default value is 100.</p>
CacheHighWater	<p>Specifies the number of addresses that must be available in a server's IP address cache for an IP address pool before it stops adding addresses to the cache</p> <p>The CacheHighWater value must be greater than or equal to the CacheLowWater value.</p> <p>Default value is 250.</p>
CacheChunkSize	<p>Specifies the number of addresses to retrieve every time SBR Carrier requests a block of IP addresses for an IP address pool.</p> <p>Default value is 50.</p>

Table 138: dbclusterndb.gen [IPAddressPools] Fields (continued)

Parameter	Function
EmergencyChunkSize	<p>Specifies the (0–<i>CacheChunkSize</i>) number of addresses to retrieve every time <i>SBR Carrier</i> requests a block of IP addresses to use from a pool, when that pool's cache is empty, to directly retrieve those addresses from the IP address table in the database and put them in the cache.</p> <p>Default value is 1.</p>
CacheAlertThreshold	<p>Specifies the threshold (0–4294967295) for recording log messages when the number of addresses in the cache falls below the threshold value.</p> <ul style="list-style-type: none"> • If CacheAlertThreshold is set to 0, log messages are not written. • If CacheAlertThreshold is set to 1, log messages are written when the address cache is empty. • If CacheAlertThreshold is set to a value equal to or greater than the sum of the values of CacheHighWater and CacheChunkSize, log messages are written whenever an address is pulled from the cache. <p>Default value is 0.</p>
CacheThreadVerbose	<ul style="list-style-type: none"> • If set to 1, print out an informational message about what the caching thread just did. The message contains: thread identity (OS thread ID number), how long it napped, pool cached, number of IP addresses cached, and the length of time it took to retrieve the addresses from the database. • If set to 0, no caching thread information is recorded. <p>Use this parameter to fine-tune the caching parameters.</p> <p>Default value is 0.</p>

[IPAddressPools:PoolName] Section

The [IPAddressPools:PoolName] section ([Table 139 on page 378](#)) of **dbclusterdb.gen** identifies how SBR Carrier specifies the override settings for IP address aging for a named IP address pool. You can create as many [IPAddressPools:PoolName] section as you require in the **dbclusterdb.gen** file to tune caching for individual address pools.

[IPAddressPools:Gold]

```
StartCachingAtBootTime = 0
CacheLowWater = 10
CacheHighWater = 25
CacheChunkSize = 5
EmergencyChunkSize = 2
CacheAlertThreshold = 1
```

NOTE: You cannot use an [IpAddressPools:PoolName] section to override settings other than those listed here. For example, you cannot enter a **MaxAgeRetries** setting in an [IpAddressPools:PoolName] section to override the value specified in the [IpAddressPools] section.

NOTE: Authentication requests may be rejected when you have configured the default settings for parameters in the [IpAddressPools:PoolName] section and the named IP address pool consists of a smaller number of IP addresses. To avoid this, we recommend that you set the number of IP addresses in the pool to a value greater than the values entered for **CacheLowWater**, **CacheHighWater**, and **CacheChunkSize**.

Table 139: dbclusterndb.gen [IpAddressPools:PoolName] Fields

Parameter	Function
StartCachingAtBootTime	<ul style="list-style-type: none"> • If set to 1, SBR Carrier fills its IP address pool cache immediately when it is rebooted. • If set to 0, SBR Carrier fills its IP address pool cache when it receives an address request. <p>Default value is 0.</p>
CacheLowWater	Specifies the minimum number of addresses that must be available in the named pool's IP address cache. When the number of addresses in the pool's cache falls below the CacheLowWater value, the server begins requesting blocks of IP addresses.
CacheHighWater	<p>Specifies the number of addresses that must be available in a server's IP address cache before it adding addresses to the IP address cache for the named IP address pool.</p> <p>The CacheHighWater value must be greater than or equal to the CacheLowWater value.</p>

Table 139: dbclusterndb.gen [IpAddressPools:PoolName] Fields (continued)

Parameter	Function
CacheChunkSize	Specifies the number of addresses to retrieve every time SBR Carrier requests a block of IP addresses for the named IP address pool.
EmergencyChunkSize	Specifies the (0– <i>CacheChunkSize</i>) number of addresses to retrieve every time SBR Carrier requests a block of IP addresses to use from a pool, when that pool's cache is empty, to directly retrieve those addresses from the IP address table in the database and put them in the cache. Default value is 1.
CacheAlertThreshold	Specifies the threshold (0–4294967295) for recording log messages when the number of addresses in the cache falls below the threshold value. <ul style="list-style-type: none"> • If CacheAlertThreshold is set to 0, log messages are not written. • If CacheAlertThreshold is set to 1, log messages are written when the address cache is empty. • If CacheAlertThreshold is set to a value equal to or greater than the sum of the values of CacheHighWater and CacheChunkSize, log messages are written whenever an address is pulled from the cache. Default value is 0.

Using the georedSess.ses File to Configure the Geo-Redundancy Feature

You can use the **georedSess.ses** file to configure the operational characteristics of Geo-redundancy.

[Bootstrap] Section

The **[Bootstrap]** section ([Table 140 on page 380](#)) of the **georedSess.ses** file specifies information that SBR Carrier uses to enable Geo-redundancy.

Table 140: georedSess.ses [Bootstrap] Syntax

Parameter	Function
Enable	<p>Specifies whether Geo-redundancy is enabled.</p> <ul style="list-style-type: none"> • If set to 1, Geo-redundancy is enabled. • If set to 0, Geo-redundancy is disabled. <p>Default value is 0.</p>

[ClientSettings] Section

You can configure the [ClientSettings] section ([Table 141 on page 380](#)) of the **georedSess.ses** file to extract session information from SBR, prepare replication information, and send the replication information to a remote server.

Table 141: georedSess.ses [ClientSettings] Syntax

Parameter	Function
ClientComponent	<p>Specifies whether the client component is activated or not.</p> <ul style="list-style-type: none"> • If set to true, the SBR acts as a client component for the remote server. This option can also be configured for the standalone edition of SBR. <p>NOTE: The client component is not activated if you have specified an invalid IPv4 address in the GeoRedHost parameter.</p> <ul style="list-style-type: none"> • If set to false, the client component is not activated. <p>Default value is true.</p>
ClientRetryThreadPoolSize	<p>Specifies the size of the retry thread pool used by the Geo-redundancy client.</p> <p>The value entered in this parameter must be greater than 0.</p> <p>Default value is 100.</p>
ClientThreadPoolSize	<p>Specifies the size of the thread pool used by the Geo-redundancy client.</p> <p>The value entered in this parameter must be greater than 0.</p> <p>Default value is 500.</p>
DelayBetweenRetries	<p>Specifies the number of seconds the client component waits between retries while resending replication packets.</p> <p>Default value is 5 seconds.</p>

Table 141: georedSess.ses [ClientSettings] Syntax (*continued*)

Parameter	Function
GeoRedAckPort	<p>Specifies the UDP port on which the client component listens for acknowledgments from the remote cluster for incoming requests.</p> <p>Default value is 9095.</p>
GeoRedHost	<p>Specifies the IPv4 address of the remote cluster.</p>
GeoRedHostPort	<p>Specifies the port on which the remote cluster component listens for replication packets.</p> <p>Default value is 9090.</p>
GeoRedMaxSessions	<p>Specifies the maximum number of accounting packets allowed for a retry.</p> <p>The value entered in this parameter must be greater than 0.</p> <p>Default value is 200000.</p>
ReplicationRequest	<p>Indicates the integer codes that specify the CST fields to be replicated. (For the list of all integer codes, refer to Table 142 on page 382)</p> <p>The codes must be comma separated.</p> <p>Mandatory codes are 1,3,4,6,16,17,21,22,25,28,31.</p>
Retries	<p>Specifies the number of times the client component attempts to resend a replication packet.</p> <p>You can enter the value in the range of 1 through 100.</p> <p>Default value is 10.</p> <p>NOTE: The replication request cache contributes to the RADIUS process space. Setting this value to more than 2 makes the replication request caches grow instantly, which may cause memory outage during high-load conditions.</p>
ServerId	<p>Specifies the unique server ID that is used to identify the server in the inventory of the remote server.</p> <p>A sequence ID is incremented for each replication request. If multiple client components replicate information to a remote server, sequence IDs may become duplicated due to replication request caches that are maintained by the remote server. To uniquely identify a replication request, a combination of sequence ID and the server ID is used.</p> <p>Default value is 100.</p> <p>NOTE: The value entered in this parameter must be greater than 0.</p>

Table 142: Replication Request Codes

Integer Code	CST Field Name
1	UniqueSessionId
2	CreationTime
3	ExpirationTime
4	Ipv4Address
5	IpPoolOrdinal
6	NasName
7	SessionState
8	UserConcurrencyId
9	MobileIpType
10	3gpp2ReqType
11	WimaxClientType
12	WimaxAcctFlows
13	3gpp2HomeAgentAddr
14	AcctAutoStop
15	ClassAttribute
16	UserName
17	AcctSessionId
18	TransactionId
19	NasPortType
20	NasPort
21	CallingStationId

Table 142: Replication Request Codes (*continued*)

Integer Code	CST Field Name
22	CalledStationId
23	MobileCorrelationId
24	Ipv6InterfaceId
25	NasIpv4Address
26	NasIpv6Address
27	(Reserved for future use)
28	AcctMultiSessionId
29	FunkOuterUserName
30	Ipv6Prefix
31	Ipv6Address

[ServerSettings] Section

You can configure the **[ServerSettings]** section ([Table 143 on page 383](#)) of the **georedSess.ses** file to receive replication information from client components of the remote cluster.

Table 143: georedSess.ses [ServerSettings] Syntax

Parameter	Function
GeoRedServerAckPort	Specifies the port on which the server component sends acknowledgments to the remote server. Default value is 9095.
GeoRedHome	Specifies the IPv4 address of the server component.
GeoRedServerPort	Specifies the port on which the local cluster component listens for replication packets. Default value is 9090.

Table 143: georedSess.ses [ServerSettings] Syntax (continued)

Parameter	Function
ServerComponent	<p>Specifies whether the server component is activated.</p> <ul style="list-style-type: none"> • If set to true, SBR acts as the server component for client components of the remote server. This option is applicable only to the cluster edition of SBR because the replicated sessions are stored in a shared SSR cluster. <p>NOTE: The server component is not activated if you have specified an invalid IPv4 address in the GeoRedHome parameter.</p> <ul style="list-style-type: none"> • If set to false, the server component is not activated. <p>Default value is true.</p>
ServerDispatchThreadPoolSize	<p>Specifies the size of the dispatch thread pool used by the Geo-redundancy server.</p> <p>The value entered in this parameter must be greater than 0.</p> <p>Default value is 100.</p>
ServerThreadPoolSize	<p>Specifies the size of the thread pool used by the Geo-redundancy server.</p> <p>The value entered in this parameter must be greater than 0.</p> <p>Default value is 500.</p>

3

PART

Back-End Authentication and Accounting

SQL Plug-Ins | **386**

LDAP Plug-Ins | **429**

CDR Accounting Plug-Ins | **463**

SQL Plug-Ins

IN THIS CHAPTER

- Common Configuration Items | 386
- SQL Authentication | 403
- SQL Accounting | 413
- SQL Accessors | 419
- Detailed Use Cases | 422

This chapter explains the SQL authentication, accounting and accessor plug-ins.

Common Configuration Items

This section explains the configuration items that are common across all SQL plug-ins. These configuration items are found in the **.acc** (accounting), **.aut** (authentication), and **.gen** (generic accessor) files.

[Bootstrap] Section

The [Bootstrap] section ([Table 144 on page 386](#)) of the SQL configuration file specifies information that Steel-Belted Radius Carrier uses to load and start an SQL method.

Table 144: [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the executable binary that implements the SQL plug-in. For example, radsql_auth_ora.so in the case of the Oracle native SQL Authentication Plug-in. It should not be necessary to change the default value unless you have developed your own plug-in.

Table 144: [Bootstrap] Syntax (*continued*)

Parameter	Function
Enable	<p>Specifies whether the SQL method is enabled.</p> <ul style="list-style-type: none"> • If set to 0, the SQL method is disabled • If set to 1, the SQL method is enabled. <p>Default value is 0.</p>
InitializationString	<p>Specifies the name of the SQL method.</p> <p>The name of each SQL method must be unique.</p>

[Settings] Section

The [Settings] section of the SQL configuration file defines parameters that control the database connection.

NOTE:

The following log file warning message is shown when the Connect parameter is used along with a server definition in any of the .acc (accounting), .aut (authentication), and .gen (generic accessor) files:

WARNING: Deprecated 'Connect=' string in [Settings] section of '/opt/JNPRsbr/radius/radsql.acc' - The connect string may overlap with [Server] connection/activation target settings.

The Connect parameter should only be defined in the [Server/*name*] section. For information on using the Connect parameter, see “[*Server/name*] Sections” on page 399.

Table 145: [Settings] Syntax

Parameter	Function
ConcurrentTimeout	<p>Specifies the number of seconds a request may wait for execution before it is discarded. Because there may be only up to MaxConcurrent SQL statements executing at one time, new requests must be queued as they arrive until other statements are processed.</p> <p>NOTE: The MaxConcurrent parameter is valid only for Oracle SQL methods (for example, radsql.aut). It is not valid for JDBC SQL (for example, radsqljdbc.aut).</p>

Table 145: [Settings] Syntax (continued)

Parameter	Function
ConnectDelimiter	<p>(JDBC only) Specifies the character used to separate fields (DSN, UID, PWD) in the connect string.</p> <p>Default value is ; (semicolon). If the JDBC connect string requires use of semicolons as part of a field value, you can use this parameter to specify a different delimiter, such as ^ (caret).</p> <p>NOTE: In the case of JDBC, the Connect parameter is closely related to the Driver parameter. The exact syntax of these parameters is highly dependent upon the particular JDBC driver that you are using. You should always obtain your JDBC driver directly from your database vendor and consult the vendor's documentation for more details.</p>
DisableMetaData	<p>(MySQL JDBC only) Specifies whether to suppress errors that occur while using integer data types in a MySQL argument.</p> <ul style="list-style-type: none"> • If set to 1, SBR Carrier suppresses errors that occur while using integer data types in a MySQL argument. • If set to 0, SBR Carrier displays errors that occur while using integer data types in a MySQL argument. <p>Default value is 0.</p> <p>Consider an input variable MaxSessions of type varchar(20) which holds an integer value. It can be converted to its corresponding type by using the following: DECLARE maxSess DECIMAL; set maxSess=(SELECT CAST(MaxSessions as DECIMAL));In the above example, if DisableMetaData is set to 0 (disabled), the value "maxSess" will be NULL and an error "(CDataAccessorClassObject::getOutputVariable(): failed to get variable (result) from container" will be logged. To avoid this error and to set the proper value for "maxSess", set DisableMetaData set to 1 (enabled).</p>

Table 145: [Settings] Syntax (continued)

Parameter	Function
Driver	<p>(JDBC only) Specifies the third-party JDBC driver to support the database connection. For example:</p> <p>Driver=com/mysql/jdbc/Driver/</p> <p>Oracle example (using ojdbc14.jar)</p> <p>Driver=oracle/jdbc/OracleDriver</p> <p>MySQL example (using mysql-connector-java-5.0.5-bin.jar)</p> <p>Driver=com/mysql/jdbc/Driver</p> <p>MSSQL example (using mssqlserver.jar)</p> <p>Driver=com/provider/jdbc/sqlserver/SQLServerDriver</p> <p>NOTE: In the case of JDBC, the Connect parameter is closely related to the Driver parameter. The exact syntax of these parameters is highly dependent upon the particular JDBC driver that you are using. You should always obtain your JDBC driver directly from your database vendor and consult the vendor's documentation for more details.</p> <p>NOTE: Third-party JDBC drivers must be installed in the <code><JRE-path>/lib/ext</code> directory. Where, <code><JRE-path></code> indicates the path where the JRE (that is integrated with SBR Carrier) is installed in your system.</p> <p>See "JDBC Plug-ins" on page 401 for more information.</p>
ErrorMap	<p>Specifies the name of the file that contains the native error codes that are treated as soft errors (that is, errors that do not require the SQL method to disconnect from and reconnect to the remote database).</p> <p>NOTE: Steel-Belted Radius Carrier includes three default error map files: mssql.ini is for Microsoft SQL, mysql.ini is for MySQL, and oracle.ini is for Oracle. See "ErrorMap" on page 395 for information about configuring error map files.</p>

Table 145: [Settings] Syntax (*continued*)

Parameter	Function
LogLevel	<p>Activates logging for the SQL method and sets the level of detail of the message. The LogLevel may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose. If the LogLevel that you set in the SQL method configuration file is different than the LogLevel in <i>radius.ini</i>, then the lowest value controls.</p>
MaxConcurrent	<p>Specifies the maximum number of instances of a single SQL statement that may be executing at one time.</p> <p>NOTE: The MaxConcurrent parameter is valid only for Oracle SQL methods (for example, radsql.aut). It is not valid for JDBC SQL (for example, radsqljdbc.aut).</p> <p>NOTE: A setting of MaxConcurrent = 1 is sufficient for all but the most demanding environments. Increase this value slowly and conservatively. For more information about executing overlapping SQL statements, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>
MaxHardErrorRetries	<p>This parameter is added to resolve an issue where the database disconnected due to inactivity timeout. The MaxHardErrorRetries parameter enables the connection to be reestablished without failing the authentication by allowing you to set the number of additional attempts you want to make after hard errors have been encountered.</p> <p>Default is 0.</p>
MaxWaitReconnect	<p>Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection.</p> <p>WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.</p>

Table 145: [Settings] Syntax (*continued*)

Parameter	Function
OracleFailoverRetry	<p>Specifies the number of retry attempts that the Oracle client can perform when attempting an Oracle failover. No upper limit exists for the number of retries. Default value is 0. The retry attempts that the Oracle client performs are per database request.</p> <p>When a value is set, Oracle failover retries are attempted until the set number of retries is reached, after which the retry process is canceled. If the default value of 0 is set, the retry attempts continue until the Oracle failover succeeds or fails.</p> <p>NOTE: The Steel-Belted Radius Carrier shutdown process is not affected by the failover retry. When an SBR shutdown is initiated, the failover process is canceled and SBR is allowed to shut down.</p>
OracleSocketReadTimeout	<p>Specifies the maximum number of seconds a socket must wait when a network error has occurred or Oracle database is terminated abruptly.</p> <p>The default value is 0 second, which means timeout does not occur. If set to 0, the socket waits until the OS level socket timeout is reached.</p> <p>If a non-zero value is set, the socket waits until the configured time, after which the JDBC driver closes the database connection. Configuring a non-zero value prevents the waiting situation when there is a network error or database disconnection.</p> <p>NOTE: The OracleSocketReadTimeout parameter is valid only for Oracle JDBC SQL plug-ins (radsqldbc.aut, radsqldbc.acc, and sqlaccessor_jdbc.gen) with Oracle JDBC driver files ojdbc5.jar, ojdbc6.jar, or later. The OracleSocketReadTimeout parameter is not valid for Oracle SQL plug-ins (for example, radsql.aut) and other JDBC SQL methods such as MySQL and MS SQL.</p> <p>For MySQL and MS SQL, you can use the socketTimeout parameter (values in milliseconds) in the Connect string. For example:</p> <pre>Connect=DSN=jdbc:mysql://10.212.11.47:3306/test? socketTimeout=20000;UID=sbr;PWD=sbr</pre>
ParameterMarker	<p>Specifies the character or sequence of characters used as the parameter marker in a parameterized SQL query. Normally, this is the question mark (?), but this can vary among database vendors.</p>

Table 145: [Settings] Syntax (*continued*)

Parameter	Function
QueryTimeout	<p>Specifies the number of seconds to wait for the execution of an SQL request to complete before timing out.</p> <p>NOTE: QueryTimeout is not a precise timer and this parameter does not have any effect if there are network failures. Cancelling an SQL request also depends on the network latency as well as the query timeout setting on the server.</p>
ShutdownTimeout	<p>Specifies the maximum number of seconds to wait for outstanding database transactions when the server is in the process of shutting down. If this timeout expires, then any outstanding database transactions are forcibly terminated in order to allow the server to shut down. Changing the default value is not recommended.</p>
SQL	<p>Specifies the SQL statement used to access the database. The SQL statement may be broken over several lines by ending each line with a backslash. The backslash must be preceded by a space character, and followed by a newline character. The subsequent lines may be indented for better readability.</p> <p>See “SQL Parameter” on page 393 for more information.</p>
WaitReconnect	<p>Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.</p>

This section also explains the following:

- Limitations of Underlying Database APIs
- SQL Parameter
- ErrorMap
- LogLevel

Limitations of Underlying Database APIs

The **MaxConcurrent** parameter is valid only for Oracle SQL authentication (**radsql.aut**). It is not valid for JDBC SQL authentication (**radsqljdbc.aut**).

The **OracleSocketReadTimeout** parameter is valid only for Oracle JDBC SQL plug-ins (**radsqljdbc.aut**, **radsqljdbc.acc**, and **sqlaccessor_jdbc.gen**) with Oracle JDBC driver files **ojdbc5.jar**, **ojdbc6.jar**, or later.

SQL Parameter

- SBR tracks certain data pertaining to each RADIUS transaction. Data that comes directly from the RADIUS attributes contained in the RADIUS packets are referred as “attributes” (including subattributes). Data generated internally by SBR as a side-effect of processing the RADIUS transaction are referred as “items”.
- Attributes are defined in SBR configuration files known as “dictionaries”. Dictionaries determine data type of each attribute.

Items are not defined by dictionaries and the SBR data type of each item is fixed internally.

- SQL plug-ins support all attributes, as well as items such as *AuthType*, *TransactionTime*, *Name*, *FullName*, *UserName*, *Password*, *User*, *EffectiveUser*, *Realm*, *EffectiveRealm*, *RadiusClientName*, *NASName*, *NASModel*, *NASAddress*, and *NASIPv6Address*.
- The SQL= configuration parameter specifies an SQL statement to be executed.

In the case of authentication, there are typically some inputs supplied by SBR that are used by the database in order to retrieve some outputs that are then processed further by SBR.

In the case of accounting, there are typically only inputs supplied by SBR that are stored by the database (outputs could also be retrieved by the database, but are ignored by SBR as it does not do any further processing for accounting).

In general, either attributes or items can be supplied as inputs or outputs, or both to SQL statements.

- Inputs and outputs can be specified by substituting 'terms' of the SQL statement that would normally represent literal values or parameters to a database stored procedure. The substituted terms of the SQL statement are known as 'positional parameters' because they are automatically assigned integer placeholders in the order they appear. For example, :1, :2, :3, ...
- Outputs can also be specified by mapping 'columns' of the SQL statement 'result set'. However, the SQL statement result set must not contain more than one 'row'. An error occurs if SBR attempts to process a 'result set' that contains more than one row. See the [Results] section for more information.
- SBR cannot interpret the meaning of an SQL statement beyond the simple mechanics of substituting positional parameters and mapping columns of the result set. This makes it nearly impossible to detect errors and inconsistencies in the configuration. In particular, SBR cannot determine whether the SQL statement generates a result set or whether the only outputs are positional parameters. Ensure that you comment out the [Results] section if no result set is generated.
- The general syntax for substituting a positional parameter for a term of the SQL statement is as follows:

```
%<item>/[size][.digits][type][&][!direction]
```

Or

```
@<attribute>/[size][.digits][type][&][!direction]
```

Where, <> indicates required fields and [] indicates optional fields.

- item—name of an SBR item, for example, *UserName*
- attribute—name of an SBR attribute, for example, *Acct-Status-Type*

- size—maximum size in bytes required to represent this parameter
- digits—maximum number of decimal digits required to represent this parameter
- type—database type used to represent this parameter.

The 'type' of a positional parameter refers to the database type used to represent the parameter within the database, as opposed to the SBR type that is used to represent the corresponding item or attribute internally within the SBR. SBR must convert each item or attribute into a compatible database representation before these can be passed to the database as positional parameters. The specified type provides a hint to the SBR as to what type of conversion should be performed. The database may also perform additional type conversions, either explicitly as part of the SQL statement or implicitly when the type of the positional parameter is not exactly that which is required to perform some calculation or store/retrieve a column value.

SBR supports the following type flags for positional parameters: "s" = VARCHAR; "b" = VARBINARY; "t" = TIMESTAMP; "numeric" = NUMERIC; "n32" = INTEGER (32bit); "n64" = BIGINT (64bit); "n16" = SMALLINT (16bit); "n8" = TINYINT (8bit).

NOTE: The exact database types in capital may vary across database vendors and implementations. The default type depends on the SBR type and the database type detected by the SBR, but usually "s" = VARCHAR is assumed.

- &—This flag indicates that this parameter may be NULL (optional).

The optional null flag "&" indicates that a positional parameter may pass a NULL value. NULL values may be provided as inputs in lieu of missing/undefined items and attributes. NULL values may also be retrieved as outputs that do not cause any changes to the values of existing items and attributes. Without the optional null flag "&", the SBR converts any NULL values to something appropriate for the type of positional parameter involved, for example, an empty string for VARCHAR or zero for NUMERIC. By default, the SBR assumes that NULL values should not be passed and all NULL values are converted to some appropriate non-NULL value.

- direction—Direction in which the data flows.

The direction of a positional parameter refers to the direction in which the data flows. SBR supports the combination of the following direction flags for positional parameters: "i" = INPUT (data flows from the SBR into the database); "o" = OUTPUT (data flows out of the database to the SBR); "r" = REJECT OUTPUT (data flows out of the database to the SBR, but is only included in a RADIUS ACCESS-REJECT packet). The default direction is "i" = INPUT.

- It is often best to experiment with and verify the execution of the SQL statement using a command line utility such as Oracle sqlplus before attempting to configure the SQL plug-in. However, the configured SQL statement may differ in some minor ways, for example, terminating semi-colon (;) required when using the command line utility but prohibited when configuring the SQL plug-in. These minor differences may vary across database vendors and implementations.

ErrorMap

IN THIS SECTION

- [mssql.ini File | 395](#)
- [mysql.ini File | 396](#)
- [oracle.ini File | 396](#)

Specifies the name of the file that contains the native error codes that are treated as soft errors (that is, errors that do not require Steel-Belted Radius Carrier to disconnect from and reconnect to the remote database).

NOTE: Steel-Belted Radius Carrier includes three default error map files: **mssql.ini** (for Microsoft SQL using ODBC), **mysql.ini** (for MySQL using JDBC), and **oracle.ini** (for Oracle using OCI).

The following sections explain the error map files:

mssql.ini File

The **mssql.ini** configuration file specifies which errors returned by a back-end MS-SQL database are classified as soft errors. Error codes not listed in **mssql.ini** are presumed to be hard errors, which cause Steel-Belted Radius Carrier to drop and re-establish the connection to the MS-SQL database. Database-dependent RADIUS transactions fail while the connection to the MS-SQL database is being re-established.

Each entry in the **mssql.ini** configuration file consists of an error number (positive integer), followed by a descriptive comment. For best performance, use the **mssql.ini** file to identify only the most common soft errors.

NOTE: If you incorrectly define a hard error as a soft error and the error is encountered during processing, you may need to restart Steel-Belted Radius Carrier to reset the database plug-in.

[SoftErrors] Section

The [SoftErrors] section identifies each MS-SQL error code to be classified as a soft error. To include a comment or description for the error code, enter a semi-colon after the error code, followed by the comment.

```
[SoftErrors]
151 ; '%.*ls' is an invalid money value.
206 ; Operand type clash: %ls is incompatible with %ls
210 ; Syntax error converting datetime from binary/varbinary string.
212 ; Expression result length exceeds the maximum. %d max, %d found.
220 ; Arithmetic overflow error for data type %ls, value = %ld.
229 ; %ls permission denied on object '%.*ls', database '%.*ls', owner '%.*ls'.
```

mysql.ini File

The **mysql.ini** configuration file specifies which errors returned by a back-end MySQL database are classified as soft errors. Error codes not listed in **mysql.ini** are presumed to be hard errors, which cause Steel-Belted Radius Carrier to drop and re-establish the connection to the MySQL database. Database-dependent RADIUS transactions fail while the connection to the MySQL database is being re-established.

Each entry in the **mysql.ini** configuration file consists of an error number (positive integer), followed by a descriptive comment. For best performance, use the **mysql.ini** file to identify only the most common soft errors.

NOTE: If you incorrectly define a hard error as a soft error and the error is encountered during processing, you may need to restart Steel-Belted Radius Carrier to reset the database plug-in.

[SoftErrors] Section

The [SoftErrors] section identifies each MySQL error code to be classified as a soft error. To include a comment or description for the error code, enter a semi-colon after the error code, followed by the comment.

```
[SoftErrors]
1000 ; SQLSTATE: HY000 (ER_HASHCHK) hashchk
1001 ; SQLSTATE: HY000 (ER_NISAMCHK) isamchk
1002 ; SQLSTATE: HY000 (ER_NO) NO
1003 ; SQLSTATE: HY000 (ER_YES) YES
```

oracle.ini File

The **oracle.ini** configuration file specifies which errors returned by a back-end Oracle database are classified as soft errors. Error codes not listed in **oracle.ini** are presumed to be hard errors, which cause Steel-Belted Radius Carrier to drop and re-establish the connection to the Oracle database. Database-dependent RADIUS transactions fail while the connection to the Oracle database is being re-established.

Each entry in the **oracle.ini** configuration file consists of an error number (positive integer), followed by a descriptive comment. For best performance, use the **oracle.ini** file to identify only the most common soft errors.

NOTE: If you incorrectly define a hard error as a soft error and the error is encountered during processing, you may need to restart Steel-Belted Radius Carrier to reset the database plug-in.

[SoftErrors] Section

The [SoftErrors] section identifies each Oracle error code to be classified as a soft error. To include a comment or description for the error code, enter a semi-colon after the error code, followed by the comment.

```
[SoftErrors]
00001 ; unique constraint (string.string) violated
00036 ; maximum number of recursive SQL levels (string) exceeded
00054 ; resource busy and acquire with NOWAIT specified
00055 ; maximum number of DML locks exceeded
00057 ; maximum number of temporary table locks exceeded
00060 ; deadlock detected while waiting for resource
00100 ; no data found
```

LogLevel

Activates logging for the SQL method and sets the level of detail of the message. The LogLevel may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose. If the LogLevel that you set in the configuration file is different than the LogLevel in **radius.ini**, the lower value of the setting controls.

[Server] Section

There are two methods to establish a connection to the server:

- Connecting to a Single SQL Server
- Connecting to Multiple Servers

To connect to a single server, include a connect statement in the [Settings] section of **sqlaccessor.gen** or **sqlaccessor_jdbc.gen**. For example:

```
[Settings]
MethodName=SQL Accessor
Connect=username/password@servicename
.
.
.
```

Steel-Belted Radius Carrier can maintain multiple SQL server connections and authenticate users against authentication databases in a round-robin fashion. This convention distributes the authentication workload across several servers. The [Server] section of the SQL configuration file gives Steel-Belted Radius Carrier a pool of servers from which to create the round-robin list. The [Server] section names each server that might be used. It also provides rules for when to include or exclude each of the possible servers in the round-robin list.

```
[Server]
ServerName=TargetNumber
ServerName=TargetNumber
.
.
.
```

Table 146: [Server] Syntax

Parameter	Function
ServerName	The name of the configuration file section that contains configuration information for that server.
TargetNumber	An <i>activation target number</i> , a number that controls when this server is activated for backup purposes. <i>TargetNumber</i> is optional and may be left blank.

A Steel-Belted Radius Carrier server maintains connectivity with its SQL servers according to the following rules:

- The priority of the server by order. The first entry in the [Server] section has the highest priority.
- By activation target number. The rule for the activation target is that if the number of SQL servers to which Steel-Belted Radius Carrier is connected is less than the activation target, Steel-Belted Radius Carrier connects to the server and includes it in the round-robin list. While the number of active servers is equal to or greater than the activation target, Steel-Belted Radius Carrier does not use that server in the round-robin list. An activation target of 0 indicates that, in the current configuration, this machine is never used.

NOTE: If you configure the [Server] section, then you should also configure a [Server/ServerName] section for each *ServerName* specified in the [Server] section. The *ServerName* represents an arbitrary but unique name for each server. The [Server/ServerName] sections may be used to override parameters specified in the [Settings] section. Each [Server/ServerName] section must at least override the **Connect** parameter. Unlike other parameters, the **Connect** parameter in the [Settings] section should always be commented out when it is overridden; otherwise, the **Connect** parameter would be counted as an additional activation target, leading to an unexpected behavior.

[Server/name] Sections

You must provide a [Server/ *name*] section for each server you named in the [Server] section:

```
[Server/name]
Connect=username/password@servicename
```

where the values for *username* and *password* are specific to the SQL database, and *servicename* is the Oracle service name.

The Connect parameter specifies the string that must be passed to the database client engine to establish a connection to the database. This string has (or refers to) information about the name of the database, its location on the network, the password required to access it, and so forth.

The format of the Connect string depends on the type of database you use:

Oracle plug-ins:

Connect=<dB_username> /< dB_password>

```
Connect=scott/tiger@DBNAME
```

JDBC plug-ins:

Connect=DSN=< jdbc: provider:driver:dsn_name_here>;UID= <username_for_dB>;PWD= <password_for_dB>

MySQL Example (DBNAME instance at IP address 10.10.10.10 port 3306):

Connect=DSN=jdbc:mysql://10.10.10.10:3306/DBNAME; UID=scott;PWD=tiger

MS-SQL Example (DBNAME instance at IP address 10.10.10.10 port 1433):

**Connect=DSN=jdbc:microsoft:sqlserver://10.10.10.10:1433;
databaseName=DBNAME;UID=scott;PWD=tiger**

Oracle Example (DBNAME instance at IP address 10.10.10.10 port 1521):

Connect=DSN=jdbc:oracle:thin:@10.10.10.10:1521:DBNAME;

UID=scott;PWD=tiger

NOTE: Do not use the SA account or leave the password blank.

Last Resort Server

You may identify a “last resort” SQL server by providing a **LastResort** parameter in one of these [Server/**name**] sections, and setting its value to 1. If a SQL query against some other server results in “no record found,” the authentication server tries the last resort server before accepting or rejecting the user.

In the following example, server **s3** is the last resort server. The **@mydb** string refers to the service name for an Oracle database in the **tnsnames.ora** file (the server cannot connect to the Oracle database without this).

```
[Server]
s1=2
s2=2
s3=1

[Server/s1]
Connect=system1/manager

[Server/s2]
Connect=system2/manager@mydb2

[Server/s3]
Connect=system3/manager@mydb3
LastResort = 1
```

You might use the **LastResort** parameter to identify your primary accounts database. This enables Steel-Belted Radius Carrier to authenticate the user in the case where a user account is newly added to the primary accounts database but has not yet been propagated to all the SQL databases.

Load Balancing Example

The following excerpt from a **.acc** example file configures load balancing between two SQL servers (so that the work load is shared nearly equally between two servers). The tradeoff with this technique is that the data is split between two servers and must be reintegrated when processed. For example, the Accounting-START for an end user may be stored on one server and the corresponding Accounting-STOP on the other.


```

[Server]
s1=2
s2=2
[Server/s1]
Connect=system/*****@thor
[Server/s2]
Connect=system/*****@odin
[Type]
1=User
2=User
3=User
[Type/User]
SQL=INSERT INTO acct1(TransTime, FullName, \
Authenticator, NASName, NASAddress, Type, \
PacketsIn, PacketsOut) \
VALUES (%TransactionTime/t, %FullName/40s, \
%AuthType/40s, %NASName/40s, %NASAddress, \
%Type, @Acct-Input-Packets/n, \
@Acct-Output-Packets/n)

```

JDBC Plug-ins

This section describes how to install the JDBC driver from MySQL, MS-SQL, Oracle.

For MySQL:

1. Download the latest JDBC driver from Oracle that is appropriate for your MySQL server version.
2. Follow the installation directions provided by the vendor.

This typically involves extracting a zip file to obtain .jar files that implement the JDBC driver, for example, **mysql-connector-java-5.0.5-bin.jar** and possibly others.

3. Copy all the .jar files to the **<JRE-path>/lib/ext** directory. Where, **<JRE-path>** indicates the path where the JRE (that is integrated with SBR Carrier) is installed in your system.
4. Determine the name of the JDBC driver by consulting the documentation provided by the vendor.

It is usually possible to determine the name of the JDBC driver by executing a shell command similar to: **jar -tf mysql-connector-java-5.0.5-bin.jar |grep Driver**. Select the name that most resembles /com/mysql/jdbc/Driver and omit the .class extension.

5. Use the driver name obtained in Step 4 to configure the Driver parameter in the JDBC plug-in configuration file.

See <https://dev.mysql.com/doc/connector-j/5.1/en/connector-j-versions-java.html> for JDBC driver specifications and its compliance with MySQL.

For MS-SQL:

1. Download the latest JDBC driver from Microsoft that is appropriate for your MS-SQL server version.
2. Follow the installation directions provided by the vendor.

This typically involves extracting a tar file and running an install script to obtain a directory such as /opt/mssql or /opt/msSQLjdbc that contains the .jar files that implement the JDBC driver, for example, **mssqlserver.jar** and possibly others such as msbase.jar and msutil.jar.

3. Copy all the .jar files to the <JRE-path>/lib/ext directory. Where, <JRE-path> indicates the path where the JRE (that is integrated with SBR Carrier) is installed in your system.
4. Determine the name of the JDBC driver by consulting the documentation provided by the vendor.

It is usually possible to determine the name of the JDBC driver by executing a shell command similar to: **jar -tf mssqlserver.jar |grep Driver**. Select the name that most resembles /com/microsoft/jdbc/sqlserver/SQLServerDriver and omit the .class extension.

5. Use the driver name obtained in Step 4 to configure the Driver parameter in the JDBC plug-in configuration file.

See <https://docs.microsoft.com/en-us/sql/connect/jdbc/system-requirements-for-the-jdbc-driver> for JDBC driver specifications and its compliance with MS-SQL.

For Oracle:

1. Download the latest JDBC driver from Oracle that is appropriate for your Oracle server version.
2. Follow the installation directions provided by the vendor.

This typically involves extracting a tar file to obtain .jar files that implement the JDBC driver, for example, ojdbc14.jar and possibly others.

3. Copy all the .jar files to the `<JRE-path>/lib/ext` directory. Where, `<JRE-path>` indicates the path where the JRE (that is integrated with SBR Carrier) is installed in your system.
4. Determine the name of the JDBC driver by consulting the documentation provided by the vendor.
It is usually possible to determine the name of the JDBC driver by executing a shell command similar to: `jar -tf ojdbc14.jar |grep Driver`. Select the name that resembles `/oracle/jdbc/OracleDriver` and omit the .class extension.
5. Use the driver name obtained in Step 4 to configure the Driver parameter in the JDBC plug-in configuration file.

A similar procedure should be used for other databases that are not listed here. Consult the database vendor documentation for further details.

See http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html#01_03_1 and http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html#01_03_2 for JDBC driver specifications and its compliance with Oracle.

NOTE: In the case of Oracle, the native Oracle Plug-ins are preferred over JDBC Plug-ins since the JDBC API is necessarily more generic, less capable, and less performant than the native OCI API that is used by the native Oracle Plug-ins. Juniper does not recommend the use of Oracle JDBC Plug-ins in cases where no native Oracle Plug-in exists for very old or very new versions of Oracle servers that are no longer or not yet supported by SBRC.

SQL Authentication

This section describes the configuration file that controls SQL authentication in Steel-Belted Radius Carrier. The configuration file resides in the *radiusdir* directory.

The format of a configuration file is comparable to that of a Windows INI file: composed of several sections; section names are enclosed in brackets; each section may contain multiple parameter/value pairs.

Table 147: SQL Authentication Configuration File

File	Function
radsql.aut	Configures Oracle SQL authentication for Steel-Belted Radius Carrier.
radsqljdbc.aut	Configures JDBC SQL authentication for Steel-Belted Radius Carrier.

[Settings] Section

The [Settings] section of the SQL authentication configuration file defines parameters that control the database connection.

PasswordFormat

- If set to 0, Steel-Belted Radius Carrier tries to determine password format automatically.
- If set to 3, Steel-Belted Radius Carrier expects the password value encrypted with UNIXcrypt.

By default, the **PasswordFormat** parameter is in **radsql.aut** and **radsqljdbc.aut**.

ClearTextBinary

Specifies whether cleartext binary passwords are allowed. Setting this parameter to a non-zero value allows you to use cleartext binary passwords.

This parameter must be set to the length of the binary passwords in operation.

NOTE: In parameters that specify the SQL statement to be executed, cleartext binary passwords must be passed using binary compatible data types, for example, VARBINARY in the case of Oracle.

DefaultResults

- If set to 0, no default values are assumed and the user must explicitly enter all result items (if you are not calling a stored procedure).
- If set to 1, the default values for Results are used. This is the backward-compatibility setting and the setting if no value is specified in the file. In this case, each Result item must be explicitly specified.

SuccessResult

Specifies the string that is the expected result of a successful authentication, to be compared to the %result parameter.

If a value is specified for this field, it is used in the following manner upon execution of the SQL statement: if the value of %result is not equal to the value given for this field, the user is rejected. The test for textual equality is not case sensitive.

No such test, or rejection, is performed if no value is specified for this field.

This is a useful technique for coordinating with the custom functionality of stored procedures.

UpperCaseName

Specifies whether the user's login name is converted to uppercase characters before using it in the SQL statement execution.

- 0—Use the name exactly as received.
- 1—Convert the name to uppercase.

[Results] Section

The [Results] section ([Table 148 on page 405](#)) of the SQL authentication configuration file maps the columns named in its **SELECT** query to the type of data that Steel-Belted Radius Carrier expects these columns to contain.

```
[Results]
Password=1/48
Profile=2/48
```

The following parameters ([Table 148 on page 405](#)) may be present in a [Results] section. Each parameter represents a type of data required to authenticate an Access-Request, and if desired, apply authorization information as well.

NOTE: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 148: *.aut [Results] Syntax

Parameter	Function
%LoginLimit	Specifies the name of the variable identifying the Maximum Concurrent Connection limits.
%Password	<p>The value returned from this column is understood to be the user's password. The value returned by the SQL query is then matched with the user's password received in the Access-Request.</p> <p>By default, Steel-Belted Radius Carrier expects the user's password to be stored in the SQL table in clear-text format. If you want to configure Steel-Belted Radius Carrier to expect that the password value is encrypted with UNIXcrypt, then set PasswordFormat to 3 in the [Settings] section of the SQL authentication configuration file.</p>

Table 148: *.aut [Results] Syntax (continued)

Parameter	Function
%Profile	<p>The value returned from this column is interpreted as the name of the profile to associate with the user. The value returned by the SQL query is matched with an existing Profile entry of the same name. If the value is prof1, and a Profile called prof1 exists in the Steel-Belted Radius Carrier database, any return list or check list attributes in prof1 are applied to the user's connection.</p> <p>If the value cannot be matched with an existing Profile in the Steel-Belted Radius Carrier database, the user is rejected due to "Insufficient Resources."</p>
%ProxyRealm	<p>Specifies the realm to which the authentication must be proxied. If ProxyRealm is not set, Routed Proxy does not occur.</p>
%ProxyUserName	<p>Specifies the User-Name attribute, which must be sent in the proxy request. If ProxyUserName is not set, the User-Name from the original request packet is used.</p> <p>NOTE: Enter the value for %ProxyUserName in capital letters.</p>

Table 148: *.aut [Results] Syntax (*continued*)

Parameter	Function
%Alias	<p>Specifies the value returned from this column that is matched with an existing Steel-Belted Radius Carrier Native User entry of the same name.</p> <p>For example, if the value is max1, and a native user called max1 exists in the Steel-Belted Radius Carrier database, then any return list or check list attributes, as well as any concurrent connection limit configured for max1, are applied to the user's connection.</p> <p>If you want to apply concurrent connection limits to users who are being authenticated by means of SQL, you must set up a Native User entry with no password.</p> <p>NOTE: Use of %Alias is not recommended. Instead, use %Profile.</p> <p>The %LoginLimit value lets you implement the concurrent connection limits previously available through %Alias.</p> <p>Generally, even if a very large number of users resides in the SQL database, you need to add only one or two Native User entries to the Steel-Belted Radius Carrier database. The concurrent connection limit associated with a single Native User entry may be applied to any number of users in the SQL database. Often a Native User entry with a connection limit of 1, and a second Native User entry with a connection limit of 2, is sufficient for an entire SQL database.</p> <p>For example, analog users may be allowed a connection limit of 1, while ISDN users are allowed a connection limit of 2.</p> <p>NOTE: The Native User authentication method displayed in the Authentication Methods page does not need to be activated for the %Alias feature to work.</p>
%FullName	<p>The value returned from this column is interpreted as the full name of the user. This feature is often used to distinguish the user's full name from the actual User-Name sent in the Access-Request.</p>
RADIUS attributes	<p>Any RADIUS attribute (preceded by an @) can be returned from the database and mapped into the [Results] section. Use attribute names as they appear in the appropriate .dct or .jdict (subattributes) files.</p>

Consider the following **SELECT** statement:

```
SELECT user_pwd, attribs, fullname FROM rasusers WHERE user_id = %name
```

where **user_pwd**, **attribs**, **fullname**, and **user_id** are the names of columns in the SQL table, and **rasusers** is the name of the SQL table itself. The [Results] section of this configuration file must map the SQL table columns **user_pwd**, **attribs**, and **fullname** to authentication or authorization data types, or both; for example.

```
[Results]
Password=1
Profile=2
FullName=3
```

Columns in the SQL query are identified in the [Results] section by number; 1 represents the first column in the **SELECT** query (from left to right), and if other columns are also referenced, 2 represents the second, and 3 the third.

Along with a number representing the column order, each entry in the [Results] section also specifies the storage format of the column in the SQL table, using the same slash (/), length, and type conventions as the SQL query.

Default [Results] Parameters

The **DefaultResults** flag in the [Settings] section of **radsql.auth_ora.so** specifies whether default values for **Password**, **Profile**, **Alias**, and **FullName** are automatically bound to the returned SQL data. The default **radsql.auth_ora.so** file sets it to 0.

With **DefaultResults=0**, the results list is no longer automatically bound, and only explicit columns in the [Results] section, or embedded Parameters to a stored procedure, are used. This is the recommended setting.

The **DefaultResults=1** option remains only for backward-compatibility with old **.aut** files that rely on the default results behavior to ensure that the set of default columns are automatically bound.

[FailedSuccessResultAttributes] Section

The [FailedSuccessResultAttributes] section ([Table 149 on page 409](#)) of the SQL authentication configuration file can be used to map any RADIUS attribute returned from the database. Attributes can be specified in two ways:

- Attributes can be specified with a literal value enclosed in single quotes. Values must be enclosed with single quotes, even when they represent numeric values.
- Attributes can be specified with a numeric value that corresponds to the ordering of values returned from the SQL **select** statement.

Precede attribute names with @ and enter them as they appear in the dictionary (.dict) files, or the subattribute dictionary (.jdict) files. Enclose attribute values (including integers and IP addresses) in single quotes. For example:

```
[FailedSuccessResultAttributes]
@Reply-Message = 'Please re-enter your password.'
@Filter-Id = '3'
```

[Failure] Section

The [Failure] section of the SQL authentication configuration file can be used to determine the result of the authentication process (accept or reject) when connectivity to all of the configured SQL databases has failed. For example:

```
[Failure]
Accept = 1
Profile = XYZ
FullName = Unauthenticated!
```

NOTE: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 149: *.aut [Failure] Syntax

Parameter	Function
Accept	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier returns an Access-Accept packet with the Profile and any combination of the FullName and Alias attributes specified in the corresponding [Failure] section parameters. • If set to 0, the user is rejected.
Profile	Specifies the name of a Steel-Belted Radius Carrier profile whose check list and return list attributes are applied to the user's connection.
FullName	By indicating a FullName , Steel-Belted Radius Carrier returns a value in the class attribute, allowing for all [Failure] connections to be accounted.

[Strip] Sections

The [Strip] sections ([Table 150 on page 411](#)) of the SQL authentication configuration file ([Table 150 on page 411](#)) allow User-Name stripping to occur. These sections enable Steel-Belted Radius Carrier to identify the username that the SQL database expects by stripping the incoming User-Name attribute value of realm names and other “decorations.”

You may or may not need to employ User-Name stripping for SQL authentication. Your need for this feature depends upon the naming conventions that you employ on your network and in your SQL database entries. The SBR Carrier usual name parsing features work independently of this feature.

The following [Strip] syntax is available to enable and configure User-Name stripping for SQL authentication:

```
[Strip]
Authentication=Yes

[StripPrefix]
String
String
.
.
.
[StripSuffix]
String
String
.
.
.
```

Table 150: *.aut [Strip] Syntax

Parameter	Function
Authentication	<ul style="list-style-type: none"> • If set to No, prefix and suffix stripping is disabled for authentication. • If set to Yes, prefix and suffix stripping is enabled for authentication packets. When an authentication packet comes into the Steel-Belted Radius Carrier server and a SQL authentication method is active, stripping of the incoming UserName attribute value occurs before SQL authentication: <ol style="list-style-type: none"> a. Prefixes listed in the [StripPrefix] section are stripped from the incoming UserName attribute value. b. Suffixes listed in [StripSuffix] are stripped. c. Any other name processing that is appropriate at this point (for example, tunnel or proxy name parsing) is performed. d. The fully stripped name is authenticated against the SQL database.
[StripPrefix]	<p>Lists strings that are to be stripped from the beginning of the UserName value. The strings are listed in order of priority. A string that appears earlier in the list takes precedence over later strings.</p> <p>In the following example, if the incoming UserName is <i>seattleUser201</i>, the stripped name is <i>User201</i>. If the incoming UserName is <i>seatac2000</i>, the stripped name is <i>tac2000</i>:</p> <p>[StripPrefix]</p> <p>seattle</p> <p>sea</p>

Table 150: *.aut [Strip] Syntax (continued)

Parameter	Function
<i>String</i>	<p>Each <i>String</i> that you provide in a [Strip] section may be a character string, or a regular expression according to the following rules:</p> <p>? is a wildcard character.</p> <p>A dash (-) indicates a range of alphanumeric characters; brackets must enclose lists of characters or ranges. For example, [A-Za-z] means any letter and [0-9.,] means any number, including decimal points and commas.</p> <p>A backslash (\) followed by a non-alphanumeric character indicates that character literally, for example \' indicates the question mark.</p> <p>\ is also used as an escape character:</p> <p>\a bell (7)</p> <p>\b backspace (8)</p> <p>\t tab (9)</p> <p>\n newline (10)</p> <p>\v vertical tab (11)</p> <p>\f formfeed (12)</p> <p>\r return (13)</p> <p>\xnn hex value, where nn are 2 hex digits</p> <p>\nnn decimal value, where nnn are 3 decimal digits</p>
[StripSuffix]	<p>Lists strings that are to be stripped from the end of the User-Name value. Conventions are the same as for [StripPrefix].</p>

Example: SQL Authentication Configuration File

This section provides an example of an Oracle **radsql.aut** configuration file.

```
[Bootstrap]
Enable=0
LibraryName=radsql_auth_ora.so
InitializationString=SQL-ORACLE
```

```

[Settings]
Connect=dBusername/dBpassword
SQL=SELECT password, profile FROM userlist WHERE name = %name/40
ParameterMarker=?
MaxConcurrent=1
ConcurrentTimeout=30
WaitReconnect=2
MaxWaitReconnect=360
PasswordFormat = 0
DefaultResults = 0
;ShutdownTimeout=360
ErrorMap=oracle.ini
LogLevel=0

[Server]
s1=2
s2=2

[Server/s1]
Connect=db_admin1/db_password1

[Server/s2]
Connect=db_admin2/db_password2

[Results]
Password=1/48
Profile=2/48
Alias=2/48

[Failure]
Accept=0
Profile=xyz
FullName=Remote User

```

SQL Accounting

This section describes the configuration file that configures SQL accounting in Steel-Belted Radius Carrier. The configuration files reside in the *radiusdir* directory.

The configuration file used to configure SQL accounting methods must have a *.acc* extension: for example, *radsql.acc*. The format of a configuration file is comparable to that of a Windows INI file: composed of

several sections; section names are enclosed in brackets; each section may contain multiple parameter/value pairs.

Table 151: SQL Accounting Configuration File

File	Function
radsql.acc	Configures Oracle SQL accounting for Steel-Belted Radius Carrier.
radsqljdbc.acc	Configures JDBC SQL accounting for Steel-Belted Radius Carrier.

[Settings] Section

The [Settings] section of the SQL accounting configuration file defines parameters that control the database connection.

[Type] Section

Each entry in the [Type] section of the SQL accounting configuration file maps an AcctStatusType attribute value to a statement name that you may assign arbitrarily. The statement name is then used to look up another section in the configuration file that describes that statement. The secondary section names are composed as [Type/ **statement**], where **statement** is the arbitrarily assigned name for the statement.

For example, to perform separate accounting updates for network access server and user activity, you might provide the following [Type] and [Type/ **statement**] sections:

```
[Type]
1=user
2=user
3=user
7=nas
8=nas
639=nas
28=nas

[Type/user]
SQL=INSERT INTO usagelog \
(Time, NASAddress, SessionID, \
Type, Name, BytesIn, BytesOut) \
VALUES \
(%TransactionTime/t, %NASAddress, \
@Acct-Session-Id, @Acct-Status-Type, \
%FullName/40s, @Acct-Input-Octets, \
@Acct-Output-Octets)
```

```
[Type/nas]
SQL=INSERT INTO ...
```

Note the numeric values used in the preceding [Type] section. The AcctStatusType values 1, 2, 3, 7, and 8 have been reserved by the RADIUS accounting standard with names and meanings, as described in [Table 152 on page 415](#).

Table 152: Acct-Status-Type Values

Acct-Status-Type Value	Name	Meaning
1	Start	A user session has started.
2	Stop	A user session has stopped, request contains final statistics.
3	Interim	A user session is in progress, request contains current statistics.
7	Accounting-On	The network access server has started.
8	Accounting-Off	The network access server is about to shut down.

Additional values for Acct-Status-Type have been defined by network access server vendors for use with their equipment. These vendor-specific values may also be listed in the [Type] section.

[Type/statement] Sections

[Table 153 on page 416](#) lists the parameters that may be present in a [Type/ *statement*] section of the SQL accounting configuration file.

Table 153: *.acc [Type|statement] Syntax

Parameter	Function
SQL	<p>Specifies the exact SQL statement used to update the SQL database with accounting information. The SQL statement may be broken over several lines by ending each line with a backslash. The backslash must be preceded by a space character, and followed by a newline. The subsequent lines may be indented for better readability.</p> <p>The general syntax of the SQL statement is as follows:</p> <p>%<item>/[size][.digits][type][&][!direction]</p> <p>Or</p> <p>@<attribute>/[size][.digits][type][&][!direction]</p> <p>Where, <> indicates required fields and [] indicates optional fields.</p> <p>For example:</p> <pre>SQL=INSERT INTO accounting\ (TransTime, FullName, Authenticator, NASName, \ NASAddress, Type, PacketsIn, PacketsOut) \ VALUES (%TransactionTime/t, %FullName/40s, \ %AuthType/40s, %NASName/40s, %NASAddress, \ %Type, @Acct-Input-Packets/n, \ @Acct-Output-Packets/n)</pre> <p>NOTE: Include the /t (timestamp) data type qualifier with the %TransactionTime argument in SQL statements. If you do not, the %TransactionTime output is formatted as character, with differing results on JDBC and Oracle.</p> <p>See “SQL Parameter” on page 393 for more information.</p>

Table 153: *.acc [Type|statement] Syntax (*continued*)

Parameter	Function
MaxConcurrent	<p>If present, MaxConcurrent overrides the value of MaxConcurrent specified in the [Settings] section for this particular statement.</p> <p>NOTE: The MaxConcurrent parameter is valid only for Oracle SQL accounting (radsql.acc). It is not valid for JDBC SQL accounting (radsqljdbc.acc).</p> <p>NOTE: A setting of MaxConcurrent = 1 is sufficient for all but the most demanding environments. Increase this value slowly and conservatively. For more information about executing overlapping SQL statements, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>
ConcurrentTimeout	<p>If present, ConcurrentTimeout overrides the value of ConcurrentTimeout specified in the [Settings] section for this particular statement.</p>

[TypeNames] Section

Each entry in the [TypeNames] section of the SQL accounting configuration file maps an **AcctStatusType** attribute value to a string. If a %Type parameter is present in the corresponding SQL statement, this %Type parameter contains the given string.

If no string is given for a particular AcctStatusType, when an accounting request of that type is received, %Type is set to the numeric value of the AcctStatusType attribute, formatted as a string.

The syntax for the [TypeNames] section is:

```
[TypeNames]
TypeID=TypeName
TypeID=TypeName
.
.
.
```

You can include RADIUS standard and vendor-specific accounting packet types; for example:

```
[TypeNames]
1=Start
```

```

2=Stop
3=Interim
7=On
8=Off
639=AscendType
28=3ComType

```

Example: SQL Accounting Configuration File

This section provides an example of an Oracle **radsql.acc** configuration file.

```

[Bootstrap]
Enable=0
LibraryName=radsql_acct_ora.so
InitializationString=SQL-ORACLE-ACCT

[Settings]
Connect=dbusername/dbpassword@servicename
ParameterMarker=?
MaxConcurrent=1
ConcurrentTimeout=30
WaitReconnect=2
MaxWaitReconnect=360
ShutdownTimeout=360
ErrorMap=oracle.ini
LogLevel=0

[Server]
s1=2
s2=2

[Server/s1]
Connect=admin1/passwd1@mydb1

[Server/s2]
Connect=admin2/passwd2@mydb2

[Type]
1=User
2=User
3=User

```

```
[Type/User]
SQL=INSERT INTO accounting (TransTime, FullName, Authenticator, NASName, NASAddress,
  Type, PacketsIn, PacketsOut) \ VALUES (%TransactionTime, %FullName/40s,
  %AuthType/40s, %NASName/40s, %NASAddress, %Type, @Acct-Input-Packets/n,
  @Acct-Output-Packets/n)
```

SQL Accessors

You can use an external SQL database to authorize subscribers. The **sqlaccessor.gen** file stores the settings needed by the SQLAccessor plug-in to authorize subscribers. SQLAccessor requires three items of information from the database:

- IMSI
- MSISDN
- Authorization String

This section describes the configuration choices that you need to make to configure **sqlaccessor.gen** or **sqlaccessor_jdbc.gen**.

The **sqlaccessor.gen** file stores the settings used by the SQL data accessor plug-in. It is composed of several sections. Section names are enclosed in square brackets.

NOTE: For information about the configuration items that are common across all SQL plug-ins, see [“Common Configuration Items” on page 386](#).

NOTE: The databases used must support stored procedures.

NOTE: Oracle front-end applications are not supported on a Linux platform. The **sqlaccessor.so** and **sqlaccessor.gen** files are specific to Oracle plug-ins and must not be installed on a Linux platform. You must instead use the **sqlaccessor_jdbc.gen** file.

[Settings] Section

The [Settings] section of the **sqlaccessor.gen** file defines parameters that control the database connection.

Table 154: [Settings] Section

Field	Description
MethodName	Identifies the name under which the data accessor registers itself with Steel-Belted Radius Carrier. Default value is SQL Accessor.
Driver	(JDBC only) Specifies the third-party JDBC driver to load for authentication. For example: Driver=com/mysql/jdbc/Driver NOTE: Third-party JDBC drivers must be installed in the <i><JRE-path>/lib/ext</i> directory. Where, <i><JRE-path></i> indicates the path where the JRE (that is integrated with SBR Carrier) is installed in your system. Refer to the JDBC driver documentation for information about how to install the JDBC driver and supporting files.
ConnectDelimiter	Specifies the character used to separate fields (DSN, UID, PWD) in the connect string. Default value is ; (semicolon). If the connect string requires use of semicolons as part of a field value, you can use this parameter to specify a different delimiter, such as ^ (caret).

[Results] Section

The [Results] section maps the position of a column name in the SELECT SQL statement with the data needed.

[Failure] Section

The [Failure] section of the **sqlaccessor.gen** file can be used to determine the result of the authentication process (accept or reject) when connectivity to all of the configured SQL databases has failed.

Table 155: [Failure] Section

Setting	Description
Accept	<ul style="list-style-type: none"> • If set to 1, Steel-Belted Radius Carrier returns an Access-Accept packet with the Profile and any combination of the FullName and Alias attributes specified in the corresponding [Failure] section parameters. • If set to 0, the user is rejected.
Profile	Specifies the name of a Steel-Belted Radius Carrier profile whose check list and return list attributes are applied to the user's connection.

Table 155: [Failure] Section (*continued*)

Setting	Description
Fullname	By indicating a FullName , Steel-Belted Radius Carrier returns a value in the class attribute, allowing for all [Failure] connections to be accounted.

Example: SQL Accessor Configuration File

This section provides an example of an Oracle **sqlaccessor.gen** configuration file.

```
[Bootstrap]
LibraryName=radsql_accessor_ora.so
Enable=0

[Settings]
MethodName=SQLAccessor
MethodName=SQL Accessor
Connect=username/password@servicename
SQL=SELECT user, msisdn, authstring FROM my_database WHERE user=@KeyToRecord
ParameterMarker=?
MaxConcurrent=1
ConcurrentTimeout=30
WaitReconnect=2
MaxWaitReconnect=360
MaxHardErrorRetries=0

[Server]
s1=2
s2=2

[Server/s1]
Connect=admin1/passwd1@mydb1

[Server/s2]
Connect=admin2/passwd2@mydb2

[Server/s3]
Connect=admin3/passwd3@mydb3
LastResort = 1

[Results]
ResultIMSI = 1/16
ResultMSISDN = 2/16
```

```
ResultAuthString = 3/16
```

```
[Failure]
```

```
Accept=0
```

```
Profile=xyz
```

```
FullName=Remote User
```

Detailed Use Cases

This section provides a few use cases on the SQL Plug-ins.

Working with Stored Procedures

A stored procedure is a sequence of SQL statements that form a logical unit and perform a particular task. You can use stored procedures to encapsulate a set of queries or operations that can be executed repeatedly on a database server. For example, you can code operations on an employee database, such as password lookup, as stored procedures that can be executed by application code. For more information about stored procedures, see the *SBR Carrier Administration and Configuration Guide*.

The SQL example in the previous section can be replaced by a custom stored procedure. This stored procedure might look something like the following:

```
PROCEDURE myProc
(
  ttime in varchar2,
  nasaddr in varchar2,
  sessid in varchar2,
  ttype in varchar2,
  uname in varchar2,
  bytein in varchar2,
  byteout in varchar2
);
END myProc;
CREATE OR REPLACE PACKAGE BODY myPack1 IS
PROCEDURE myProc
(
  ttime in varchar2,
  nasaddr in varchar2,
  sessid in varchar2,
  ttype in varchar2,
```

```

uname in varchar2,
bytein in varchar2,
byteout in varchar2
)
IS
BEGIN
INSERT INTO usagelog
( Time, NASAddress, SessionID, Type, Name,
BytesIn, BytesOut )
VALUES
( ttime, nasaddr, sessid, ttype, uname, bytein,
byteout );
END myProc;
END myPack1;

```

When you invoke the stored procedure, delineate each parameter as an input (!i), output (!o), or input/output (!io) variable.

This stored procedure can be invoked with the following connect string in the **radsql.acc** file:

```

SQL=BEGIN myPack1.myProc(%TransactionTime!i,
%NASAddress!i, @Acct-Session-Id!i, %Type!i,
%FullName!i, @Acct-Input-Packets!i,
@Acct-Output-Packets!i); END;

```

SQL Database Data Retrieval Methods

There are two methods for retrieving the required data items (IMSI, MSISDN, and Authorization String) from the SQL database:

- **SQL=SELECT Statement Method**
For specific information, see [“SQL=SELECT Method for Data Retrieval from SQL Databases” on page 423](#).
- **Stored Procedure Method**
For specific information, see [“Stored Procedure Method for Data Retrieval from SQL Databases” on page 426](#).

SQL=SELECT Method for Data Retrieval from SQL Databases

The SQL=SELECT method for retrieving the IMSI, MSISDN, and Authorization String from the SQL database involves including a SQL=SELECT statement and a corresponding [Results] section in the **sqlaccessor.gen** or **sqlaccessor_jdbc.gen** file.

SQL=SELECT Method: SELECT Statement

Place a SQL=SELECT Statement in the [Settings] section of `sqlaccessor.gen` or `sqlaccessor_jdbc.gen` to retrieve the IMSI, MSISDN, and Authorization String.

In the following example, the IMSI, MSISDN, and Authorization String are selected from a subscriber database in which the IMSI is the key. (The user column contains IMSI values.)

```
SQL=SELECT user, msisdn, authstring FROM my_database WHERE
user=@KeyToRecord
```

NOTE: You can also retrieve IMSI, MSISDN, and Authorization String from the subscriber database by setting the MSISDN as the key. In this case, the SELECT statement would be:

```
SQL=SELECT user, msisdn, authstring FROM my_database WHERE
msisdn=@KeyToRecord
```

SQL=SELECT Method: [Results] Section

The [Results] section declares output container variables and maps them to the columns in the SQL query result set. Columns in the SQL query are identified in the [Results] section by the position number in the SQL query and maximum number of characters in the SQL database. For example:

```
[Results]
ResultIMSI = 1/16
ResultMSISDN = 2/16
ResultAuthString = 3/16
```

It is mandatory that you define values for all three output variables (ResultIMSI, ResultMSISDN, and ResultAuthString). The SQLaccessor module may fail to work properly if you leave any of the output variables blank.

If the SQL database does not contain an IMSI column or a MSISDN column, then the SQL query would not include this unavailable column in the SQL=SELECT statement. However, a value should be set for the unavailable output variable in the [Results] section instead of being left blank. This value should be the same as that of the value set for the output variable of the key field.

The following example explains how to retrieve Authorization String from the SQL database by setting the IMSI as the key if the database only contains the IMSI and Authorization String columns. In this case, the SELECT statement and the [Results] section would be:


```
SQL=SELECT user, authstring FROM my_database WHERE user=@KeyToRecords
```

```
[Results]
```

```
ResultIMSI = 1/16
```

```
ResultMSISDN = 1/16
```

```
ResultAuthString = 2/16
```

SQL=SELECT Method: Section Correlations Illustrated

The **sqlaccessor.gen** file indicates an SQL statement that uses the IMSI contained in @KeyToRecord to match a value in the user column, thereby retrieving an associated row of values in the user, msisdn, and authstring columns. Therefore, the user column must contain IMSI values. The **sqlaccessor.gen** file [Results] section indicates that the values of the user, msisdn, and authstring columns are returned in the temporary attributes @ResultIMSI, @ResultMSISDN, and @ResultAuthString. Therefore, the user, msisdn, and authstring columns must contain IMSI, MSISDN, and authorization string values respectively. The temporary attributes described in this example are fixed and specific to the SIM feature in combination with the SQL Accessor plug-in.

[Figure 12 on page 426](#) illustrates the correlation between the SQL=SELECT statement, the [Results] section, the SQL database, and the key identified in **gsmmap.gen**.

ResultIMSI=1/16 indicates that the first column in the SQL=SELECT statement, the column named user contains the value of ResultIMSI that is retrieved. In [Figure 12 on page 426](#), the user column contains IMSI values, which are integer values that uniquely identify the cellular network of the user.

ResultMSISDN=2/16 indicates that the second column in the SQL=SELECT statement, the column named msisdn contains the value of ResultMSISDN that is retrieved. In [Figure 12 on page 426](#), the msisdn column contains MSISDN values, which are integer values that uniquely identify a subscription in a GSM or an UMTS network.

ResultAuthString=3/16 indicates that the third column in the SQL=SELECT statement, the column named authstring contains the value of ResultAuthString that is retrieved. In [Figure 12 on page 426](#), the authstring column contains authorization key string. Using this string, the correct profile is fetched and assigned to a particular IMSI or MSISDN.

NOTE: The column headings need not be user, msisdn, or authstring. However, they must be mapped to ResultIMSI, ResultMSISDN, and ResultAuthString as shown in [Figure 12 on page 426](#).

In [Figure 12 on page 426](#), the line KeyForAuthorization=IMSI in the **gsmmap.gen** file indicates that the temporary attribute @KeyToRecord contains IMSI values to be used as the key in obtaining further information for the user to be authenticated.

Figure 12: Relationship Between Sections in sqlaccessor.gen File

gsmmap.gen

```
[SQLDatabase]
ModuleType=Database
DatabaseAccessorMethodName=SQL Accessor
KeyForAuthorization=IMSI
:
```

sqlaccessor.gen

```
[Select]
SQL = SELECT user, msisdn, authstring FROM my_database WHERE user=@KeyToRecord
:

[Results]
ResultIMSI = 1/16
ResultMSISDN = 2/16
ResultAuthString = 3/16

my_database


| user            | msisdn       | authstring      | st_address      |
|-----------------|--------------|-----------------|-----------------|
| 214070123456789 | 358405627015 | substr1:substr2 | 1 main street   |
| 325070181234567 | 469516738126 | abc:def         | 5 school street |
| 436171810234567 | 680738950358 | xyza:fghi       | 7 park street   |


```

Stored Procedure Method for Data Retrieval from SQL Databases

You can use a stored procedure, rather than a **SQL=SELECT** statement, to retrieve the IMSI, MSISDN, and Authorization String from the database for use by the SQLAccessor plug-in.

The stored procedure must be created in the Oracle database before using it in Steel-Belted Radius Carrier.

Simauth requires the IMSI, MSISDN, and Authorization String in the format used by the MAP gateway. However, the SQL database schema might not allow these strings to be obtained in the expected format. Therefore, the SQLAccessor module can use a stored procedure to convert the database information to the expected format.

Stored Procedure Method: BEGIN Statement of sqlaccessor.gen

Include a **SQL=BEGIN** statement in the [Settings] section of **sqlaccessor.gen** or **sqlaccessor_jdbc.gen** to convert the data from the database to the output parameters, ResultIMSI, ResultMSISDN, and ResultAuthString.

Example:

```
SQL=BEGIN SIM_Server_stored_proc.produce_return_vals(@KeyToRecord!i,
@ResultIMSI!o, @ResultMSISDN!o, @ResultAuthString!o); END;
```

Stored Procedure Method: [Results] Section of `sqlaccessor.gen`

The stored procedure converts and maps SQL values to the variables listed in the [Results] section. If you are using the stored procedure method, include this [Results] section in `sqlaccessor.gen` or `sqlaccessor_jdbc.gen` exactly as shown here.

```
[Results]
ResultIMSI
ResultMSISDN
ResultAuthString
```

Stored Procedure Method: Database Schema

The database schema must exist for the database key and the data to be retrieved from the database.

Example:

```
imsi VARCHAR2(32)
msisdn VARCHAR2(32)
auth_string VARCHAR2(32)
subscriber_id VARCHAR2(32)
```

(The column named user contains IMSI values.)

Stored Procedure Method: Data Retrieval

The stored procedure must retrieve the values for the IMSI, MSISDN, and Authorization String from the database and return them in the values of ResultIMSI, ResultMSISDN, and ResultAuthString.

Stored Procedure Method: Example

The following lines retrieve the values for IMSI, MSISDN, and Authorization String and place them in the output parameters ResultIMSI, ResultMSISDN, and ResultAuthString. The database columns are named msisdn, authstring, and user, where user contains IMSI values. The IMSI values are the key values.

```
.
.
.

CREATE OR REPLACE PACKAGE SIM_Server_stored_proc IS
PROCEDURE produce_return_vals(

KeyToRecord IN VARCHAR2,
ResultIMSI OUT VARCHAR2,
ResultMSISDN OUT VARCHAR2,
ResultAuthString OUT VARCHAR2) IS
.
.
```

```
.  
.   
-- The cursor holds the result of the query.  
SELECT * FROM subscribers WHERE subscriber_id=KeyToRecord;  
. -- Execute the query  
OPEN cur;  
FETCH cur INTO row;  
.   
.   
.   
  
-- If the row was found then convert the data  
  
IF( cur%FOUND ) THEN  
  
ResultMSISDN : = row.msisdn;  
ResultIMSI : = row.imsi;  
ResultAuthString: = row.auth_string;  
.   
.   
. 
```

LDAP Plug-Ins

IN THIS CHAPTER

- Overview | 429
- Common Configurations | 429
- LDAP Authentication | 444
- LDAP Accessor Files | 460

This chapter explains the LDAP authentication and accessor plug-ins.

Overview

This chapter describes the LDAP plug-ins used to configure LDAP authentication in Steel-Belted Radius Carrier. The following topics are included in this chapter:

NOTE: Throughout this chapter, the term *attributes* refers to both standard RADIUS attributes and structured attributes. For information about specifying structured attributes, see [“Structured Attributes” on page 199](#).

Common Configurations

This section explains the common configurations of LDAP plug-ins.

[Bootstrap] Section

The [Bootstrap] section ([Table 156 on page 430](#)) of the LDAP configuration file specifies information that Steel-Belted Radius Carrier uses to load and start the LDAP method.

Table 156: [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the LDAP module, for example, ldapauth.so or ldapaccessor.so.
Enable	<ul style="list-style-type: none"> • If set to 1, the LDAP module is enabled. • If set to 0, the LDAP module is disabled. Default value is 0.
InitializationString	Specifies the identifier for the LDAP module. The name of each LDAP module must be unique. Default value is LDAP.

[Settings] Section

The [Settings] section of the LDAP configuration file forms a basis for all Bind and Search requests to the LDAP database server(s).

Search sequencing is flexible. You can override search results using the **\$reject** and **\$accept** keywords.

You can proceed to a new search even if the current search returns no data by using the **OnNotFound** parameter.

For examples of using flexible searching, see “[[Server/name](#)] Sections” on page 437.

The parameters in the [Settings] section apply to all LDAP servers listed in the configuration file. The following parameters are usually present. If any of these parameters is not provided in the [Settings] section, the parameter assumes a system default value.

The values set in [Settings] for some parameters, such as **ConnectTimeout**, **MaxConcurrent**, or **WaitReconnect**, provide defaults that apply to all servers. These default values can be overridden for a particular server by entering the same parameter with a different value in a [[Server/ name](#)] section.

Table 157: [Settings] Syntax

Parameter	Function
MaxConcurrent	<p>Specifies the maximum number of LDAP requests that may be executing at one time.</p> <p>You can enter the value in the range from 1 through 500. Default value is 1.</p> <p>NOTE: The value specified in this parameter can be overridden in individual [Server/<i>name</i>] sections of this file.</p> <p>NOTE: A setting of MaxConcurrent = 1 is sufficient for all but the most demanding environments. Increase this value slowly and conservatively. For more information about executing overlapping LDAP statements, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>
Timeout	<p>Specifies the maximum number of seconds for the overall timeout for each request, which includes the delay in acquiring resources, attempts against multiple LDAP servers, and so forth.</p> <p>Default value is 20 seconds.</p>
ConnectTimeout	<p>Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is passed to the client database engine, which may or may not implement the feature.</p> <p>Default value is 25 seconds.</p> <p>NOTE: The value specified in this parameter can be overridden in individual [Server/<i>name</i>] sections of this file.</p>
QueryTimeout	<p>Specifies the timeout value in seconds for an individual search performed against the LDAP server.</p> <p>Default value is 10 seconds.</p>
WaitReconnect	<p>Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.</p> <p>NOTE: The value specified in this parameter can be overridden in individual [Server/<i>name</i>] sections of this file.</p>

Table 157: [Settings] Syntax (*continued*)

Parameter	Function
MaxWaitReconnect	<p>Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection.</p> <p>WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.</p> <p>NOTE: The value specified in this parameter can be overridden in individual [Server/<i>name</i>] sections of this file.</p>
BindName	<p>For BindName, you must omit the Bind parameter from the LDAP configuration file. Use the BindName and BindPassword parameters instead.</p> <p>In the [Settings] section, BindName and BindPassword specify a default LDAP Bind template to use for all servers. You can also use BindName and BindPassword in [Server/<i>name</i>] sections to override this default for an individual server</p> <p>See “[Server/<i>name</i>] Sections” on page 437.</p>
LogLevel	<p>Activates logging for the LDAP method and sets the level of detail of the message. This value may be the number 0, 1, or 2, where 0 is the lowest logging level, 1 is intermediate, and 2 is the most verbose.</p> <p>If the LogLevel that you set in the configuration file is different than the LogLevel in radius.ini, the lower value of the setting controls.</p>
UpperCaseName	<ul style="list-style-type: none"> • If set to 0, preserves the case of the username. • If set to 1, converts the username to uppercase. <p>Default value is 0.</p>
PasswordCase	<ul style="list-style-type: none"> • If set to U or Upper, the password returned from the LDAP database is converted to uppercase before authentication. • If set to L or Lower, the password is converted to lowercase. • If set to O or Original, the password is not altered before authentication. <p>Default value is Original.</p>

Table 157: [Settings] Syntax (*continued*)

Parameter	Function
PasswordFormat	<p>By default, the PasswordFormat parameter is not listed in the [Settings] section of the LDAP configuration file. With no listing, Steel-Belted Radius Carrier expects the user's password in the LDAP table is in cleartext format.</p> <p>If you want to configure Steel-Belted Radius Carrier to automatically handle password values correctly when it detects that they have been encrypted using UNIXcrypt or a SHA1+Base64 hash, set PasswordFormat to auto.</p>
Search	Specifies an LDAP Search request by referencing a [Search/ <i>name</i>] section elsewhere in the same *.aut file.
SSL	<ul style="list-style-type: none"> • If set to 0, SSL is not used over the LDAP connection. • If set to 1, SSL is used over the LDAP connection. <p>Default value is 0.</p> <p>NOTE: The value specified in this parameter can be overridden in individual [Server/<i>name</i>] sections of this file.</p> <p>If SSL=1, then the Host parameter in [Server/<i>name</i>] accepts LDAP-style URIs. For example, ldaps://hostname:port.</p>
MaxScriptSteps	<p>Specifies the maximum number of statements that a script can execute before terminating. You can use the MaxScriptSteps parameter to make sure a script does not get caught in an infinite loop.</p> <p>Default value is 10000.</p>
ScriptTraceLevel	<p>Specifies the level of detail for line-by-line script tracing in the log.</p> <ul style="list-style-type: none"> • If set to 0, no traces are logged. • If set to 1, traces are only logged when the SbrTrace() function is executed by the script. • If set to 2, a trace is generated for every line executed by the script. <p>Default value is 0.</p>

Table 157: [Settings] Syntax (continued)

Parameter	Function
ShutdownTimeout	<p>Specifies the maximum number of seconds to wait for outstanding database transactions when the server is in the process of shutting down. If this timeout expires, then any outstanding database transactions are forcibly terminated in order to allow the server to shut down.</p> <p>Default value is 180 seconds.</p> <p>NOTE: Changing the default value is not recommended.</p>
FilterSpecial CharacterHandling	<ul style="list-style-type: none"> • If set to 1, specifies that non-alphanumeric characters, such as ('), is converted to an ASCII hex value preceded by a backslash when they are encountered in a username during authentication. • If set to 0, non-alphanumeric characters are not converted during authentication. <p>Default value is 1.</p> <p>In support of RFC 2254, the following substitutions are made when set:</p> <ul style="list-style-type: none"> • replace '(' with "\\28" • replace ')' with "\\29" • replace '*' with "\\2a" • replace '\' with "\\5c"

Table 157: [Settings] Syntax (*continued*)

Parameter	Function
FlashReconnect	<ul style="list-style-type: none"> • If set to 1, SBR Carrier attempts to reconnect to an LDAP database server when the LDAP database server goes down. When this setting is enabled, SBR Carrier immediately attempts to reconnect to the LDAP database server if a Bind or a Search operation fails and sends an Access-Reject if the connection attempt is unsuccessful. <p>NOTE: If SBR Carrier has not successfully connected to the LDAP database server since startup, SBR Carrier directly rejects the request without performing any Bind or Search operation even if this parameter is set to 1.</p> <ul style="list-style-type: none"> • If set to 0, SBR Carrier sends an Access-Reject before the reconnection is attempted. <p>This setting applies to all servers. To apply this setting for a particular server, configure the FlashReconnect parameter in the [Server/<i>name</i>] section.</p> <p>Default value is 1.</p>
LdapVersion	<p>Specifies the version of LDAP protocol.</p> <p>Default value is 2.</p>
OnFound	<p>Specifies the next request section when data is found. The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/<i>name</i>] section elsewhere in the same .aut file. If there is no next request section, the overall operation succeeds. This can be overridden using the \$reject keyword, which causes the operation to fail when data is found.</p>
OnNotFound	<p>Specifies the next request section when data is not found. The value of this parameter is a string, name. The name specifies an LDAP Search request by referencing a [Search/<i>name</i>] section elsewhere in the same .aut file. If there is no next request section, the overall operation fails. This can be overridden using the \$accept keyword, which causes the operation to succeed when data is not found.</p>
Password	<p>Specifies the password string, which can include variables, used to specify a Bind before any search within a request. If this parameter is not specified, the packet's password is used.</p>

Table 157: [Settings] Syntax (*continued*)

Parameter	Function
UTC	<ul style="list-style-type: none"> • If set to 0, time values are displayed using the local time. • If set to 1, time values are displayed using UTC (GMT).

[Server] Section

The [Server] section lists the LDAP servers that may be used to perform authentication. You can specify more than one server in the [Server] section for load-balancing or backup. When more than one server is specified, Steel-Belted Radius Carrier authenticates against these databases in a round-robin fashion.

The syntax is:

```
[Server]
ServerName=TargetNumber
ServerName=TargetNumber
.
.
.
```

where **ServerName** is the name of a configuration file section that contains configuration information for that server, and **TargetNumber** is an activation target number, a number that controls when this server is activated for backup purposes. **TargetNumber** is optional and may be left blank. For example:

```
[Server]
s1 =
s2 =
[Server/s1]
.
. ;Connection details for server s1
.
[Server/s2]
.
. ;Connection details for server s2
```

A Steel-Belted Radius Carrier server maintains connectivity with its LDAP servers according to the following rules:

- The priority of the server by order. The first entry in the [Server] section has the highest priority.
- By activation target number. The rule for the activation target is that if the number of LDAP servers that Steel-Belted Radius Carrier is connected to is less than the activation target, Steel-Belted Radius Carrier connects to the server and includes it in the round-robin list. While the number of active servers is equal

to or greater than the activation target, Steel-Belted Radius Carrier does not use that server in the round-robin list. An activation target of 0 indicates that, in the current configuration, this machine is never used.

[Server/name] Sections

Several sections of the LDAP configuration file work together to configure the connection between the Steel-Belted Radius Carrier server and the LDAP database server. The sections are [Server], [Server/**name**], and [Settings].

Each [Server/**name**] section of the LDAP configuration file contains configuration information about a single LDAP server. You must provide a [Server/**name**] section for each server named in the [Server] section. For example:

```
[Server]
s1=
s2=

[Server/s1]
Host = ldap_1
Port = 389
.
.
.
[Server/s2]
Host = 130.4.67.1
LastResort = 1
.
.
.
```

[Table 158 on page 438](#) lists the settings that may be present in a [Server/ **name**] section:

Table 158: [Server/name] Syntax

Parameter	Function
Bind	<p>For Bind, you must specify a Bind template in the [Settings] section of the LDAP configuration file.</p> <p>The Bind template must follow conventional LDAP syntax. It may be as simple or as complex as LDAP syntax permits, with multiple attribute/value assertions in Boolean combination. It may also include replacement variables from the Variable Table.</p> <p>Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Bind request, the value of the variable replaces the variable name.</p> <p>For example, a Bind template that uses the User-Name attribute from the RADIUS request might look like this:</p> <pre>uid=<User-Name>, ou=Special Users, o=bigco.com</pre>
BindName	<p>For BindName, the BindName parameter specifies the distinguished name (DN) to be used in the Bind request that connects to the LDAP server. The [Server/<i>name</i>] section lets you specify a unique BindName for a specific server. Use the [Settings] section to specify a default BindName to use for all servers.</p> <p>For Bind, omit all Bind, BindName and BindPassword parameters and use the Bind parameter in the [Settings] section.</p> <p>See “[Settings] Section” on page 430.</p>
BindPassword	<p>For BindName, you must provide a BindPassword. The BindPassword specifies the password to be used in the Bind request that connects to the LDAP server. The [Server/<i>name</i>] section lets you specify a unique BindPassword for a specific server. Use the [Settings] section to specify a default BindPassword to use for all servers.</p> <p>For Bind, omit the BindName and BindPassword parameters. Use the Bind parameter instead.</p>
Certificates	<p>Specifies the path of the certificate database for use with SSL. This path must not end in a filename.</p>
ConnectTimeout	<p>Specifies the number of seconds to wait when attempting to establish the connection to the database before timing out. This value is passed to the client database engine, which may or may not implement the feature.</p>

Table 158: [Server/name] Syntax (*continued*)

Parameter	Function
FlashReconnect	<ul style="list-style-type: none"> • If set to 1, SBR Carrier attempts to reconnect to an LDAP database server when the LDAP database server goes down. When this setting is enabled, SBR Carrier immediately attempts to reconnect to the LDAP database server if a Bind or a Search operation fails and sends an Access-Reject if the connection attempt is unsuccessful. <p>NOTE: If SBR Carrier has not successfully connected to the LDAP database server since startup, SBR Carrier directly rejects the request without performing any Bind or Search operation even if this parameter is set to 1.</p> <ul style="list-style-type: none"> • If set to 0, SBR Carrier sends an Access-Reject before the reconnection is attempted. <p>This setting applies to a particular server. To apply this setting for all servers, configure the FlashReconnect parameter in the [Settings] section.</p> <p>Default value is 1.</p>
Host	<p>The hostname or IP address of the LDAP server.</p> <p>NOTE: For SSL configurations, the host name field accepts only LDAP-style URIs. For example, ldaps://hostname:port.</p>
LastResort	<p>You may identify a <i>last resort</i> LDAP server by providing a LastResort parameter in one of these [Server/<i>name</i>] sections, and setting its value to 1. If an LDAP query against some other server results in no record found, the server tries the last resort server before accepting or rejecting the user.</p> <p>You might use the LastResort parameter to identify your primary accounts database. This enables Steel-Belted Radius Carrier to cover the case in which a user account is newly added but has not yet been propagated to all the LDAP databases.</p>
LdapVersion	<p>Specifies the version of LDAP protocol, if needed to override the default given in the [Settings] section.</p>

Table 158: [Server/name] Syntax (*continued*)

Parameter	Function
MaxConcurrent	<p>Specifies the maximum number of LDAP requests that may be executing at one time.</p> <p>You can enter the value in the range from 1 through 1000. Default value is 1.</p> <p>NOTE: A setting of MaxConcurrent = 1 is sufficient for all but the most demanding environments. Increase this value slowly and conservatively. For more information about executing overlapping LDAP statements, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>
MaxWaitReconnect	<p>Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the database connection.</p> <p>WaitReconnect specifies the time to wait after failure of the database connection. This value is doubled on each failed attempt to reconnect, up to a maximum of MaxWaitReconnect.</p>
Password	<p>Specifies the password string, which can include variables, used to specify a Bind before any search within a request. If this parameter is not specified, the packet's password is used.</p>
Port	<p>The TCP port of the LDAP server, or 0 to use the standard port.</p> <p>Default value is 0.</p> <p>NOTE: For SSL configurations, the default port setting is ignored and the LDAP-style URI for Host is applied. For example, ldaps://hostname:port.</p>
QueryTimeout	<p>Specifies the number of seconds to wait for the execution of an LDAP request to complete before timing out. This value is passed to the database engine, which may or may not implement the feature.</p>
SSL	<ul style="list-style-type: none"> • If set to 0, SSL is not used over the LDAP connection. • If set to 1, SSL is used over the LDAP connection. <p>Default value is 0.</p>
WaitReconnect	<p>Specifies the number of seconds to wait after a failure of the database connection before trying to connect again.</p>

[Search/DoLdapSearch] Sections

Each [Search/*name*] section (Table 159 on page 441) in the LDAP configuration file specifies the complete details of one LDAP Search request. You can use the same Search request on various databases, because the details of the database connection are specified separately.

For BindName, you must ensure that each [Search/*name*] section searches for a database entry that matches the incoming username and retrieves from it an attribute containing that user's password. Steel-Belted Radius Carrier must compare this password to the one it received in the incoming AccessRequest packet.

A [Search/*name*] section may retrieve other LDAP attributes as well; however, if you are authenticating with BindName, the user's password is a minimum requirement. Use the Attributes parameter to specify the list of items you want returned.

For example:

```
[Search/DoLDAPSearch]
Base = ou=Special Users, o=bigco.com
Scope = 1
Filter = uid=<User-Name>
Attributes = InterestingAttributes
Timeout = 20
%DN = dn

[Attributes/InterestingAttributes]
User-Secret
RADIUS-Profile
Inactivity-Timeout

[Response]
%Password = User-Secret
%Profile = RADIUS-Profile
Vendor-Specific-NAS-Attribute = Inactivity-Timeout
```

Table 159: [Search/*name*] Syntax

Parameter	Function
%DN	Specifies a variable into which the distinguished name that results from the Search is placed.
Attributes	Specifies the LDAP attributes relevant to Steel-Belted Radius Carrier, by referencing an [Attributes/ <i>name</i>] section elsewhere in the same .aut file.

Table 159: [Search/name] Syntax (*continued*)

Parameter	Function
Base	<p>Specifies the distinguished name (DN) of the entry that serves as the starting point for the search. This filter is a template for an LDAP distinguished name string. The filter follows conventional LDAP syntax and may be as simple or as complex as LDAP syntax permits. It may also include replacement variables from the Variable Table.</p> <p>Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Search request, the value of the variable replaces the variable name.</p>
OnFound	<p>Specifies the next request section when data is found. The value of this parameter is a string, <i>name</i>. The <i>name</i> specifies an LDAP Search request by referencing a [Search/<i>name</i>] section elsewhere in the same .aut file. If there is no next request section, the overall operation succeeds. This can be overridden using the \$reject keyword, which causes the operation to fail when data is found.</p>
OnNotFound	<p>Specifies the next request section when data is not found. The value of this parameter is a string, <i>name</i>. The <i>name</i> specifies an LDAP Search request by referencing a [Search/<i>name</i>] section elsewhere in the same configuration file. If there is no next request section, the overall operation fails. This can be overridden using the \$accept keyword, which causes the operation to succeed when data is not found.</p>
Search	<p>(Optional) Specifies an LDAP Search request by referencing a [Search/<i>name</i>] section elsewhere in the same configuration file. Steel-Belted Radius Carrier tries this Search request next, if the current Search yields no result. Each [Search/<i>name</i>] section may contain at most one Search parameter.</p>

Table 159: [Search/name] Syntax (*continued*)

Parameter	Function
Filter	<p>Specifies the filter to apply to the search. This filter is a template for an LDAP Search string. The filter follows conventional LDAP syntax and may be as simple or as complex as LDAP syntax permits, with multiple attribute/value assertions in Boolean combination. It may also include replacement variables from the Variable Table.</p> <p>Each replacement variable consists of the variable name enclosed in angle brackets (<>). Upon execution of the LDAP Search request, the value of the variable replaces the variable name.</p> <p>For example, a Search template that uses the User-Name and Service-Type attributes from the RADIUS request might look like this:</p> <p>(&(uid = <User-Name>)(type = <Service-Type>))</p>
Scope	<p>Specifies the scope of the search; 0 (search the base), 1 (search all entries one level beneath the base), or 2 (search the base and all entries beneath the base at any level).</p>

The Search parameter can be used in one [Search/ *name*] section after another to create a serial chain of Search requests. Every Search in the *chain* is tried. If any Search fails to return data, the Access-Request is rejected.

An example of a two-part chained Search follows:

```
[Settings]
Search = DoLdapSearch

[Search/DoLdapSearch]
Base = ...
Filter = ...
Search = GetMoreLdapInfo

[Search/GetMoreLdapInfo]
Base = ...
Scope = ...
Filter = ...
```

Search sequencing is flexible. You can proceed to a new search even if the current search returns no data by using the **OnNotFound** parameter. You can override search results using the **\$reject** and **\$accept** keywords. The following is an example of flexible searching:

```

[Search/DoSearch2]
Base = o=xyz.com
Scope = 2
Filter = uid=<User-Name>
Attributes = AttrList
Timeout = 20
%DN = dn
OnFound = DoSearch8
OnNotFound = DoSearch9

[Search/DoSearch8]
Base = o=xyz.com
Scope = 2
Filter = uid=<User-Name>
Attributes = AttrList
Timeout = 20
%DN = dn
OnFound = DoSearch9
OnNotFound = DoSearch9

[Search/DoSearch9]
Base = o=xyz.com
Scope = 2
Filter = uid=<User-Name>
Attributes = AttrList
Timeout = 20
%DN = dn
OnNotFound = $accept

```

LDAP Authentication

The LDAP authentication configuration file is located in the same directory that contains the Steel-Belted Radius Carrier daemon. The configuration file must have the extension **.aut** and is usually called **ldapauth.aut**.

An LDAP authentication configuration file consists of several sections, where each section may contain multiple entries. Section names are enclosed in square brackets, for example [Bootstrap]. Each entry in the section appears on one line, and is of the form ***parameter = value***. A section ends at the next section, or at the end of the file. Everything to the right of a semicolon (;) is ignored until the end of that line.

When Steel-Belted Radius Carrier extracts RADIUS attribute values from the incoming Access-Request and adds them to the Variable Table, the name that it gives to each variable is the same as the name of the corresponding attribute, for example User-Name or Calling-Station-ID. You may refer to the variable

by this name in any subsequent entry in the **.aut** configuration file. This convention means that RADIUS attribute names are treated as reserved keywords. However, the **.aut** configuration file syntax also permits you to assign the value of an incoming RADIUS attribute to any variable.

When the LDAP Search request returns LDAP attribute values, they are added to the Variable Table. Steel-Belted Radius Carrier gives each variable the name of the corresponding LDAP attribute. This schema produces variable names such as User-Secret and Last-Name. For the names to use in your own **.aut** configuration file, consult your LDAP database schema. Like RADIUS attribute names, LDAP attribute names are treated as reserved keywords. However, the **.aut** configuration file syntax permits you to assign the value of a returned LDAP attribute to any variable.

LDAP Authentication Variable Names

When Steel-Belted Radius Carrier extracts RADIUS attribute values from the incoming Access-Request and adds them to the Variable Table, the name that it gives to each variable is the same as the name of the corresponding attribute, for example User-Name or Calling-Station-ID. You may refer to the variable by this name in any subsequent entry in the **.aut** configuration file. This convention means that RADIUS attribute names are treated as reserved keywords. However, the **.aut** configuration file syntax also permits you to assign the value of an incoming RADIUS attribute to any variable.

When the LDAP Search request returns LDAP attribute values, they are added to the Variable Table. Steel-Belted Radius Carrier gives each variable the name of the corresponding LDAP attribute. This schema produces variable names such as User-Secret and Last-Name. For the names to use in your own **.aut** configuration file, consult your LDAP database schema. Like RADIUS attribute names, LDAP attribute names are treated as reserved keywords. However, the **.aut** configuration file syntax permits you to assign the value of a returned LDAP attribute to any variable.

[Bootstrap] Section

The [Bootstrap] section ([Table 160 on page 446](#)) of the LDAP authentication configuration file specifies information that Steel-Belted Radius Carrier uses to load and start the LDAP Authentication plug-in.

After you edit **ldapauth.aut** and restart Steel-Belted Radius Carrier, the **InitializationString** value that you entered in the [Bootstrap] section of **ldapauth.aut** appears in the **Authentication Methods** page in Web GUI. You can then enable, disable, or prioritize this method like any other entry in the list.

You can configure more than one LDAP authentication method. Each requires its own **.aut** file in the same directory as **ldapauth.aut**. The [Bootstrap] section of each **.aut** file must provide a **LibraryName** of **ldapauth.so**. The **InitializationString** in each **.aut** file must be unique, so that you can distinguish between authentication methods in the **Authentication Methods** page.

Table 160: *.aut [Bootstrap] Syntax

Parameter	Function
AcceptsAuthorizeOnly	<ul style="list-style-type: none"> • If set to 1, Authorize-Only requests are accepted for this authentication method, when all of the following conditions are true: <ul style="list-style-type: none"> • The Access-Request contains the Service-Type attribute with a value=Authorize-Only • Message-Authenticator is present and valid • A session already exists in SBR Carrier for the AAA session ID (WiMAX) • AuthorizeOnly=1 in the [Configuration] section of radius.ini • If set to 0, Authorize-Only requests are not accepted for this authentication method regardless of whether AuthorizeOnly=1 in radius.ini. <p>NOTE: SBR Carrier is only able to process Authorize-Only requests for WiMAX sessions because a session must be located in the current session table, indexed by a WiMAX session Id.</p>

[Settings] Section

The [Settings] section ([Table 161 on page 447](#)) of the LDAP authentication configuration file defines parameters that control the database connection. Refer [Table 157 on page 431](#) for the common parameters of the LDAP configuration file.

The syntax is:

```
[Settings]
DelayConnect=1
```

Table 161: *.aut [Settings] Syntax

Parameter	Function
DelayConnect	<p>Specifies whether to establish a connection to the LDAP server during SBR startup.</p> <ul style="list-style-type: none"> • If set to 0, Steel-Belted Radius Carrier establishes a connection to the LDAP server during SBR startup. • If set to 1, Steel-Belted Radius Carrier does not establish a connection to the LDAP server during SBR startup. Only when an LDAP authentication request is received from a user, Steel-Belted Radius Carrier establishes a connection to the LDAP server. This setting is useful when not all users are required to be authenticated through the LDAP server, for example when JavaScript in the ldapauth.aut file is used to differentiate users. <p>Default value is 0.</p> <p>NOTE: The DelayConnect parameter is applicable only for the Bind authentication.</p>

[JavaScript] Section

The [JavaScript] section [Table 162 on page 447](#) of the LDAP authentication configuration file contains the configuration parameters for the JavaScript engine.

The syntax is:

```
[JavaScript]
;JSEngineRuntimeMemory=8
```

Table 162: *.aut [JavaScript] Syntax

Parameter	Function
JSEngineRuntimeMemory	<p>Sets the size of the runtime memory arena from which new instances of JavaScript engines are allocated for the LDAP server.</p> <p>NOTE: LDAP and core SBR Carrier use independent instances of JavaScript engines.</p> <p>Default value is 8 MB.</p> <p>NOTE: Increasing the value of JSEngineRuntimeMemory will decrease the frequency of garbage collection but negatively affect performance.</p>

[Attributes/*name*] Sections

LDAP database entries may have many attributes, many of which may be irrelevant to the authentication process. An LDAP Search returns all of the attributes associated with an LDAP entry. Therefore, when specifying an LDAP Search for authentication purposes, you may want to provide a list of specific LDAP attributes relevant to Steel-Belted Radius Carrier. Only these attributes are placed in the Variable Table.

Each [Attributes/ ***name***] section in the LDAP authentication configuration file lists LDAP attributes relevant to a specific LDAP Search request. The syntax is:

```
[Attributes/name]
attribute
attribute
.
.
.
```

where ***attribute*** is the name of an LDAP attribute and ***name*** is an arbitrary name for the section. You must type the ***attribute*** names exactly as they appear in your LDAP database schema. Use one line per attribute. For example:

```
[Attributes/InterestingAttributes]
User-Secret
RADIUS-Profile
Inactivity-Timeout
```

An [Attributes/ ***name***] section is associated with a Search request by referencing it from within a [Search/ ***name***] section using the Attributes parameter. For example:

```
[Search/DoLdapSearch]
Attributes = InterestingAttributes
```

If the Attributes parameter is omitted from a [Search/ ***name***] section, Steel-Belted Radius Carrier retains all of the attributes associated with the LDAP entry. Of these attributes, Steel-Belted Radius Carrier uses only those referenced in the **.aut** configuration file; all others stay in the Variable Table until the authentication transaction is complete and the table is discarded.

For BindName authentication, you must ensure that the [Attributes/ ***name***] section lists the attribute in which the user's password is stored and that your [Response] section assigns the value of this attribute to the outgoing **%Password** parameter. Steel-Belted Radius Carrier completes authentication by comparing the returned **%Password** value with the password that arrived in the Access-Request. For example:


```
[Attributes/InterestingAttributes]
User-Secret
RADIUS-Profile
Inactivity-Timeout

[Response]
%Password = User-Secret
%Profile = RADIUS-Profile
Vendor-Specific-NAS-Attribute = Inactivity-Timeout
```

[RejectResponse] Section

This [RejectResponse] section defines the attributes you want to return in an Access-Reject message.

The [RejectResponse] section syntax

```
[RejectResponse]
attribute = variable
attribute = variable
.
.
.
```

where **attribute** is the name of a RADIUS attribute you want to include in the Access-Reject message, and **variable** is the name of a variable in the Variable Table. The end result of the [RejectResponse] syntax is that the value in the variable is assigned to the attribute.

Example:

```
[Response]
Kineto-UMA-Reg-Reject-Cause = Kineto-UMA-Reg-Reject-Cause
Service-Type = Service-Type
```

```
[RejectResponse]
Kineto-UMA-Reg-Reject-Cause = Kineto-UMA-Reg-Reject-Cause
Filter-ID = Filter-ID
```

[Response] Section

During an authentication transaction, the [Response] section is the last section in the LDAP authentication configuration file to be processed. At this point in processing, all Bind and Search requests to the LDAP database have been completed.

The [Response] section instructs Steel-Belted Radius Carrier what to do with the information that it has retrieved from the incoming Access-Request and from the LDAP database. The goal at this point is for Steel-Belted Radius Carrier to complete authentication and issue an Access-Response to the RADIUS client.

The [Response] section syntax ([Table 163 on page 450](#)) is:

```
[Response]
attribute = variable
attribute = variable
.
.
.
```

where **attribute** is the name of a RADIUS attribute or other special item needed to complete authentication, and **variable** is the name of a variable in the Variable Table. The end result of the [Response] syntax is that the value in the variable is assigned to the attribute.

An IP pool can be returned for any attribute of the appropriate type. If the returned string appears to be an IP address (that is, in the format, **a.b.c.d**), it is considered an IP address; otherwise, it is considered an address pool, from which an IP address is allocated.

attribute may be the name of a RADIUS attribute, or it may be one of the following keywords, which identify various special items associated with Steel-Belted Radius Carrier. Each of these keywords begins with the percent sign (%) to distinguish it clearly from the RADIUS attributes.

NOTE: The Framed-IPv6-Address attribute is a single-value attribute, which can appear only once in the [Response] section.

Table 163: *.aut [Response] Syntax

Item	Function
%LoginLimit	The name of the variable specifying the Maximum Concurrent Connection limits.

Table 163: *.aut [Response] Syntax (*continued*)

Item	Function
%Password	<p>For BindName authentication, you must provide a %Password entry in the [Response] section and you must assign it the value of the password attribute retrieved from the LDAP database. Steel-Belted Radius Carrier validates the password received in the AccessRequest by comparing it with the value assigned to %Password. If the passwords do not match, the request is rejected.</p> <p>NOTE: The user's password may be in clear-text, or encrypted with UNIXcrypt or a SHA1+Base64 hash.</p> <p>For Bind authentication, omit %Password. When processing reaches the [Response] section, the password has already been validated.</p>
%Profile	<p>The name of a Profile entry in the Steel-Belted Radius Carrier database.</p> <p>If the password has been validated (by BindName or Bind), with %Profile listed in the [Response] section, %Profile may be set to any variable, for example:</p> <p style="padding-left: 40px;">%Profile = userpolicy</p> <p>When the search filter is set to find a user or object in the LDAP database that includes the userpolicy LDAP attribute, this value is retrieved and returned to the Steel-Belted Radius Carrier database so that it may be matched with an existing Profile entry of the same name. If the userpolicy LDAP attribute is multi-valued, the first value of userpolicy is used and subsequent values are ignored.</p> <p>If the value of userpolicy is "prof1" and a Profile called prof1 exists in the Steel-Belted Radius Carrier database, any return list or check list attributes in prof1 are applied to the user's connection.</p> <p>If the value returned from LDAP cannot be matched with an existing Profile in the Steel-Belted Radius Carrier database, the user is rejected due to "Insufficient Resources."</p>
%ProxyRealm	<p>The realm to which the authentication must be proxied. If ProxyRealm is not set, Routed Proxy does not occur.</p>
%ProxyUserName	<p>The User-Name attribute, which must be sent in the proxy request. If ProxyUserName is not set, the User-Name from the original request packet is used.</p> <p>NOTE: Enter the value for %ProxyUserName in capital letters</p>

Table 163: *.aut [Response] Syntax (*continued*)

Item	Function
%Alias	<p>The name of a Native User entry in the Steel-Belted Radius Carrier database.</p> <p>If the password has been validated (by BindName or Bind), with %Alias listed in the [Response] section, %Alias may be set to any variable, for example:</p> <p>%Alias = userpolicy</p> <p>Important: We strongly recommend you use %Profile, because use of %Alias has been deprecated.</p> <p>The %LoginLimit value lets you implement the concurrent connection limits previously available through %Alias.</p> <p>Generally, even if a very large number of users reside in the LDAP database, you need to add only one or two Native User entries to the Steel-Belted Radius Carrier database. The concurrent connection limit associated with a single Native User entry may be applied to any number of users in the LDAP database. Often a Native User entry with a connection limit of 1, and a second Native User entry with a connection limit of 2, is sufficient for the entire LDAP database.</p> <p>For example, analog users may be allowed a connection limit of 1, while ISDN users are allowed a connection limit of 2.</p> <p>NOTE: The Native User authentication method displayed in the Authentication Methods page does not need to be activated for the Alias feature to work.</p>
%FullName	<p>The fully distinguished name of the User, for Steel-Belted Radius Carrier accounting purposes. This is the exact name against which authentication was performed. Depending on what may have occurred during Steel-Belted Radius Carrier name parsing, this name may or may not be different from the value of the User-Name attribute as it originally arrived in the Access-Request.</p>

[Request] Section

The [Request] section ([Table 164 on page 453](#)) of the LDAP authentication configuration file indicates which RADIUS attribute values Steel-Belted Radius Carrier extracts from the incoming Access-Request.

Steel-Belted Radius Carrier places these values in the Variable Table before moving on to the LDAP Bind and Search requests indicated in the file.

The syntax is:

```
[Request]
attribute = variable
attribute = variable
.
.
.
```

where **attribute** is the name of a RADIUS attribute or other special item associated with the incoming Access-Request, and **variable** is the name of a variable in the Variable Table. The end result of the [Request] syntax is that the value in the incoming attribute is assigned to this variable.

attribute may be the name of a RADIUS attribute, or it may be one of the following keywords, which identify various special items also associated with the connection request. Each of these keywords begins with the percent sign (%) to strongly distinguish it from the RADIUS attributes.

Table 164: *.aut [Request] Syntax

Item	Function
%OriginalUserName	The original full identification of the user, before any processing (that is, user@realm).
%User	The user portion of OriginalUserName (the section before @).
%UserName	The full user identification (user and realm strings) after all stripping and processing has been performed.
%Name	Synonym for UserName.
%EffectiveUser	The name of the user (the section before @) as presented to the authentication method. This may be a modified version of the original username.
%Realm	The realm portion of the original user identification (the section after @) as presented to the authentication method. This may be a modified version of the original realm name.
%EffectiveRealm	The realm portion of the user identification as presented to the method. This may be a modified version of the original realm name.

Table 164: *.aut [Request] Syntax (*continued*)

Item	Function
%NASName	The name of the network access server that originated the request. This may be the name of the RADIUS Clients entry in the database or the value of the NAS-Identifier or NAS-IP-Address attribute.
%NASAddress	The address of the NAS, in dotted notation.
%NASModel	The make/model of the NAS, as specified in the Steel-Belted Radius Carrier database.
%Password	The PAP password. NOTE: The %Password function cannot be used in a filter.
%AllowedAccessHours	The time periods in which the user is allowed to access the network.
%RADIUSClientName	The name of the network access server, as specified in a RADIUS Clients entry in the Steel-Belted Radius Carrier database.

variable may be omitted from any [Request] entry. If so, the value in the incoming **attribute** is assigned to a variable named **attribute**.

```
[Request]
attribute =
```

In the following [Request] section example, the **nasid** variable receives the value of the NAS-Identifier attribute from the request packet, the Service-Type variable receives the value of the Service-Type attribute, and the **%NASAddress** variable receives the NAS address in dotted notation.

```
[Request]
NAS-Identifier = nasid
Service-Type =
%NASAddress =
```

[Defaults] Section

The [Defaults] section of the LDAP authentication configuration file lets you add entries to the variable table before the request is processed. You can reference these variables in your query, even if they are not supplied in the request. Any variable not listed in the [Defaults] section is initialized to a null value.

The format of each [Defaults] entry is:

```
variable = value
```

where **variable** is the name of a variable and **value** is the value you want to assign to it. For example:

```
[Defaults]
Default-User=SSStudent

[Search/Radius]
Base = ou=people,dc=funk,dc=com
Filter = uid=<Default-User>
Scope = 2
Attributes = RadiusAttrs
Timeout = 20
%Dn = dn
```

In this example, the **Default-User** variable is not created during request processing by the LDAP plug-in. Instead, the **Default-User** variable is inserted into the variable table by the entry in the [Defaults] section, and then substituted into the Filter setting in the [Search/Radius] section.

You can use the [Defaults] section to specify values for any variable, including temporary variables and those that represent RADIUS attributes or LDAP attributes. This way, if the Access-Request packet and LDAP database do not provide Steel-Belted Radius Carrier with all of the values that it needs to respond to an Access-Request, in each case it has an acceptable alternative value that can be used instead.

You can store multiple values for any variable; if that variable is mapped to a RADIUS attribute, all values are returned in the RADIUS response. Multiple entries set within this section are considered multiple values of the same variable.

Variable values are not additive between this section and each search. Therefore, if a search returns one or more values, all current values are replaced.

NOTE: The [Defaults] section is the only section in the configuration file that lets you assign static values to variables.

[Failure] Section

The [Failure] section of the LDAP authentication configuration file ([Table 165 on page 456](#)) can be used to determine the result of the authentication process (accept or reject) when connectivity to all of the configured LDAP databases has failed. For example:

```
[Failure]
Accept = 1
Profile = XYZ
FullName = Mr Stanley Smith
```

NOTE: The Profile option and the Alias option cannot be used together. Read the following descriptions and choose the one that suits your needs.

Table 165: *.aut [Failure] Syntax

Parameter	Function
Accept	<ul style="list-style-type: none">• If set to 1, Steel-Belted Radius Carrier returns an Access-Accept packet with the Profile, FullName, and/or Alias attributes specified in the corresponding [Failure] section parameters.• If set to 0, the user is rejected.
Profile	This is the name of an existing Steel-Belted Radius Carrier Profile entry, whose check list and return list attributes are applied to the user's connection.
FullName	This string is the full username, which is used in the Class attribute in the Access-Accept message.

Table 165: *.aut [Failure] Syntax (*continued*)

Parameter	Function
Alias	<p>As an alternative to using the Profile parameter, you can use the Alias parameter to name an existing Steel-Belted Radius Carrier Native User entry. Steel-Belted Radius Carrier then applies the check list and return list attributes of this User entry to the user's connection.</p> <p>NOTE: The Alias feature permits the Maximum Concurrent Connection limit (settable in the Users page) to be applied to the user's connection.</p> <p>Important: We strongly recommend you use Profile, because use of Alias has been deprecated. The LoginLimit value lets you implement the concurrent connection limits previously available through Alias.</p> <p>If you want to apply concurrent connection limits to users who are being authenticated by means of LDAP, you must set up a Native User entry specifically for this purpose, with all of the appropriate check list and return list attributes, and with no password. You can set up as many such accounts as you require. These entries store a specific set of check list and return list attributes for LDAP authentication, for use only with the Alias parameter.</p> <p>NOTE: Native User entries without passwords cannot be authenticated. This is a safety feature built into Steel-Belted Radius Carrier. Therefore, setting up User entries in preparation for using the Alias parameter with LDAP authentication does not pose a back door security risk.</p> <p>NOTE: The Native User authentication method displayed in the Authentication Methods page does not need to be activated for the Alias feature to work.</p>

Grouped Attributes

This section explains the use of the following LDAP grouped attributes:

- **ProfileData:** Stores multiple RADIUS attribute value pairs within a single LDAP container, removing the need for multiple entries in a LDAP user object.
- **GlobalProfile:** Configures users based on a global profile, which can be specified as any username concatenated with the company name (such as **Profile1@Company**).

GlobalProfile Attribute

The **GlobalProfile** attribute takes the value from an LDAP attribute and parses it to match a profile. The format of the data is that of a DN attribute. Store it as:

```
cn=profile-name, {optional ou's}, o=name,
{optional dc's \o's \c's}
```

profile-name and **name** are concatenated to build **profilename@name**. Make sure this value matches a profile stored in Steel-Belted Radius Carrier. For example:

```
cn=Global1, ou=Profile, ou=Radius, ou=IP Services,
o=acme, o=directoryroot
```

This value is parsed to form a new string: **Global1@acme**. This new string is then passed back as the profile by making the following entry in the response section:

```
[Response]
%profile= LDAP attribute that contains the global profile
```

This value is ignored if:

- There is no o keyword value
- The string does not begin with the cn keyword
- **%profile** is not set to the name of the attribute that contains the **Globalprofile** data

An incorrect profile name results if the **name** parameter is not the first value of the organization name (o).

ProfileData Attribute

This feature allows an administrator to store multiple RADIUS attribute-value pairs within a single LDAP container, removing the need for multiple entries in a LDAP user object.

For example, the values for **framed-ip-address**, **service-type**, and framed-protocol can be stored in one attribute called **stdDialin**. Combining them saves space on the LDAP server.

Make the attribute a string data-type (directory string or string case insensitive). The format for the data stored in this attribute is:

```
<r|R>;attribute-name;type;value&
```

- **r** or **R**—Specifies that the attribute may be single or multi-valued.
- **attribute-name**—Specifies the name of the attribute that is being added.
- **value&**—The value returned with this attribute, terminated with &.

NOTE: The **type** field is ignored; the values are interpreted according to the RADIUS dictionary.

For example:

```
stdDialin: r; service-type; integer; 1&r; framed-protocol; integer; 2& r;
framed-ip-address; string;192.168.2.2&
```

The **Profiledata** attribute is retrieved from the LDAP server in the same way that other attributes are retrieved; they might be specified from the [Attributes\] section referenced in the relevant search.

Have the [Response] section of the **ldapauth.aut** file list each attribute contained in the **profiledata** attribute.

Configure the [Response] section for **stdDialin** to operate:

```
[Response]
service-type=
framed-protocol=
framed-ip-address=
```

NOTE: If the **ProfileData** attribute stores multiple attribute-value pairs and one or more of those attributes appears in the applicable dictionary, then that attribute and its value are returned to the RADIUS client even if the attribute is not enumerated in the [Response] section of the file.

Modifying **ldapauth.aut**

To modify **ldapauth.aut** to support the extensions:

1. Add the following field:

```
[Settings]
UpdateResponse = 1
```

2. Add a [GroupedAttributes] section to specify the **GlobalProfile** or **ProfileData** attributes, or both.

```
[GroupedAttributes]
GlobalProfile = GlobalProfileLDAPAttribute
ProfileData = ProfileDataLDAPAttribute
```

3. In the appropriate [Attributes/ **name**] section, add the actual LDAP attributes as specified previously.

```
[Attributes/name]
GlobalProfileLDAPAttribute
ProfileDataLDAPAttribute
any other attributes
```

4. In the [Response] section, set **%Profile** to the GlobalProfile and list any attributes that are contained in the **ProfileData** attribute:

```
[Response]
%Profile = GlobalProfileLDAPAttribute
radiusattribute1=
radiusattribute2=
```

LDAP Accessor Files

The **ldapaccessor.gen** file stores the settings used by the LDAP data accessor plug-in. The **ldapaccessor.gen** file is composed of several sections. Section names are enclosed in square brackets.

[Settings] Section

The [Settings] section ([Table 166 on page 460](#)) of the **ldapaccessor.gen** file defines parameters that control the database connection.

Table 166: ldapaccessor.gen [Settings] Fields

Field	Description
MethodName	Identifies the name under which the data accessor registers itself with Steel-Belted Radius Carrier. Default value is LDAP Accessor.
Timeout	Specifies the number of seconds that a request waits for execution before it is discarded. Because as many as MaxConcurrent LDAP statements can be executing at one time, new requests must be queued as they arrive until other statements are processed.
ConnectTimeout	Specifies the number of seconds to wait when attempting to establish the connection to the LDAP directory before timing out. This value is passed to the client LDAP directory, which might or might not implement the feature. Default value is 25 seconds.

Table 166: `ldapaccessor.gen` [Settings] Fields (*continued*)

Field	Description
QueryTimeout	<p>Specifies the number of seconds to wait for a response to a query before timing out. This value is passed to the client LDAP directory, which might or might not implement the feature.</p> <p>Default value is 10 seconds.</p>
WaitReconnect	<p>Specifies the number of seconds to wait after a failure of the LDAP directory connection before trying to connect again.</p> <p>Default value is 2 seconds.</p>
MaxWaitReconnect	<p>Specifies the maximum number of seconds to wait after successive failures to reconnect after a failure of the LDAP directory connection.</p> <p>The WaitReconnect setting specifies the time to wait after failure of the LDAP directory connection. This value is doubled on each failed attempt to reconnect, up to the value of the number of seconds specified by the MaxWaitReconnect setting.</p> <p>Default value is 360 seconds (6 minutes).</p>
UpperCaseName	<p>Specifies whether the username is converted to uppercase. Choices are: 0 (preserve the case of the username), 1 (convert username to uppercase).</p> <p>Default value is 0.</p>
Search	<p>The value of this field is a string, name. The name specifies an LDAP Search request by referencing a [Search/name] section elsewhere in the file.</p>
SSL	<p>Specifies whether to use SSL over the LDAP connection. The choices are: 0 (do not use SSL), 1 (use SSL).</p> <p>Default value is 0.</p>

[Request] Section

You must use the [Request] section of **ldapaccessor.gen** to bind the *KeyToRecord* variable provided in the **gsmmap.gen** file to the Key variable used in the LDAP search definitions. The value specified here (Key) must match the value specified in the [Search/DoLdapSearch] section.

See [“Detailed Use Cases” on page 615](#) for more information about key fields.

NOTE: Do not modify the **KeyToRecord** keyword in the [Request] section. The value *KeyToRecord* is hard-coded into the **gsmmap.gen** file.

[Response] Section

The [Response] section of the **ldapaccessor.gen** file maps the information retrieved by the LDAP search to values expected by the gsmmap module. Do not change the [Response] section unless instructed to do so by Juniper Networks Technical Services.

[Attributes/AttrList] Section

The [Attributes/AttrList] section identifies the attributes contained in the LDAP schema. Replace the attribute names in the sample file with the attributes used at your site.

```
[Attributes/AttrList]
```

```
wlanMSISDN  
wlanIMSI  
wlanAuthorization  
wlanPrepayFlag
```

CDR Accounting Plug-Ins

IN THIS CHAPTER

- CDR Process Overview | 463
- Types of Call Detail Records | 464
- Configuring Accounting Options with `cdراعct.acc` | 465
- Displaying CDR Information | 471
- CDR Fields | 475
- CDR Field Formats for Binary and ASN.1 CDR Files | 488

The SIM authentication module for Steel-Belted Radius Carrier include Call Detail Record (CDR) accounting capability. The CDR accounting capability manages all CDR-based subscriber accounting for the purposes of billing. The following topics are included in this chapter:

CDRs can be generated for authentications performed by `simauth.aut`, `ldapauth.aut`, `radsql.aut` or `radsqljdbc.aut`.

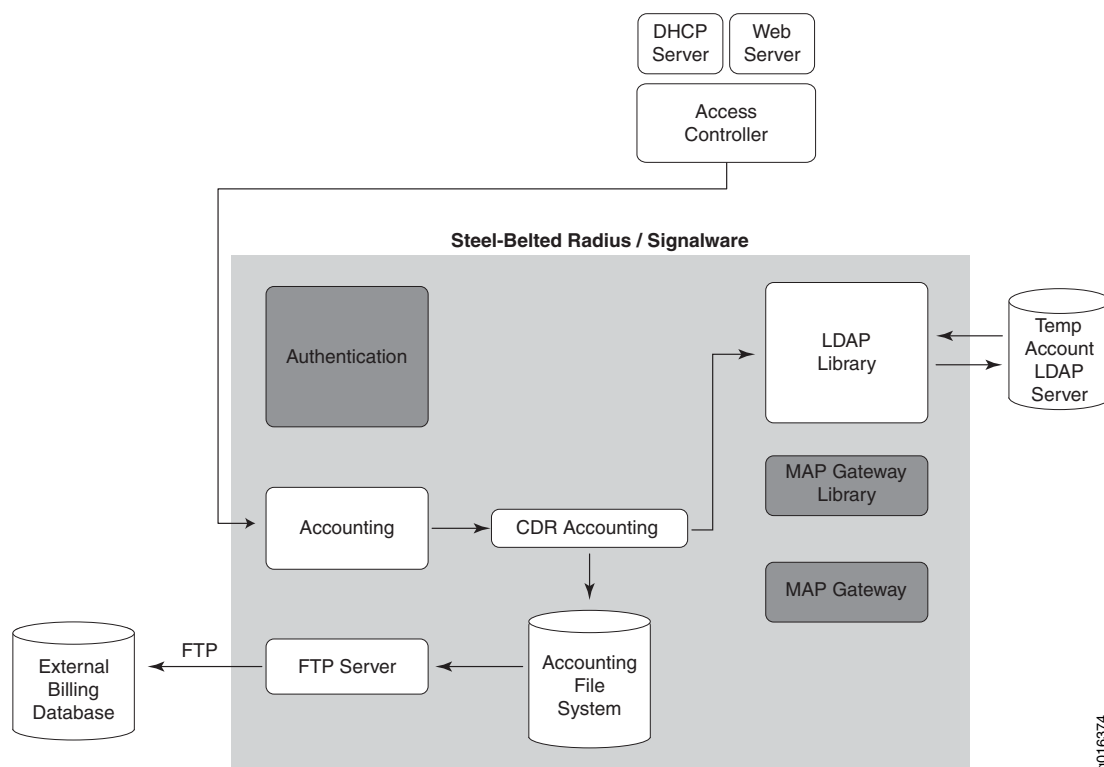
NOTE: Class attributes need to be included in accounting requests in order for CDR accounting to work properly.

CDR Process Overview

The CDR capability manages all CDR-based subscriber accounting for the purposes of billing. CDRs are forwarded to a charging gateway through an FTP server, or other transport method to a billing application.

Figure 13 on page 464 shows the call detail record (CDR) accounting process.

Figure 13: CDR Accounting



The following steps take place after account provisioning and authentication:

1. After it receives a RADIUS Access-Accept message, the Access Controller grants the subscriber access to the network, and sends a RADIUS Accounting-Start message to Steel-Belted Radius Carrier.
2. Steel-Belted Radius Carrier passes the Accounting-Start message to the Call Detail Record (CDR) accounting engine (**cdracct**), and determines whether accounting requests have been previously received for the account.
3. Steel-Belted Radius Carrier records fixed-fee, session, or partial CDRs to the accounting file system.
4. The service provider periodically exports CDR billing information to a billing application.

Types of Call Detail Records

Steel-Belted Radius Carrier can forward call detail records to a billing gateway when subscribers initiate and terminate sessions. Steel-Belted Radius Carrier can produce three types of CDRs:

- **Session CDRs**—Generated at the end of each WLAN session. A session CDR records the length of time of the WLAN session and the number of bytes of data sent and received. Session CDRs can be generated during SIM authentication.
- **Partial CDRs**—Generated periodically during a WLAN session either (a) after a specified amount of time has elapsed or (b) after a session has transferred a specified threshold quantity of data. A provider can specify how frequently a system generates a partial CDR by specifying time and volume (number of bytes) settings in the **cdracct.acc** file. After a partial CDR is generated for a subscriber session because a time or volume threshold is exceeded, both threshold triggers are reset to zero.

Partial CDRs can be generated during SIM authentication. A provider can configure time and volume thresholds for SIM authentication.

NOTE: The Acct-Termination-Cause attribute reports the cause or reason for the subscriber session termination. For more information on the Acct-Termination-Cause codes and the corresponding termination causes, refer the *Standard RADIUS Accounting Attributes* section in the *SBR Carrier Administration and Configuration Guide*.

Configuring Accounting Options with cdracct.acc

You can configure the available options for CDR accounting in the **cdracct.acc** [Settings] section.

[Bootstrap] Section of cdracct.acc

The [BootStrap] section of the **cdracct.acc** file contains the settings listed in [Table 167 on page 465](#).

Table 167: cdracct.acc [Bootstrap] Fields

Field	Description
LibraryName	Specifies the name of the executable binary. The default value is cdracct.so .
Enable	<ul style="list-style-type: none"> • Set to 1 to enable this file. • Set to 0 to disable this file. Default is 0.
InitializationString	The cdracct initialization string. Default is CDRACCT.

Example

```
[Bootstrap]
LibraryName=cdracct.so
Enable=1
InitializationString=CDRACCT
```

[Settings] Section of cdracct.acc

The [Settings] section of the **cdracct.acc** file contains the settings listed in [Table 168 on page 466](#).

Table 168: cdracct.acc [Settings] Fields

[Settings] Field	Description
ConfigLog	<p>Specifies the method for logging cdracct.acc configuration information:</p> <ul style="list-style-type: none">• None= Configuration information is not captured.• ConsoleAndLog= Log information is sent to both the console and the log.• Console= Log information is sent to the console only.• Log= Log information is sent to the log file only. <p>Default is ConsoleAndLog.</p>
CDRDirectory	<p>The directory where the CDR records are stored.</p> <p>The default value is ./CDR under the Steel-Belted Radius Carrier install directory. If you modify this directory name, make sure it exists before you use the SIM authentication module in Steel-Belted Radius Carrier.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 168: cdracct.acc [Settings] Fields (*continued*)

[Settings] Field	Description
CDRNodeID	<p>The name for the authentication server machine in the NodeID field of the generated CDR.</p> <p>If CDRNodeID is omitted or commented out, the correct value is updated automatically by Steel-Belted Radius Carrier on system startup. This value is the Solaris system name for the machine on which Steel-Belted Radius Carrier is running.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
UserPartialCdrEnable	<p>Specifies whether or not partial CDRs are generated for the LDAP authentication plug-in: ldapauth.aut, or the SQL authentication plug-ins: radsql.aut or radsqldb.aut users.</p> <ul style="list-style-type: none"> • 0—Partial CDRs are not generated. • 1—Partial CDRs are generated. <p>The default is 0.</p> <p>If UserPartialCdrEnable=1, then UserSessionCdrEnable must be set to 0.</p>
UserSessionCdrEnable	<p>Specifies whether or not session CDRs are generated for the LDAP authentication plug-in: ldapauth.aut, or the SQL authentication plug-ins: radsql.aut or radsqldb.aut users.</p> <ul style="list-style-type: none"> • 0—Session CDRs are not generated. • 1—Session CDRs are generated. <p>The default is 1.</p> <p>If UserSessionCdrEnable=1, then UserPartialCdrEnable must be set to 0.</p>

Table 168: cdracct.acc [Settings] Fields (*continued*)

[Settings] Field	Description
SIMPartialCdrEnable	<p>Specifies whether or not partial CDRs are generated for SIMauth users.</p> <ul style="list-style-type: none"> • 0—Partial CDRs are not generated. • 1—Partial CDRs are generated. <p>The default is 0.</p> <p>NOTE: If SIMPartialCdrEnable=1, then SIMSessionCdrEnable must be set to 0.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
SIMSessionCdrEnable	<p>Specifies whether or not session CDRs are generated for SIMauth users.</p> <ul style="list-style-type: none"> • 0—Session CDRs are not generated. • 1—Session CDRs are generated. <p>The default is 1.</p> <p>NOTE: If SIMSessionCdrEnable=1, then SIMPartialCdrEnable must be set to 0.</p>
VolumeThresholdMegaBytes	<p>Specifies the threshold (in megabytes) for creating a partial CDR.</p> <p>Default is 10 MB.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
TimeThresholdSeconds	<p>Specifies the threshold (in seconds) for creating a partial CDR.</p> <p>Default is 600 seconds (10 minutes).</p> <p>NOTE: The value entered for TimeThresholdSeconds must match the value specified for the Acct-Interim-Interval return list attribute for the user (or the profile assigned to the user) in Steel-Belted Radius Carrier.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 168: cdracct.acc [Settings] Fields (*continued*)

[Settings] Field	Description
VolumeThresholdEnable	<p>Specifies whether partial CDRs are generated when the volume threshold is crossed.</p> <ul style="list-style-type: none"> • 0—Partial CDRs are not generated. • 1—Partial CDRs are generated. <p>The default is 0.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
TimeThresholdEnable	<p>Specifies whether partial CDRs are generated when the time threshold is crossed.</p> <ul style="list-style-type: none"> • 0—Partial CDRs are not generated. • 1—Partial CDRs are generated. <p>The default is 0.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
CdrDownlink	<p>Specifies the RADIUS attribute to which the CDR Downlink field is mapped. Options are:</p> <ul style="list-style-type: none"> • Acct-Input-Octets • Acct-Output-Octets <p>The default is Acct-Input-Octets.</p> <p>NOTE: The CDR Uplink field is automatically mapped to whichever attribute is not assigned to the Downlink field.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 168: cdracct.acc [Settings] Fields (*continued*)

[Settings] Field	Description
CdrType	<p>Specifies the format for the CDR type.</p> <ul style="list-style-type: none"> • BinaryV1—Version 1 type CDRs are generated with extension <code>.cdr1</code>. • BinaryV2—Version 2 type CDRs are generated with extension <code>.cdr2</code>. • Asn1V2—ASN.1 type CDRs are generated with extension <code>.cdr2a</code>. <p>If CdrType is not specified, Version 1 type CDRs are generated with extension <code>.cdr</code>. Files with both <code>.cdr</code> and <code>.cdr1</code> extensions are identical (version 1). The <code>.cdr</code> extension is retained for backward compatibility.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
DefaultCUIDType	<p>Specifies the user id to be used when no CUID (ChargeableUserId) attribute is received with the accounting request. This setting applies only if CdrType is set to BinaryV2 or Asn1V2.</p> <p>Allowed values are:</p> <ul style="list-style-type: none"> • IMSI • MSISDN • NAI <p>The default value is NAI.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Example

```
[Settings]
ConfigLog=ConsoleAndLog
CDRDirectory=./CDR
CDRNodeID=
SIMPartialCdrEnable=0
SIMSessionCdrEnable=1
UserPartialCdrEnable=0
UserSessionCdrEnable=1
```

```
VolumeThresholdMegaBytes=10
TimeThresholdSeconds=600
VolumeThresholdEnable=0
TimeThresholdEnable=0
CdrDownlink=Acct-Input-Octets
CdrType=BinaryV2
DefaultCUIDType=NAI
```

NOTE: The **radius.ini** file installed with Steel-Belted Radius Carrier must contain the following lines to ensure that the MSISDN is transported to the **cdracct.acc** plug-in.

```
[EmbedInClass]
Funk-SS7-MSISDN=Encrypt,Remove
```

Displaying CDR Information

CDRs are created periodically (for partial CDRs), at the end of a session (for session CDRs), or at logon (for fixed fee). You set fields in the [Settings] section of **cdracct.acc** to generate CDRs within sessions. See [Table 168 on page 466](#) for a list of settings that affect CDR creation.

CDR Files

CDR filenames are assigned to CDRs with the filename incremented by one for every CDR file generated within the same session. All CDR files have a **.cdr**, **.cdr1**, **.cdr2**, or **.cdr2a** extension, depending on the setting for the **CdrType** field in the **cdracct.acc** file described in [Table 168 on page 466](#). The **CdrType** and **.cdr** file extension are summarized in [Table 169 on page 471](#)

Table 169: CdrType Settings in cdracct.acc and Resulting CDR Versions and Filename Extensions

CdrType Setting in cdracct.acc file	CDR Version Generated	CDR Filename Extension
CdrType=BinaryV1	Binary Version 1	.cdr1
CdrType=BinaryV2	Binary Version 2	.cdr2
CdrType=Asn1V2	ASN1 Version 2	.cdr2a

Table 169: CdrType Settings in cdracct.acc and Resulting CDR Versions and Filename Extensions (continued)

CdrType Setting in cdracct.acc file	CDR Version Generated	CDR Filename Extension
no setting specified	Binary Version 1	.cdr

The following example shows a listing of the CDR files in the CDR directory:

```
$ ls
4436830d_00000003.cdr1 44368607_00000003.cdr1 4436865f_00000003.cdr1
4436830d_00000004.cdr2 44368607_00000004.cdr2 4436865f_00000004.cdr2
4436830d_00000001.cdr2a 44368607_00000001.cdr2a 4436865f_00000001.cdr
443c0b94_00000001.cdr 4436830d_00000002.cdr2 44368607_00000002.cdr2
4436865f_00000002.cdr2
```

Using cdrdump to Display CDR File Contents

CDR files are binary or ASN.1 type files. The **cdrdump** tool provided with the optional authentication module displays the contents of a CDR file.

NOTE: Use the **CdrType** field in the [Settings] section of the **cdracct.acc** file to set the type of CDR file (binary version 1, binary version 2, or ASN.1). See [Table 168 on page 466](#) for more information.

To display CDR contents using **cdrdump**:

1. Go to the **radius/CDR** directory and enter the list command to view the CDR filenames. For example:

```
$ ls
```

2. Enter the **cdrdump** command in the following format:

```
$ ../cdrdump [-r] filename
```

where:

filename is the CDR filename as described in [“CDR Files” on page 471](#).

-r sets the display to raw format.

The **-r** raw format command produces unformatted data but does display field names and data types. In raw format, every byte is displayed, even the insignificant bytes (such as those past the end-of-string **NUL** character). Omitting the **-r** field causes the display to appear in formatted mode.

[Table 170 on page 474](#) lists the differences between raw mode and formatted mode.

You cannot use the `-r` field for ASN1 (extension **cdr2a**) files. To display ASN1 files in raw format, see [“Displaying ASN1 CDR Files in Raw Format Using dumpasn1”](#) on page 475.

NOTE: You can send the output of `cdrdump` to a file with the command
\$./cdrdump filename > output_filename.

3. View the **cdrdump** output. Refer to [“CDR Fields”](#) on page 475 for information about each field of information. The following listing shows example output from a CDR file:

```
$ ./cdrdump 45081ce7_00000002.cdr2
DUMP OF CDRv2 FILE "45081ce7_00000002.cdr2":
RecordType.....: 95
ServedImsi.....: 212864080212345
ChargingId.....: 902
GgsnAddress.....: 0.0.0.0
NasAddress.....: 172.25.97.133
NasPortType.....: 0
NasTimeZone(15min,Dst).....: 255,255
AsAddress.....: 172.25.98.242
NodId.....: sbrha-4
AccessPointName/NasId.....: 172.25.97.133
ProtocolType(Org,Val).....: 1,33
MtAddress.....: 10.10.10.10
DataVolumeUplink.....: 10000000
DataVolumeDownlink.....: 2000000
RecordOpeningTime.....: Wed Sep 13 15:04:34 2006 UTC
ChangeTime.....: Wed Sep 13 15:04:54 2006 UTC
Duration.....: 20
CauseForRecordClosing.....: 0
RecordSequenceNumber.....: 1
ChargingType.....: 8
ChargingCharacteristics.....: 0
ConnectionType.....: 255
ServedMsisdn.....: 1212812345
ChargeableUidType.....: 1
ChargeableUidLength.....: 15
DomainIndex.....: 0
ChargeableUid.....: 212864080212345
LocationName.....: Airport
LocationInfo.....: country=US;A1=MA;A3=Boston;ZIP=02128
VisitedOperatorId.....: REALM:foobar.com
OperatorName.....: REALM:sim.com
ExtChargingIdLength.....: 2
ExtChargingId.....: 20
PdpChargingCharacteristics.....: 0
QosTrafficClass.....: 5
PdpAddress.....: 10.10.10.10
```

NOTE: You can use the UNIX **od** command to display the **cdrdump** file contents in purely raw format (all hexadecimal with no field names or data types displayed).

Example: **\$- od -x fname**

cdrdump Output

You can specify that cdrdump output be formatted or raw using the **-r** switch with the **cdrdump** command. (See Step 2 in [“Using cdrdump to Display CDR File Contents” on page 472.](#)) [Table 170 on page 474](#) lists the differences between formatted and raw output.

Table 170: Differences Between Raw and Formatted cdrdump Output

Output Type	Formatted	Raw
Numbers	Decimal	Hexadecimal
IP addresses	IPv4 (dotted quad) if the first 12 bytes are zeroes. IPv6 if the first 12 bytes are non-zero.	IPv6
Timestamps	Version 1: Local time Version 2 and ASN.1: Universal Coordinated Time	Hexadecimal number of seconds since the UNIX epoch followed by ISO format
strings	Terminated at the first null character	Every character is included
BCD strings	Terminated at the first nibble inconsistent with BCD encoding (such as 0xf)	All bytes are displayed in hexadecimal
Version 1 reserved fields	Not displayed	Displayed
Bytes displayed	Only relevant characters are displayed.	Every byte is displayed, including those past the end-of-string null characters.

Displaying ASN1 CDR Files in Raw Format Using dumpasn1

To display ASN.1 CDRs in raw format, use the **dumpasn1** tool located in the **radius** directory. When invoking the **dumpasn1** command, always use the **-z** option to ensure that zero-length fields are displayed properly, as shown in the following example:

```
dumpasn1 -z filename.cdr2a
```

NOTE: CDR files for ASN.1 type can be displayed with formatting (not raw) using the **cdrdump** file as described in [“Using cdrdump to Display CDR File Contents” on page 472](#).

CDR Fields

CDR fields can be of type Version 1, Version 2, or ASN.1. [Table 171 on page 475](#) describes the fields that are contained in the CDR and displayed using **cdrdump**. [Table 172 on page 488](#) describes the field formats for Version 1 and Version 2 CDRs. [Figure 14 on page 490](#) shows the field formats for ASN.1 CDRs.

NOTE: Use the **CdrType** field in the [Settings] section of the **cdracct.acc** file to set the type of CDR file (binary version 1, binary version 2, or ASN.1). See [Table 168 on page 466](#) for more information.

Table 171: CDR Fields

Field Name in CDR Display	Present in Version 1	Present in Version 2 and ASN.1	Field Format	Meaning of Field in cdrdump
RecordType	Yes	Yes	V1/V2: Byte length 1 ASN.1:Integer	Access context of the record. This is always a WLAN context record. The value of this field is always 95.

Table 171: CDR Fields (*continued*)

ServedImsi	Yes	Yes	V1/V2: BCD encoding. Length 8 bytes. ASN.1: NumericString	IMSI of the served party, the user. If IMSI is not present or available, this field contains exactly the home operator's MCC+MNC.
ChargingId	Yes	Yes	Integer length 4	Particular session of the user, together with the GGSN (or AC) address and the Record Sequence Number. Subsequent sessions of a user have a different Charging ID.
GgsnAddress	No	Yes	V2: IP-Address length 16 ASN.1: IPv6 Address	IP address of the GGSN in IPv6 format; for IPv4 addresses, the first 12 bytes are all 0x0. If the user is not connected with GGSN, then use the Access Controller's IP address.
NasAddress	Yes	Yes	V1/V2: IP-Address length 16 ASN.1: IPv6 Address	IP address of the Network Access Server (Access Point in 802.1x and Access Controller in Open System) used for the session. Field is in IPv6 format; for IPv4 addresses, the first 12 bytes are all 0x00.
NasPortType	No	Yes	V2:Integer length 4 ASN.1: Integer	15=Ethernet, 19=802.11. Defines the port type. Value directly from GSMA Vendor-Specific NAS port type attribute received from NAS.

Table 171: CDR Fields (*continued*)

NasTimeZone	No	Yes	V2:Byte length 2 ASN.1: NAS-TimeZone	Time zone and daylight saving usage. First byte indicates Time zone in 15-minute intervals preceded by + for positive or - for negative from GMT. Second byte is daylight saving indication. 1 indicates 1 hour adjustment for Daylight Saving Time. 2 indicates 2 hour adjustment for Daylight Saving Time.
AsAddress	Yes	Yes	V1/V2: IP-Address length 16 ASN.1: IPv6 Address	Address of Application Server (AS) that generates the CDR in IPv6 format; for IPv4 addresses, the first 12 bytes are all 0x00.
NodeId	Yes	Yes	V1/V2: Text length 20 ASN.1: UTF8String	Distinguished Name of the AS that created the record.
AccessPoint Name/NasId	Yes	Yes	V1/V2: Text length 63 ASN.1: UTF8String	NAS-identifier or other available data concerning the location of access zone. This field can be zero.

Table 171: CDR Fields (*continued*)

ProtocolType	Yes	Yes	V1/V2: Word length 2 ASN.1: Protocol-Type	<p>First byte is the <i>PDP type organization</i> (0=ETSI, 1=IETF). Second byte is the <i>PDP type value</i>; for ETSI, valid values are 0 (X.25), 1 (PPP) and 2 (OSP:IHOSS). For IETF valid values are HEX(21) (End User Address information element for IPv4) and HEX(57) (for IPv6).</p> <p>In OWLAN context, the PDP type organization is always 1=IETF.</p>
MtAddress	Yes	Yes	V1/V2: IP-Address length 16 ASN.1: IPv6 Address	<p>IP address of the Mobile Terminal.</p> <p>IP address of the end user's terminal can be sent in the Framed-IP-Address attribute in Accounting-Requests [RFC2866].</p> <p>If the Framed-IP-Address attribute is present in Accounting-Request, AS includes that IP address in the MT Address field of CDR.</p>
DataVolume Uplink	Yes	Yes	Integer length 4	Number of bytes transmitted from the MT since the opening of the CDR.
DataVolume Downlink	Yes	Yes	Integer length 4	Number of bytes transmitted toward the MT since the opening of the CDR
Record Opening Time	Yes	Yes	V1: Local time V2 and ASN.1: UTC	Number of times when the record was opened; that is, when an Accounting-Request with Acct-Status-Type Start (session started) or Interim-Update (partial CDR written) was received. In CDR, the time is AS local time.

Table 171: CDR Fields (*continued*)

ChangeTime	Yes	Yes	V1: Local time V2 andASN.1: UTC	Time when the container was closed, that is, when an Accounting-Request with Acct-Status-Type Stop or Interim-Update was received. V1: Local time. V2 and ASN.1:Timestamp
Duration	Yes	Yes	Integer length 4	Duration of the session in seconds. This value is received from the NAS and is not necessarily the difference between Change Time and Record Opening Time. Use this field as the basis for time-based billing.
CauseFor RecordClosing	Yes	Yes	V1/V2: Byte length 1 ASN.1:Integer	Reason why the CDR is closed. The valid values for this field are: <ul style="list-style-type: none"> • 0 indicates user logged out, lost service, NAS request, callback, or host request. • 1 indicates partial CDR volume threshold exceeded. • 2 indicates partial CDR time threshold exceeded. • 7 indicates user's session was lost, session timeout, port error, NAS error, NAS reboot, port unneeded, port preempted, port suspended, service unavailable, user error, or idle timeout. • 16 indicates Acct-Input/Output-Gigawords counter value has incremented. User has transferred over 232 bytes. • 20 indicates a management action caused session termination.

Table 171: CDR Fields (*continued*)

Record Sequence Number	Yes	Yes	Integer length 4	Running sequence number starting from 1, which is used to link charging records generated for a given end user's session. Value is incremented for each partial record.
ChargingType	Yes	Yes	V1/V2: Byte length 1 ASN.1: Integer	Always 8 (normal postpaid record).
Charging Characteristics	Yes	Yes	V1/V2: Byte length 1 ASN.1: String	Type of CDR. CDRs are created based on the authentication method. The possible value is: <ul style="list-style-type: none"> • 0-EAP-SIM This value is also used in partial CDRs.
Connection Type	No	Yes	Integer	Identifies the type of connection. <ul style="list-style-type: none"> • 0 indicates direct • 7 indicates GGSN
ServedMslsdn	Yes	Yes	V1/V2: BCD encoding. Length 9 bytes. ASN.1: NumericString	Mobile Station ISDN number of the served party.

Table 171: CDR Fields (*continued*)

ChargeableUid Type	No	Yes	V1/V2: Byte length 1 ASN.1: Integer	<p>Field specifies the used type of ChargeableUid (Charging Type identifier).</p> <p>Charging Type identifiers are initially assigned as follows:</p> <ul style="list-style-type: none"> • 00—reserved • 01—IMSI (example: 1231231231244...) • 02—NAI (example: foo@bar.com) • 03—E.164 (a MSISDN - example: +358405627015) • 04—TMPID (as described in 3GPP TS33.234 Temporary Identity Generation example) <p>The choice of the identifier (IMSI, NAI, or MSISDN) is determined by the value set for ChargeableUserIdInResponse in the simauth.aut file. If more than one value is set, the identifier returned by the NAS is used. Usually, the identifier returned by the NAS is the first in the list of multiple identifiers specified for ChargeableUserIdInResponse in the simauth.aut file.</p> <p>For more information, see ChargeableUserIdIn Response in Table 184 on page 508.</p> <p>Also see ChargeableUid Length and ChargeableUid in this table.</p>
ChargeableUid Length	No	Yes	V1/V2: Byte length 1 ASN.1: String	<p>Length of the string in ChargeableUserID.</p> <p>Valid only if the Charging Type Identifier value is 2.</p> <p>Also see ChargeableUidType and ChargeableUid in this table.</p>

Table 171: CDR Fields *(continued)*

DomainIndex	Yes	Yes	V1/V2: Byte length 1 ASN.1: Integer	Location where the domain part starts in the Username and domain field. V2: Valid only if the Charging Type Identifier value is 2 (NAI).
User Name And Domain	Yes	No	Text length 253	Username and realm or domain of the user in NAI format.

Table 171: CDR Fields (*continued*)

ChargeableUid	No	Yes	V1/V2: Integer length 1 ASN.1: UTF8String	<p>Replaces User Name field in Version 1.</p> <p>GSMA-specified chargeable user with ChargeableUid Type and with ChargeableUid Length. The content string interpretation is based on the ChargeableUidType. (It is the value of the IMSI, NAI, or MSISDN.)</p> <p>The choice of the identifier (IMSI, NAI, or MSISDN) is determined by the value set for ChargeableUserIdInResponse in the simauth.aut file. If more than one value is set, the identifier returned by the NAS is used. Usually, the identifier returned by the NAS is the first in the list of multiple identifiers specified for ChargeableUserIdInResponse in the simauth.aut file.</p> <p>If the NAS fails to return a CUID, the value set for DefaultCUIDType in the cdracct.acc file is used.</p> <p>For more information, see ChargeableUserIdIn Response in Table 184 on page 508.</p> <p>Also see ChargeableUidLength and ChargeableUidType in this table.</p>
---------------	----	-----	---	--

Table 171: CDR Fields (continued)

LocationName	Yes	Yes	V1/V2: Text length 32 ASN.1: String	<p>Textual description of the WLAN Hot Spot. For example, "London City Airport." Human readable string without mandated format - printable. Attribute can be used:</p> <ul style="list-style-type: none"> • For string information printed into subscriber's detailed bill • For bilaterally agreed data between operators <p>The contents are copied directly from GSMA Vendor-Specific Location-Name attribute received from NAS. Possible truncation may be done at the end of the string.</p> <p>V1: Format is WISPr. Source is WISPr specific attribute Location-Name.</p> <p>V2 and ASN.1:Format changed from WISPr to IR.61. Source is Vendor Specific Location-Name.</p>
--------------	-----	-----	--	--

Table 171: CDR Fields (continued)

LocationInfo	Yes	Yes	<p>V1/V2: Text length 64</p> <p>ASN.1: String</p>	<p>Location-Information attribute:</p> <ul style="list-style-type: none"> • The ISO 3166 country code is mandatory. • The location identifies the network ("what" is code 3). • Other recommended information includes: <ul style="list-style-type: none"> • A1—State, region, province, or prefecture • A2—County, parish, gun (Korean county), or district • A3—City or township • NAM—Name (residence, business or office occupant) • Additional location information fields may be defined according to bilateral agreements between operators. The contents are copied directly from the GSMA Vendor-Specific Location-Info attribute received from NAS. Strings may be truncated. <p>V1: Format is WISPr. Source is WISPr vendor specific attribute: Location-ID.</p> <p>V2 and ASN.1: Format changed from WISPr to IR.61. Source is WISPr vendor specific attribute: Location-Info.</p>
--------------	-----	-----	---	---

Table 171: CDR Fields (continued)

Visited OperatorID	Yes	Yes	V1/V2: Text length 8 ASN.1: String	<p>Formatted ASCII string that has two parts separated with a colon.</p> <ul style="list-style-type: none"> • GSM:TADIG—Prefix string is “GSM” and the following code is a GSMA assigned TADIG code presented in capital ASCII letters. • REALM:realm—Prefix string is “REALM” and the following code is any valid domain name string that has been acquired from any valid registrar or registry. <p>The contents are copied directly from GSMA Vendor-Specific Visited-Operator-ID attribute received from NAS. Possible truncation may be done at the end of the string.</p> <p>V1: Format WISPr. Set to zero.</p> <p>V2 and ASN.1:Format changed from WISPr to IR.61. Source is Vendor Specific Visitor-Operator-ID.</p>
-----------------------	-----	-----	---	--

Table 171: CDR Fields (*continued*)

OperatorName	No	Yes	<p>V1/V2: Text length 128</p> <p>ASN.1: String</p>	<p>Formatted ASCII string that has two parts separated with a colon:</p> <ul style="list-style-type: none"> • GSM:TADIG—Prefix string is “GSM” and the following code is a GSMA assigned TADIG code presented in capital ASCII letters. • REALM:realm—Prefix string is “REALM” and the following code is any valid domain name string that has been acquired from any valid registrar or registry. <p>The contents are copied directly from GSMA Vendor-Specific Visited-Operator-ID attribute received from NAS. Truncation can be done at the end of the string. The contents are copied directly from GSMA Vendor-Specific Operator- Name attribute received from NAS. Possible truncation can be done at the end of the string.</p> <p>If both Visited-Operator-ID and Operator-ID are present in the Access-Response, the Visited-Operator-ID is used.</p>
ExtChargingId Length	No	Yes	<p>V1/V2: Text length 1</p> <p>ASN.1: Integer</p>	<p>Length of ExternalChargingID attribute. Contains length of ExChargingID. For example, if the value of the ExternalChargingID attribute is 20, the length is 2.</p>
ExtChargingId	No	Yes	<p>V1/V2: Text length 1</p> <p>ASN.1: String</p>	<p>Value of the RADIUS attribute, Acct-Session-ID.</p>

Table 171: CDR Fields (*continued*)

PdpCharging Characteristics	No	Yes	V1/V2: Byte length 1 ASN.1: Integer	Charging type applied to PDP context.
QosTraffic Class	No	Yes	V1/V2: Byte length 1 ASN.1: Integer	Quality of Service. Possible values = 0, 1, 2, 3. The default is 255 if no value is present.
PdpAddress			V1/V2: IP Address length 16 ASN.1:IPV-6 Address	User Equipment address on the TTG toward the GGSN. Same as “MT address” specified in this table.

CDR Field Formats for Binary and ASN.1 CDR Files

[Table 172 on page 488](#) describes the field formats for binary Version 1 and Version 2 CDRs and [Figure 14 on page 490](#) shows the field formats of the ASN.1 type formats.

Field Formats for Binary Version 1 and Binary Version 2 CDR Files

Table 172: Format Types for Binary Version 1 and Binary Version 2 CDR Fields

Field Format	Description
Byte	Unsigned 8-bit integer in network byte order.
Integer	Unsigned 32-bit integer in network byte order.
Text	ASCII characters.

Table 172: Format Types for Binary Version 1 and Binary Version 2 CDR Fields *(continued)*

Field Format	Description
BCD	<p>Binary-Coded-Decimal (BCD) with 0xF as padding.</p> <p>BCD format has the following characteristics:</p> <ul style="list-style-type: none"> • packed • swapped nibbles • hexadecimal 0xF padded
IP-Address	<p>128-bit IPv6 Address network byte order, for Ipv4 addresses, the first 12 bytes are all 0x00.</p>
Timestamp	<p>Local time in binary Version 1</p> <p>UTC time in binary Version 2</p>

4

PART

SIM Authentication Module

[Common Configurations](#) | **492**

[SIM/AKA Authentication](#) | **506**

Common Configurations

IN THIS CHAPTER

- [ss7db.gen File | 492](#)
- [gsmmap.gen File | 495](#)
- [Signalware MML Commands | 504](#)

This chapter explains the common configurations of the SIM authentication files.

ss7db.gen File

When Steel-Belted Radius Carrier is running in standalone mode and is also running the SIM authentication module, it uses an LDAP directory to extend session information, and store SIM and CDR data records. You need to configure the **ss7db.gen** file to set up these storage parameters.



WARNING: Earlier SBRC 7.2.x releases running in standalone mode and using the SIM module used a different method to store the CDR records. SBRC releases 7.2.4 and later use an LDAP directory to store these records. This directory is configured with the **ss7db.gen** file. CDR records from SBRC versions earlier than 7.2.4 will not be available after upgrading to 8.3.0.

NOTE: While configuring the **ss7db.gen** file for CDR accounting in the SBR cluster version, you need to set the **dbclusterndb.so** parameter under the [GenericPlugins] section of **radius.ini** to **dbclusterndb.gen** for getting NDB cookies in ss7db during SBR startup.

[Bootstrap] Section

The [Bootstrap] section ([Table 173 on page 493](#)) of the **ss7db.gen** contains the following settings:

Table 173: ss7db.gen [Bootstrap] Syntax

Parameter	Function
LibraryName	Specifies the name of the executable binary. Default value is ss7db.
Enable	Specifies whether to enable or disable storage of CDR records in the LDAP directory. <ul style="list-style-type: none"> Set to 1 to enable storage in the external LDAP database. Set to 0 to disable storage in the external LDAP database. Default value is 0.

[Settings] Section

If you intend to use your own external LDAP server, you can configure the following items in the [Settings] section ([Table 174 on page 493](#)) of **ss7db.gen**.

Table 174: ss7db.gen [Settings] Syntax

Parameter	Function
ConfigLog	Specifies where LDAP configuration information is logged. Options are: <ul style="list-style-type: none"> None—Do not log LDAP configuration information. Log—Record LDAP configuration information in the Steel-Belted Radius Carrier log file. Console—Display configuration information about the console only. ConsoleAndLog (default)—Record LDAP configuration information in the Steel-Belted Radius Carrier log file and display configuration information about the Steel-Belted Radius Carrier console.
LDAPServerIPAddr	The IP address of the LDAP server. The default is 127.0.0.1. This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.

Table 174: ss7db.gen [Settings] Syntax (*continued*)

Parameter	Function
LDAPServerPort	<p>The port number for the LDAP server.</p> <p>The default value is 389.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
LDAPUserName	<p>The string for the name of the LDAP server user account.</p> <p>The default is cn=Manager, o=sbrsms, c=US.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
LDAPPassword	<p>The LDAP server password. This password string must correspond to the credentials of the user account name in the previous field.</p> <p>The default is password.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
LDAPBaseDN	<p>The specification for the directory tree where LDAP SMS files, including provisioned accounts, are stored.</p> <p>The default is o=sbrsms, c=US.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
LDAPVersion	<p>The version of LDAP supported by ss7db. The version specified here must match the version number of the LDAP server. You can specify 2 or 3.</p> <p>The default is 2.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 174: ss7db.gen [Settings] Syntax (*continued*)

Parameter	Function
LDAPServerTimeOutSec	<p>If a connection to the LDAP server cannot be made within the specified number of seconds, the transaction with the server is cancelled. A value of 0 means no transaction timeout.</p> <p>The default is 0.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
LDAPMaxNumConnections	<p>The maximum number of simultaneous connections with the LDAP server.</p> <p>The default is 300.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
StaleAccountCleanerSweepFrequencyMin	<p>Specifies how often accounts that have exceeded their grace period are purged from the LDAP database.</p> <p>The default is 30 minutes.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
ExpiredAccountGracePeriodMin	<p>Grace period, in minutes, within which an expired account may be re-billed, if the subscriber authenticates with the previously provided password during this period.</p> <p>The default is 5.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

gsmmap.gen File

This section describes the **gsmmap.gen** file used by the SIM authentication module to define settings for authenticating Access-Request messages. The following topics are included in this chapter:

The **gsmmap.gen** file enables you to configure authentication settings by realm. This file consists of several sections that you need to configure, including:

- [Bootstrap] section
- [Settings] section
- [Realms] section
- Each realm section
- Target module sections

This section describes each of these configuration sections.

[Bootstrap] Section

The [Bootstrap] section ([Table 175 on page 496](#)) of the **gsmmap.gen** file enables the **gsmmap.gen** file to function.

Table 175: gsmmap.gen [Bootstrap] Fields

Field	Description
LibraryName	Specifies the name of the executable binary. Default value is gsmmap.
Enable	Set to 1 to enable this file. Set to 0 to disable this file. Default value is 0.

[Settings] Section

The [Settings] section ([Table 176 on page 497](#)) controls how log information is handled.

Table 176: gsmmap.gen [Settings] Fields

Field	Description
ConfigLog	<p>Method for capturing log information.</p> <ul style="list-style-type: none"> • None= Configuration information is not captured. • ConsoleAndLog= Log information is sent to both the console and the log. • Console= Log information is sent to the console only. • Log= Log information is sent to the log file only. <p>Default is ConsoleAndLog.</p>

[Realms] Section

The [Realms] section of the **gsmmap.gen** file contains a list of realms for which you specify authentication instructions. When an Access-Request is received, Steel-Belted Radius Carrier handles the request in different ways, depending on the settings in the [Realms] section. For example, requests from the ABC.com realm might require the IMSI retrieved from the LDAP database for authentication, requests from the XYZ.com realm might require the AKA from the MAP Gateway for authentication.

You can specify realms in several ways:

- By name—You can specify realms directly by listing names of authorized realms. Example: **abc.com**.
- By alias—You can create an alias for a realm by specifying the realm alias and realm name. Example: **realm1=abc.com**
- By wildcard alias—You can create an alias that includes a wildcard to permit authentication for multiple realms. Example: **realm2=*abc.com** or **realm=abc.***
- By unmatched realm—You can create an alias that applies to all realms that do not match any specified realm. Example: **CatchAllRealm=***
- By no realm—You can capture all authentication requests that do not contain a realm with the **NoRealm=** command.

Configuring Each Realm Section

For each realm or alias that you create in the [Realms] section, you must create a separate section identified by the specified realm name or alias in the **gsmmap.gen** file. Within each realm setting, you identify a *target module* for each type of information that might be required to authenticate a subscriber. The target module defines where to obtain the specified information for each type of authenticator.

For example, if ABC.com is one of the realms, you must create a target module for any of the EAP-SIM, EAP-AKA, IMSI, MSISDN, and Authorization authentication types that are used to authenticate subscribers from ABC.com.

Use the Default= setting to identify a target module to be called if any of the other settings are absent.

NOTE: The Setting Name can be set to **None** if you want to disable the setting. For example, Authorization=None.

Example

In the following example, these configuration choices are specified:

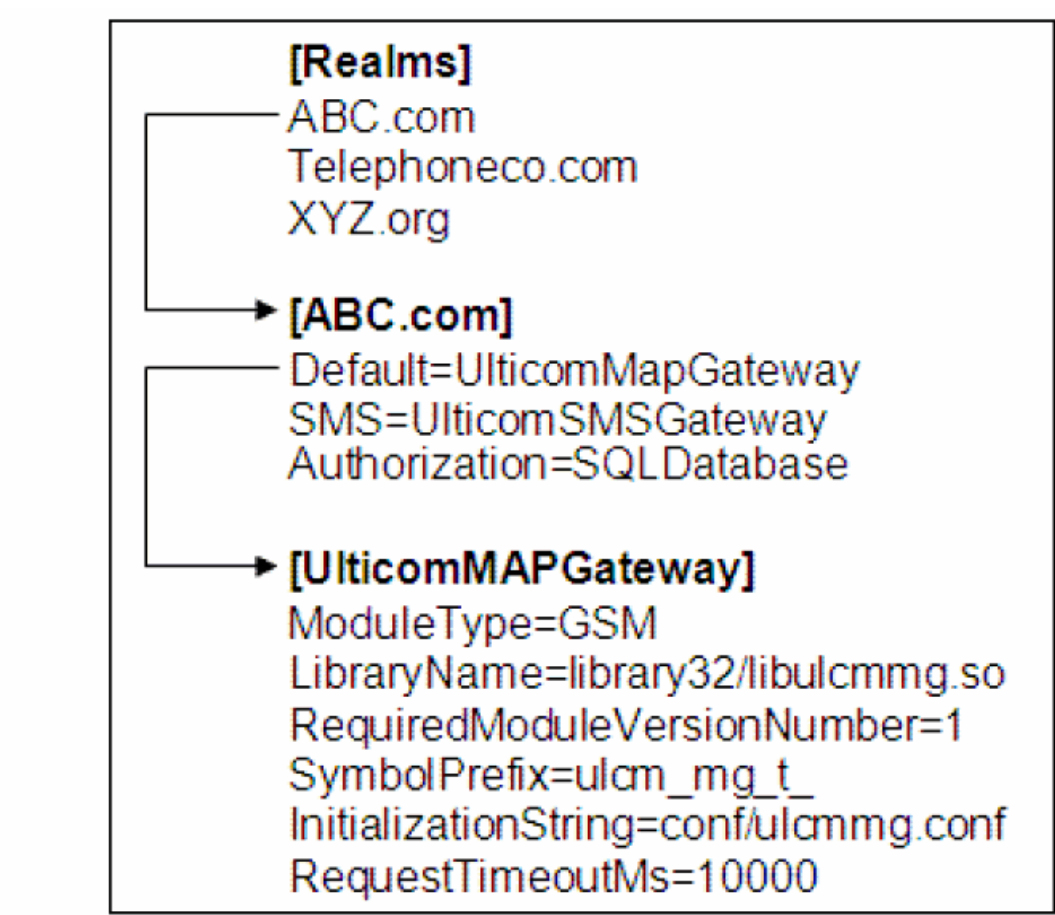
- Access-Requests requiring an authorization string are handled according to the settings in the SQLDatabase target module section of **gsmmap.gen**.

All other Access-Requests are handled according to the UlticomMapGateway target module section of **gsmmap.gen**.

Relationship Between Sections

[Figure 15 on page 499](#) illustrates the relationship between the [Realms] section, the specific named realm section, and the target module section in the **gsmmap.gen** file.

Figure 15: Relationship Between Sections in gsmmap.gen File



Network Equipment and Data Needed for Processing Access-Requests

Table 177 on page 499 identifies the network equipment needed for authentication based on the action needed to process the Access-Request.

Table 177: Network Equipment and Related Settings, Actions, and Identifiers

Setting Name	Action Needed to Process Access-Request	Identifier of the Mobile Station	Network Equipment
SIM	Obtain SIM triplets*	IMSI	HLR (supporting MAP application context version 2 or 3)
AKA	Obtain AKA quintets	IMSI	HLR (supporting MAP application context version 3)

Table 177: Network Equipment and Related Settings, Actions, and Identifiers (*continued*)

Setting Name	Action Needed to Process Access-Request	Identifier of the Mobile Station	Network Equipment
IMSI	Obtain IMSI (given the MSISDN)	MSISDN	HLR
MSISDN	Obtain MSISDN (given the IMSI)	IMSI	HLR
Authorization	Obtain Authorization string	IMSI or MSISDN	HLR or SQL or LDAP database

* If quintets are received but triplets are needed, the authentication module converts the quintets to triplets according to specification 3G TS 33.102 available at <http://www.3gpp.org>.

NOTE: You can set the Setting Name to None if you want to disable the setting. For example, **SIM=None**.

Example: Authorization String

If an authorization string is required to process an Access-Request, the following might be true:

- Authorization string is in the database
- IMSI is received in the Access-Request
- Database is keyed off the MSISDN

In this case, the Mobile Switching Center (MSC) is used to obtain the MSISDN based on the IMSI. Then the MSISDN is used to retrieve the Authorization string from the database or HLR.

Disabling Authorization from EAP-SIM

You can disable authorization completely from EAP-SIM (not fetch subscriber profile information from the HLR and not perform a SQL/LDAP query).

To disable authorization from EAP-SIM:

1. Set **Authorization=None** in the realm section of the **gsmap.gen** file.

2. Remove all authorization options (BS, TS, and ODB) from the **authGateway.conf** file for the target HLR, disable the connection between authGateway and GWrelay applications in the **GWrelay.conf** file, and disable the connection between SBR Carrier and the GWrelay application in the **ulcmmg.conf** file. For complete details on the **authGateway.conf**, **GWrelay.conf**, and **ulcmmg.conf** files, see the *SBR Carrier Installation Guide*.

Target Module Section

For each target module that you list for a realm, you must create a configuration section that identifies settings to be used for that module. The settings that you must specify depend on the type of module being called. The target modules are described in [Table 178 on page 501](#).

Table 178: Types of Target Modules

Target Module	Type	Source of Subscriber Information	Default Target Module Name
MAP Gateway	GSM	HLR	UlticomMapGateway
SQL Database	Database	SQL database	SQLDatabase
LDAP Database	Database	LDAP database	LDAPDatabase

The fields to be included in the target module section differ depending on the specific target module. For example, the MAP Gateway target module section in the **gsmmap.gen** file requires a different set of fields than the LDAP database target module. [Table 179 on page 501](#) through [Table 182 on page 504](#) list the fields required for each target module.

Target Module Fields (General Case)

Table 179: gsmmap.gen [Module] Fields (General Case)

Field	Description
ModuleType	Specifies the type of module being called. Options are: <ul style="list-style-type: none"> • Database • GSM
LibraryName	Specifies the name of the executable binary.
Required Module VersionNumber	Version number of the specified module. Default value is 1.

Table 179: gsmmap.gen [Module] Fields (General Case) (*continued*)

Field	Description
SymbolPrefix	Specifies the prefix for the symbols loaded from the library. <ul style="list-style-type: none"> For the MAP Gateway, enter ulcm_mg_t_.
InitializationString	Specifies the name of the configuration file for the library.
RequestTimeoutMs	Specifies the number of milliseconds Steel-Belted Radius Carrier waits for a request from the library to complete. Enter a value that reflects how long the SS7 network takes to complete a request. For example, a MAP Gateway communicating with an HLR requires a relatively short timeout value; for example, 10000 (10 seconds). This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.

MAP Gateway Target Module Fields

Table 180: gsmmap.gen MAP Gateway Module Fields

Field	Configure to This Value
ModuleType	GSM
LibraryName	library32/libulcmmg.so
Required Module Version Number	1
SymbolPrefix	ulcm_mg_t_
InitializationString	conf/ulcmmg.conf See the ulcmmg.conf file in the <i>SBR Carrier Installation Guide</i> .
RequestTimeoutMs	Number of milliseconds Steel-Belted Radius Carrier waits for a request from the library to complete. Enter a value that reflects how long the network takes to complete a request. For example, a MAP Gateway communicating with an HLR requires a relatively short timeout value; for example, 10,000 (10 seconds).

Example of MAP Gateway Target Module Fields

```
[UlticomMAPGateway]
ModuleType=GSM
LibraryName=library32/libulcmmg.so
RequiredModuleVersionNumber=1
SymbolPrefix=ulcm_mg_t_
InitializationString=conf/ulcmmg.conf
RequestTimeoutMs=10000
```

SQL Database Target Module Fields

Table 181: gsmmap.gen SQL Database Fields

gsmmap.gen [Database] Field	Configure to This Value
ModuleType	<p>Database</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
DatabaseAccessor MethodName	<p>Name by which the SQL data accessor registers itself with Steel-Belted Radius Carrier. This value must match the value entered in the MethodName setting in the sqlaccessor.gen or sqlaccessor_jdbc.gen file, see “SQL Accessors” on page 419.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
KeyForAuthorization	<p>Specifies whether the subscriber is identified by IMSI or MSISDN (key field). Valid values are:</p> <ul style="list-style-type: none"> • IMSI • MSISDN <p>For more information about setting database keys, see “Detailed Use Cases” on page 615.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Example of SQL Database Target Module

```
[SQLDatabase]
ModuleType=Database
DatabaseAccessorMethodName=SQL Accessor
KeyForAuthorization=MSISDN
```

LDAP Database Target Module Fields

Table 182: gsmmap.gen LDAP Database Fields

Field	Configure to This Value
ModuleType	Database
DatabaseAccessor MethodName	Name by which the SQL data accessor registers itself with Steel-Belted Radius Carrier. This value must match the value entered in the MethodName setting in the ldapaccessor.gen file, see “LDAP Accessor Files” on page 460).
KeyForAuthorization	Specifies whether the subscriber is identified by IMSI or MSISDN. Valid values are: <ul style="list-style-type: none"> • IMSI • MSISDN For more information about setting database keys, see “Detailed Use Cases” on page 615 .

Example of LDAP Database Target Module

```
[LDAPDatabase]
ModuleType=Database
DatabaseAccessorMethodName=LDAP Accessor
KeyForAuthorization=IMSI
```

Signalware MML Commands

After Signalware is installed, you can configure and provision Signalware using commands sent to the Signalware system. For details about installing and running Signalware, see the *SBR Carrier Installation Guide*.

These commands are in Man-Machine Language (MML). You can input MML commands individually using the SWMML program, or save them in a file. The procedures in this chapter assume that you save the MML commands to **.mml** text files and execute them as described in the section “Loading the MML Configuration Settings” in the *SBR Carrier Installation Guide*.

The basic activities that require MML commands are:

- Setting up link sets, links, and routes
- Configuring the authGateway location and startup information.

- Loading the MML configuration settings.

To view a list of all MML commands and definitions, enter: **man MML_Intro**

To view specific information about any MML command, enter: **man *cmdname***.

For example:

\$ man CRTE-LSET

See the *SBR Carrier Installation Guide* for more information.

SIM/AKA Authentication

IN THIS CHAPTER

- Overview | 506
- Configuring the `simauth.aut` File | 506
- Authentication Gateway | 516

This chapter explains the **simauth.aut** and authentication gateway configuration files for SIM/AKA authentication.

Overview

This chapter describes configuration tasks for using EAP-SIM or EAP-AKA credentials to authenticate mobile subscribers for wireless hotspot Internet access.

Configuring the `simauth.aut` File

The SIM authentication module handles EAP-SIM authentication for clients using SIM cards and EAP-AKA authentication for clients using USIM cards. The settings for EAP-SIM and EAP-AKA authentication are stored in the **simauth.aut** file in the Steel-Belted Radius Carrier installation directory.

NOTE: Authenticating subscribers requires communication with the HLR. To establish connection with the HLR, follow the directions for installing and configuring the Signalware SIGTRAN protocol stack and configuring the `authGateway` and `GWrelay` applications described in the *SBR Carrier Installation Guide*.

simauth.aut [Bootstrap] Section

The [Bootstrap] section ([Table 183 on page 507](#)) of the **simauth.aut** file specifies information that Steel-Belted Radius Carrier uses to load and start the SIMAuth module.

Table 183: simauth.aut [Bootstrap] Fields

Field	Description
LibraryName	Specifies the name of the binary that implements the SIM authentication method. Default value is simauth.so .
Enable	If set to 1, the simauth authentication method is enabled. If set to 0, the simauth authentication method is disabled. Default value is 0.
InitializationString	Specifies the name of a server cookie used by the simauth authentication method. The value must be set to simauth.

simauth.aut [Settings] Section

The [Settings] section ([Table 184 on page 508](#)) of the **simauth.aut** file contains these settings:

NOTE: For an overview of EAP-SIM and EAP-AKA pseudonyms and reauthentication identities, see the information about the SIM Authentication Module in the *SBR Carrier Administration and Configuration Guide*.

SBRC always tries protocols in a fixed order (EAP-SIM, EAP-AKA), skipping any protocols that are not enabled. It is possible for the NAS client to select whether a particular protocol is preferred. If the NAS client prefers to use a particular protocol, then SBRC accepts the protocol in its fixed order list. If the preferred protocol does not work for some reason (for example, the protocol is disabled), then SBRC continues with the next protocol in its fixed order list. This is a known limitation in the NAS client that it prefers a particular protocol but sends a NAK when SBRC accepts it. For example, it is not possible for SBRC to try EAP-SIM after a NAS client prefers EAP-AKA, but it is possible for SBRC to try EAP-AKA after the NAS client uses EAP-SIM.

Table 184: simauth.aut [Settings] Fields

Field	Description
ConfigLog	<p>Specifies where simauth configuration information is logged. Options are:</p> <ul style="list-style-type: none"> • None= Configuration information is not captured. • ConsoleAndLog= Log information is sent to both the console and the log. • Console= Log information is sent to the console only. • Log= Log information is sent to the log file only. <p>Default is ConsoleAndLog.</p>
EnableEAPSIM	<p>If set to 1, EAP-SIM authentication with GSM SIM cards is enabled.</p> <p>If set to 0, EAP-SIM authentication is disabled.</p> <p>Default value is 1 (enabled).</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
EnableEAPAKA	<p>If set to 1, EAP-AKA authentication with 3G USIM cards is enabled.</p> <p>If set to 0, EAP-AKA authentication is disabled.</p> <p>Default value is 1 (enabled).</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
EnforceEAPHint	<p>If set to 1, and the leading digit of the IMSI is 0 or 1, the server determines the type of authentication (0 for EAP-AKA, 1 for EAP-SIM) and will be enforced.</p> <p>If set to 0, the type of authentication is determined through negotiation with the client.</p> <p>Default value is 0.</p>

Table 184: simauth.aut [Settings] Fields (continued)

Field	Description
NumberOfTriplets	<p>The number of triplets required for each authentication. The allowable values are 2 or 3. The higher the number of triplets, the more keying material is available and, consequently, the more the authentication is resistant to attacks.</p> <p>The default value is 3.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
ReauthenticationRealm	<p>The realm returned with a Reauthentication Identity that indicates where the return responses to reauthentication requests are directed.</p> <p>The default uses the realm from the Permanent Identity (if any exists) of the client.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
ReauthenticationCount Limit	<p>The number of allowed reauthentications before requesting fresh triplets and performing a complete authentication. The range is 0–65535.</p> <p>The default is 65535.</p> <p>If you enter 0, Reauthentication Identities are not generated.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
ReauthenticationLifetimeSec	<p>The duration (in seconds) of an identity that is generated by Steel-Belted Radius Carrier for the purpose of reauthentication.</p> <p>The default is 3600 (1 hour). The client must reauthenticate within this time or use a Pseudonym or Permanent Identity.</p> <p>Set to 0 to prevent reauthentication identities from being generated.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 184: simauth.aut [Settings] Fields (continued)

Field	Description
ProfilesUser	<p>Specifies whether the authorization policy specifiers listed in the [ProfileMap] section of simauth.aut represent Steel-Belted Radius Carrier native users or profiles.</p> <ul style="list-style-type: none"> • If set to 0 (default) the profiles listed in the ProfileMap section represent profiles configured in SBR Carrier. • If set to 1 the profiles in the ProfileMap section represent users. <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
PseudonymSecret	<p>A secret used to encrypt Pseudonyms.</p> <p>You can use any text string up to 32 characters. There is no default. If you do not specify a secret, pseudonyms are not generated.</p> <p>When you change this value, all pseudonyms assigned to currently authenticated clients are invalidated, requiring reauthentication.</p> <p>NOTE: If running the SIM authentication option in an SBR Carrier cluster, all pseudonym passwords should be the same throughout the cluster.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
PseudonymLifetimeDays	<p>The lifetime of a Pseudonym in days.</p> <p>The default is 1 (day).</p> <p>The actual lifetime varies from the specified time, to twice the specified time.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 184: simauth.aut [Settings] Fields (continued)

Field	Description
UseEAPResponselidentity	<p>Set to 1 when the EAP-Response/Identity message is not altered by a proxy RADIUS server. If you set this to 1 and such changes do take place, then authentication fails.</p> <p>Set to 0 when the client EAP Identity is modified by a proxy RADIUS server.</p> <p>The default is 0.</p>
EnableFailover	<p>Specifies whether to enable or disable RADIUS failover. The RADIUS failover occurs when the HLR is inaccessible. The authentication requests are silently discarded until the HLR becomes available.</p> <ul style="list-style-type: none"> • If set to 1, enables RADIUS failover. • If set to 0, disables RADIUS failover. <p>Default value is 0.</p> <p>This parameter is reloaded every time when SBR Carrier receives a SIGHUP (1) signal.</p>
FailoverTimeoutSec	<p>If failover is enabled, the HLR is presumed to be accessible after the specified number of seconds, and the next request is not silently discarded.</p> <p>Default is 60 seconds.</p> <p>Setting this value to 0 causes EAP-SIM/EAP-AKA authentication requests to be discarded if the HLR is down.</p>

Table 184: simauth.aut [Settings] Fields (*continued*)

Field	Description
ChargeableUserIdInResponse	<p>Specifies which identifier is used as the CUID and returned in the Chargeable-User-Id (CUID) attribute in the RADIUS Access-Accept message.</p> <p>This field can be set to:</p> <ul style="list-style-type: none"> • None (default) • IMSI—The attribute format is 01:<i>imsi</i>, where <i>imsi</i> is the subscriber's IMSI. • NAI—The attribute format is 02:<i>nai</i>, where <i>nai</i> is the subscriber's NAI. • MSISDN—The attribute format is 03:<i>msisdn</i>, where <i>msisdn</i> is the subscriber's MSISDN. <p>You can specify more than one CUID identifier. The identifiers must be comma separated. The identifier returned by the NAS is used as the CUID. This identifier is usually the first identifier in the list.</p> <p>In the following example, the CUID in the CDR is the IMSI.</p> <p>ChargeableUserIdInResponse=IMSI,NAI</p> <p>The IMSI and NAI values returned in the Access-Accept are derived from the User-Name attribute in the initial Access-Request. The MSISDN value returned in the Access-Accept is derived from the target module specified in the gsmmap.gen file. See “Configuring Each Realm Section” on page 497 and “Target Module Section” on page 501.</p> <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

Table 184: simauth.aut [Settings] Fields (*continued*)

Field	Description
ChargeableUserIdAttribute	<p>Specifies the attribute that contains the CUID in the Access-Accept.</p> <p>This field can be set to either of these settings:</p> <ul style="list-style-type: none"> ● Chargeable-User-Identity—The CUID is returned in the Chargeable-User-Identity attribute. This setting complies with RFC 4372 available at http://www.ietf.org. ● TeliaSonera-Chargeable-UserId—The CUID is returned in the TeliaSonera-Chargeable-UserId attribute. This setting complies with GSM Association document IR6.1 available at http://www.gsmworld.com. <p>NOTE: The following call detail record (CDR) fields carry CUID information. See “CDR Fields” on page 475 for a list of all CDR types.</p> <ul style="list-style-type: none"> ● Charging ID ● Domain Index ● Charging Identifier <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>
SendCUIDOnlyIfReceived	<p>Used only if ChargeableUserIdAttribute is set to Chargeable-User-Identity.</p> <p>This field may be set to:</p> <ul style="list-style-type: none"> ● 0—The CUID attribute is attached to the Access-Accept regardless of whether the CUID attribute was received in the Access-Request. ● 1—The CUID attribute is attached to the Access-Accept only if the CUID was received in the Access-Request. <p>This parameter is reloaded every time that SBRC receives a SIGHUP (1) signal.</p>

simauth.aut [ProfileMap] Section

Your HLR includes a database of subscribers. The database maps subscribers to one or more classes of service to which they are subscribed. You can configure the MAP Gateway to return strings that indicate a subscriber's service authorization. The service authorization strings returned by the MAP Gateway

correspond to selected TS, BS, or ODB settings in the subscriber's profile on the HLR. For information about defining service authorization strings, see the Authorization Options section of the **authGateway.conf** file in the *SBR Carrier Installation Guide*.

Alternatively, you can configure Steel-Belted Radius Carrier to request service authorization strings from an external SQL or LDAP database instead of from an HLR. For information about requesting authorization information from an SQL database, see [“SQL Database Data Retrieval Methods” on page 423](#). For information about requesting authorization information from an LDAP directory, see the [“\[Response\] Section” on page 462](#) of the **ldapaccessor.gen** file.

You can create two types of Steel-Belted Radius Carrier entities that are used for specifying authorization policies:

- Profiles
- Native user

You can use Steel-Belted Radius Carrier profiles or native users to define classes of subscribers who are authorized for service. Refer to the *SBR Carrier Administration and Configuration Guide* for information about creating profiles and native users.

NOTE: If you configure native users for use with EAP-SIM or EAP-AKA authentication, then you must modify **ProfileIsUser** in **simauth.aut**.

You can use the [ProfileMap] section of the **simauth.aut** file to assign one or more service authorization strings to these Steel-Belted Radius Carrier profiles (or native users). By doing so, each time that a subscriber requests an account, the service that is specified by the HLR is checked against the strings that you assign to a profile or native user in the [ProfileMap] section of the **simauth.aut**:

- If the set of service authorization strings do not match those of any profile or native user who you list, the provisioning request is denied.
- The profile map entries are checked in order. When the set of service strings match those of a profile or native user who you list, the subscriber is assigned that Steel-Belted Radius Carrier profile or native user.
- Although the authorization string values must match those in **authGateway.conf**, you can include authorization strings that are valid for multiple HLRs in a given line corresponding to a profile or native user in the [ProfileMap] section of **simauth.aut**.

You can use each line in the [ProfileMap] section of **simauth.aut** to provide a set of HLR authorization strings (from **authGateway.conf**) that specify an authorization policy defined by a Steel-Belted Radius Carrier profile or native user.

The format for each line is:

```
ProfileName1=auth1:auth2:auth3
```

where **ProfileName1** is the name of a Steel-Belted Radius Carrier profile or native user, and **auth1**, **auth2**, and **auth3** are the HLR authorization strings configured in the MAP Gateway **authGateway.conf** file. The sequence and capitalization of the authorization strings are ignored. You can specify up to 128 authorization strings.

To use a particular profile or native user to define an authorization policy, make sure all the listed authorization strings exactly match the authorization strings returned from the MAP Gateway. You can also use a wildcard * to match any otherwise unmatched strings. For example, you can have the following:

```
ProfileName1=auth1:auth2:*
```

This matches any set of authorization strings as long as they include **auth1** and **auth2**.

You can specify combinations of authorization strings for which authorization is denied by entering one or more DENY lines. A DENY line is of the form:

```
<DENY>=auth1:auth2:auth3
```

where the profile name is not specified and where **auth1**, **auth2**, and **auth3** are the HLR authorization strings configured in the MAP Gateway **authGateway.conf** file. The same matching and wildcard rules apply as with PROFILE lines. For example, the following statement denies authorization if the authorization strings returned are **auth1**, **auth2**, and **auth3**.

```
<DENY>=auth1:auth2:auth3
```

The following statement denies authorization if the authorization strings returned include both **auth1** and **auth2** and any other strings.

```
<DENY>=auth1:auth2:*
```

Similarly, the following statements deny authorization if the authorization strings returned include **auth1** or **auth2**.

```
<DENY>=auth1:*  
<DENY>=auth2:*
```

Finally, the following statement denies authorization if no authorization strings are returned:

```
<DENY>=
```

We recommend you place DENY statements at the top of the list because a subscriber is assigned the policy associated with the first match in the list of profiles (or native users) that you include in the profile map. Place **PROFILE** statements with specific criteria in the middle of the list and PROFILE statements with wildcards at the bottom of the list.

NOTE: If the set of service authorization strings do not match those of any profile or native user who you list, the provisioning request is denied.

All settings are reloaded every time when SBRC receives a SIGHUP (1) signal.

Authentication Gateway

This section explains the configurations of **ulcmmg.conf**, **GWrelay.conf**, and **authGateway.conf** files.

Configuring the ulcmmg.conf File

The **ulcmmg.conf** file establishes the connection between the GWrelay application and SBR Carrier.

The **ulcmmg.conf** file contains lines of the following form (the angle brackets <> indicate required parameters and should not be included in the actual configuration):

```
LOCAL_HOST <host-name>: <port>
REMOTE_HOST <host-name>:<port><ip-address>
NUMBER_OF_RETRIES <retry_value>
DELAY_BETWEEN_RETRIES <delay_time>
DEBUG
LOG_FILE <file-name>
```

The LOCAL_HOST line is required. The required host-name parameter specifies the name of the local host, for example, the machine on which the SBRC is executing. The required port parameter specifies the outbound socket port number (2001 by default) that is used by the SBRC to transmit messages to the GWrelay process.

The REMOTE_HOST line is required. The required host-name parameter specifies the name of the host on which the GWrelay process is executing. The required port parameter specifies the inbound socket port number (2000 by default) that is used by the GWrelay process to listen for messages from SBRC. The required **ip-address** parameter specifies the IP address corresponding to the host-name parameter.

The `NUMBER_OF_RETRIES` line is optional. The **retry_value** parameter specifies the number of retry attempts to be performed by SBR Carrier when attempting to connect to the authGateway instance through the GWrelay application. Default value is 3.

The `DELAY_BETWEEN_RETRIES` line is optional. The **delay_time** parameter specifies the number of seconds SBR Carrier must wait between retry attempts. Default value is 3 seconds.

The `DEBUG` line is optional. When present, it enables debug messages to be emitted on the Signalware console device. Enabling debug messages degrades the performance and should be avoided in production environments.

The `LOG_FILE` line is optional. When present it enables diagnostic messages, including debug messages if enabled, to be written to the log file that is specified by the required **file-name** parameter. The file-name parameter should specify an absolute path-name for the log file.

For more information, see the *SBR Carrier Installation Guide*.

Configuring the GWrelay.conf File

The **GWrelay.conf** file is used to define the source and destination ports through which an SCTP connection is established between the GWrelay application and authGateway instances.

The **GWrelay.conf** file contains lines of the following form (the angle brackets <> indicate required parameters and should not be included in the actual configuration):

```
LOCAL_HOST <host-name>: <portA>
REMOTE_HOST <host-name>:<porta><ip-address>

LOCAL_HOST <host-name>: <portB>
REMOTE_HOST <host-name>:<portb><ip-address>

...

LOCAL_HOST <host-name>: <portZ>
REMOTE_HOST <host-name>:<portz><ip-address>

RELAY_SERVER <host-name>: <port>
NumberOfRetries <retry_value>
DelayBetweenRetries <delay_time>
LogLevel <level>
LogPath <file-path>
LogSize <file-size>
```

The `LOCAL_HOST` line is required. The required **host-name** parameter specifies the name of the local host on which the GWrelay process is executing. The required **port** parameter specifies the outbound socket port number that is used by the GWrelay application to listen for responses from the authGateway instances.

The REMOTE_HOST line is required. The required **host-name** parameter specifies the name of the host on which the authGateway process is executing. The **host-name** parameter value must match the value set in the -host option in the MML CREATE-PROCESS statement that is used to start the authGateway process. The **host-name** parameter value should also match the value in the LOCAL_HOST line, that is, the authGateway process is intended to execute on the same host as SBR Carrier. The required **port** parameter specifies the inbound socket port number that is used by the authGateway instances to listen for messages from the GWrelay application. The **port** parameter value must match the value set in the -port option in the MML CREATE-PROCESS statement. The required **ip-address** parameter specifies the IP address corresponding to the **host-name** parameter.

RELAY_SERVER line is required. The required **host-name** parameter specifies the name of the local host on which the GWrelay process is executing. The required **port** parameter specifies the inbound socket port number that is used by the GWrelay application to listen for messages from SBR Carrier.

The NumberOfRetries is optional. The **retry_value** parameter specifies the number of retry attempts to be performed by the SCTP socket when attempting to connect to the authGateway instance. Default value is 3.

The DelayBetweenRetries line is optional. The **delay_time** parameter specifies the number of seconds the SCTP socket must wait between retry attempts. Default value is 2 seconds.

The LogLevel line is optional. The **level** parameter activates logging for the GWrelay process and sets the level of details of the message. If the **level** parameter is set to 0, the entries in the GWrelay log file contain default information. If this parameter is set to 1, the entries in the GWrelay log file contain debug information. Default value is 0.

The LogPath line is optional. The **file-path** parameter specifies the path of the GWrelay log file. You can specify any valid path that exists in the SBR machine. Default path is **/opt/JNPRsbr/radius**.

The LogSize line is optional. The **file-size** parameter specifies the rollover size limit (in bytes) for the GWrelay log file. After the file size exceeds this limit, the current log file is closed and a new one is created. Default value is 2,147,483,647 bytes.

For more information, see the *SBR Carrier Installation Guide*.

Starting and Stopping the GWrelay Process

You can use the **sbrd** script to start, stop, or restart the GWrelay process. All **sbrd** commands can be executed only by the root user. The syntax for the sbrd usage is:

```
sbrd <start|stop|restart> <GWrelay> [force]
sbrd <status> <GWrelay>
```

You can use the **start**, **stop**, and **restart** arguments with the **GWrelay** option to start, stop, and restart the GWrelay process. The **status** argument displays information about the status of the GWrelay process. The **force** argument makes **sbrd** attempt to disregard or overcome any errors that occur when processing the command. The **sbrd** script halts if any error occurs, when executed without the **force** argument. For example, the **sbrd start** command does not attempt to start the software that is already running, but the **sbrd start force** command ignores a running process. This may produce unintended results, so use the **force** argument with great care.

NOTE: The GWrelay application gets terminated automatically when all the configured authGateway instances are down. So, you must manually start the GWrelay application when the authGateway instances are restarted.

If you have set the **GWRELAYENABLE** parameter in the **sbrd.conf** file to 1 or answered **Yes** to the question **Do you want to enable "GWrelay" Process? [n]:** while running the SBR Carrier configuration script, then the GWrelay process will be started, stopped, or restarted when you execute the **./sbrd start**, **./sbrd stop**, or **./sbrd restart** script respectively.

Configuring the authGateway.conf File

The **authGateway.conf** file configures the following authGateway options:

- Remote routing options control how the remote HLR is addressed based on the incoming IMSI.
- Authorization options control whether or not a subscriber requesting an account is authorized for WLAN access, and which Steel-Belted Radius Carrier profile or native user is used.
- The **FetchMSISDNRoutingInfoLCS** parameter specifies the type of message that is used to request MSISDN information from an HLR or HSS, and the **CamelSupportedPhases** parameter specifies which CAMEL phase services are supported in the network.
- Common configurations of the authGateway process.
- Process-specific options specify settings related to the authGateway process.

[Routing-Configuration] Section

Each line in the **authGateway.conf** file represents a target HLR, where each HLR has its own routing options and authorization options. Indicate each HLR listed in this file with the initial digits of the subscriber password, specified by the **odigits** option.

[Table 185 on page 520](#) lists the remote routing options for the **authGateway.conf** file.

Table 185: authGateway.conf [Routing-Configuration] Syntax

Option	Purpose
bs	Bearer Service. See “Authorization Options” on page 521.
msisdn	The msisdn option can be used in place of ndigits and odigits when no translation is required. See “Example 2—authGateway.conf file [Routing-Configuration] Section” on page 523.
ndigits	Replacement digits for numbering plan translation (hybrid IMSI).
odb	Operator-Determined Barring. See “Authorization Options” on page 521.
odigits	<p>Initial digits of IMSI or password for this HLR. For each request, the first digits of the IMSI are compared with odigits. The first line of the configuration file that matches is selected for the current request.</p> <p>If the routing indicator (rri) is 0 (Global Title), the leading digits are replaced with the new digits (ndigits) to perform the numbering plan translation.</p> <p>Example of direct replacement:</p> <p>If the rule is “odigits 12345 ndigits 98765” and the IMSI is 123456789012345, the resulting digits are 987656789012345.</p> <p>Example of wildcard replacement:</p> <p>If the rule is “odigits 12345* ndigits 98765” and the IMSI is 123456789012345, the resulting digits are 98765.</p>
rgti	(Global Title only) GTI value. 4 for C7; 2 for A7. (Usually 4.)
rnai	(Global Title only) Nature of Address Indicator.

Table 185: authGateway.conf [Routing-Configuration] Syntax (*continued*)

Option	Purpose
rn timer	(Global Title only) Numbering Plan. Acceptable values are: 1—ISDN/Telephony 3—DATA 4—TELEX 5—Maritime Mobile 6—Land/Mobile 7—ISDN/Mobile 10—British Telecom special 1 11—British Telecom special 2 14—Private Network
rpc	Remote Point Code. Point Code of HLR or MSC.
rri	Routing indicator - 0 for GT (Global Title), 1 for PC/SSN (Point Code/Subsystem Number).
rssn	Subsystem Number of HLR.
rtt	(Global Title only) Translation Type (usually 0).
ts	Teleservice. See “Authorization Options” on page 521 .

Authorization Options

The HLR database includes authorization information that is assigned to each subscriber. Three authorization designations are relevant to Steel-Belted Radius Carrier with the SIMauthentication module:

- BS (Bearer Service)
- TS (Teleservice)
- ODB (Operator-Determined Barring)

You can specify subscriber HLR authorization (and barred service) designations in the MAP Gateway `authGateway.conf` file.

NOTE: You can disable authorization completely from EAP-SIM (not fetch subscriber profile information from the HLR and not perform a SQL/LDAP query). For instructions about disabling authorization, see *“Disabling Authorization from EAP-SIM”* in the section on *Configuring the gsmmap.gen File for the SIM Authentication Module*, in the *SBR Carrier Reference Guide*.

Each line in the **authGateway.conf** file corresponds to an HLR in your network. Each line also specifies all potential authorization (and barred service) settings for any subscribers on this HLR.

Steel-Belted Radius Carrier with the SIM authentication module uses the service authorization information that you list for each HLR in **authGateway.conf**:

- When a TS or BS designation is assigned to a subscriber entry in the HLR database, Steel-Belted Radius Carrier with the SIM authentication module allows the subscriber the designated class of WLAN service upon authorization request.
- When an ODB designation is assigned to a subscriber, Steel-Belted Radius Carrier with the SIM authentication module denies the subscriber WLAN service upon authorization request.
- When you do not specify service designations for a HLR listed in **authGateway.conf**, then all subscribers on that HLR are authorized for WLAN service.
- You can specify up to six authorization strings of each type (TS, BS, or ODB) on any given line of **authGateway.conf**.

You can specify the service designations in **authGateway.conf**:

```
bs n1:auth1
ts n2:auth2
odb n3:auth3
```

Here, **ni** (**i=1,2,3**) is a decimal integer that specifies the setting, and **authi** (**i=1,2,3**) is the string returned from the MAP Gateway to Steel-Belted Radius Carrier with the SIM authentication module.

For example, you might specify the potential subscriber designations on one HLR with the following text in **authGateway.conf**:

```
bs 26:B1A ts 33:TS21 odb 128:bar
```

NOTE: If you require any HLR authorization strings to define different classes of service for your subscribers, you must also specify those TS, BS, and ODB authorization strings in certain files associated with the SIM authentication module. For information about how to match these strings to Steel-Belted Radius Carrier variables, see the “*simauth.aut [ProfileMap] Section*” of *Configuring EAP-SIM and EAP-AKA for the SIM Authentication Module* in the *SBR Carrier Reference Guide*.

Example 1—authGateway.conf file [Routing-Configuration] Section

(Lines are wrapped.)

```
odigits 2310 ndigits 2324 rnai 4 rnp 7 rgti 4 rtt 0 rri 0 rpc 3003 rsn 251 bs 12:gold bs 23:silver ts 91:bronze
ts 92:red ts 93:green odb 1:black aqua
```

```
odigits 31026 ndigits 32476 rnai 4 rnp 7 rgti 4 rtt 0 rri 1 rpc 3003 rsn 253 bs 23:morning bs 24:afternoon
ts 1:night
```

Example 2—authGateway.conf file [Routing-Configuration] Section

In this global title example, odigits and ndigits are the same and do not require translation. You can use the **msisdn** option in place of ndigits and odigits when no translation is required.

(Lines are wrapped.)

```
msisdn 31026 rnai 4 rnp 7 rgti 4 rtt 0 rri 0 rpc 3003 rsn 251 bs 12:gold bs 23:silver ts 91:bronze ts 92:red
ts 93:green odb 1:black aqua
```

[Supported-MAP-Messages] Section

The [Supported-MAP-Messages] section ([Table 186 on page 524](#)) of the **authGateway.conf** file specifies the method of fetching MSISDN and which CAMEL phase services are supported in the network.

Table 186: authGateway.conf [Supported-MAP-Messages] Syntax

Parameter	Description
FetchMSISDNRoutingInfoLCS	<p>Specifies the type of message that is used to fetch MSISDN information from an HLR or HSS.</p> <p>MSISDN information is usually fetched from an HLR through the d interface using the RestoreData message. Setting the FetchMSISDNRoutingInfoLCS parameter to 0 configures the authGateway process to interact with the HLR through the RestoreData message. The default SSN configured when the authGateway process starts is used as the originating SSN in the RestoreData message.</p> <p>MSISDN information is usually fetched from an HLR or HSS through the SLh or Lh interface using the SendRoutingInfoForLCS message. Setting the FetchMSISDNRoutingInfoLCS parameter to 1 configures the authGateway process to interact with the HLR or HSS through the SendRoutingInfoForLCS message. Because SBR Carrier acts as a GMLC in this case, the GMLC SSN (i.e. 145) is used as the originating SSN.</p> <p>By default, this parameter is set to 0.</p>
CamelSupportedPhases	<p>Specifies which CAMEL phase services are supported in the network.</p> <ul style="list-style-type: none"> • If set to 0 or commented out, the RestoreData message populates only the mandatory parameter IMSI. • If set to 1, the network supports CAMEL phase 1 services. • If set to 2, the network supports CAMEL phase 2 services. • If set to 3, the network supports CAMEL phase 3 services. • If set to 4, the network supports CAMEL phase 4 services. <p>By default, this parameter is set to 0.</p>

[Common-AGW-Configurations] Section

The [Common-AGW-Configurations] section ([Table 187 on page 524](#)) of the **authGateway.conf** file specifies common configurations for the authGateway process.

Table 187: authGateway.conf [Common-AGW-Configurations] Syntax

Option	Description
appctx	<p>Specifies the MAP protocol revision (2 or 3). Only MAPv3 retrieves quintets, so it must be used to support EAP-AKA.</p> <p>Default value is 2.</p>

Table 187: authGateway.conf [Common-AGW-Configurations] Syntax (*continued*)

Option	Description
connretry	Specifies the number of connection attempts. Default value is 10.
conntimeout	Specifies the connection timeout in minutes. Default value is 0 minute.
host	Specifies the local hostname. You must use the hostname associated with the IP address that the authGateway listen on. Also, you must ensure that the entry is coordinated with the radius/GWrelay.conf file (if authGateway is running as multiple instances) and radius/conf/ulcmng.conf (if authGateway is running as a single instance) files. If a hostname is not specified, 0.0.0.0 is used.
invkretry	Specifies the number of invoke retries. Default value is 1.
invktimeout	Specifies the duration of invoke timeout in seconds. Default value is 30 seconds.
lgti	(Global Title only) Specifies the local GTI value, usually 4 for C7 and 2 for A7. Default value is 4.
lmsisdn	(Global Title only) Specifies the MSISDN value of the local node. Default value is 0.
lnai	(Global Title only) Specifies the scope of the address value, such as whether it is an international number (includes country code) or a national number (no country code). <ul style="list-style-type: none"> • 1—Subscriber number with no area code (example: 5551234) • 2—Unused • 3—National significant number with no country code (example: 2015551234) • 4—International number with country code (example: 12015551234) Default value is 4.

Table 187: authGateway.conf [Common-AGW-Configurations] Syntax (*continued*)

Option	Description
lnp	<p>(Global Title only) Specifies the local numbering plan. Acceptable values are:</p> <ul style="list-style-type: none"> • 1—ISDN/Telephony • 3—DATA • 4—TELEX • 5—Maritime Mobile • 6—Land/Mobile • 7—ISDN/Mobile • 10—British Telecom special 1 • 11—British Telecom special 2 • 14—Private Network <p>Default value is 1.</p>
lpc	<p>Specifies the Local Point Code (PC).</p> <p>Default value is 0.</p>
lri	<p>Specifies the routing indicator used to address messages to the local node.</p> <ul style="list-style-type: none"> • 0—Global Title • 1—PC/SSN <p>Default value is 1.</p>
ltt	<p>(Global Title only) Specifies local translation type.</p> <p>Default value is 0.</p>
max_requests	<p>Specifies the maximum number of concurrent requests that can be handled by the authGateway process.</p> <p>Default value is 1000.</p>
monitor	<p>Activates the message activity monitor.</p>
no_rst	<p>Disables automatic restart of the authGateway process.</p>
node	<p>Specifies the node name.</p>

[Process<name>] Section

The [Process<name>] section (Table 188 on page 527) of the **authGateway.conf** file contains the parameters that control authGateway process specific behavior.

Table 188: authGateway.conf [Process<name>] Syntax

Option	Description
debug	Specifies a debug level. Default value is 0. NOTE: This parameter is reloaded whenever SBR Carrier receives a SIGHUP (1) signal.
lssn	Specifies the local subsystem number. Default value is 7.
port	Specifies the port number used by the SCTP association with the client.
prot	Specifies the variant used (C7, A7, or CH7).
trace	Enables debug tracing and displays the trace information about the console. (Consists of a trace of all MAP messages that are formatted and sent down the stack.) NOTE: We recommend setting this parameter to 0. NOTE: This parameter is reloaded whenever SBR Carrier receives a SIGHUP (1) signal.
tracefile	Captures the trace information to a file. The filename follows the -tracefile switch. Include the directory in the filename. NOTE: This parameter is reloaded whenever SBR Carrier receives a SIGHUP (1) signal.

Example

If you are using two authGateway processes—for example, **GMT1** and **GMT2**, then two sections **[ProcessGMT1]** and **[ProcessGMT2]** must be added to the **authGateway.conf** file for the authGateway processes to startup. The following example explains this configuration:

```
[ProcessGMT1]

#Remote port specified in ulcmmg.conf
#Port number used by the SCTP association with the client
port=2003

#Variant used (C7, A7 or CH7)
prot=C7
```

```

#Enables Signalware library debug logging. Sets a debug level.
debug=1

#This enables debug tracing and displays the trace information about the console.
#Consists of a trace of all MAP messages that are formatted and sent down the stack.
#Use the tracefile option to capture the trace information to a file
trace=1

#Captures the trace information to a file. The filename follows the tracefile
#switch. Include the directory in the filename
tracefile=/opt/JNPRsbr/radius/conf/Trace1.out

[ProcessGMT2]

#Remote port specified in ulcmmg.conf
#Port number used by the SCTP association with the client
port=2005

#Variant used (C7, A7 or CH7)
prot=C7

#Enables Signalware library debug logging. Sets a debug level.
debug=1

#This enables debug tracing and displays the trace information about the console.
#Consists of a trace of all MAP messages that are formatted and sent down the stack.
#Use the tracefile option to capture the trace information to a file
trace=1

#Captures the trace information to a file. The filename follows the tracefile
#switch. Include the directory in the filename
tracefile=/opt/JNPRsbr/radius/conf/Trace2.out

```

Configuring the authGateway Startup with MML Commands

The **CREATE-PROCESS** and **START-PROCESS** MML commands start the authGateway (by calling **authGateway.conf**), using options that you specify.

[Table 189 on page 529](#) describes the MML commands needed to configure and start authGateway.

Table 189: MML Commands for Configuring the Start of authGateway

MML Command	Description
CREATE-PROCESS	Identify the authGateway configuration file and the authGateway options.
START-PROCESS	Start the process.

For more information about the syntax and usage of the MML commands, see [“Signalware MML Commands” on page 504](#). See SBR Carrier Installation Guide for information about executing the MML commands.

[Table 190 on page 529](#) lists the mandatory MML options to be used with the **CREATE-PROCESS** command.

Table 190: authGateway Process Options Used with CREATE-PROCESS

Option	Description
conf	Path and name of the authGateway configuration file. The default file is \$RADIUSDIR/conf/authGateway.conf .
name	Name of the authGateway process.

The MML options listed in [Table 191 on page 529](#) are still supported for backward compatibility.

NOTE: Starting from SBR Carrier 8.4.0 release, the **authGateway.conf** file configuration takes precedence over any existing MML CREATE-PROCESS options. While upgrading to SBR Carrier Release 8.4.0, the **authGateway.conf** file will be imported from older versions of SBR Carrier. If there is no specified configuration present in the imported **authGateway.conf** file, SBR Carrier uses the existing MML configurations. If no MML configurations are present, the **authGateway.conf** defaults are used. If any mandatory parameters (port, host, and node) are missing, then error messages are logged.

Table 191: authGateway Process Options Supported for Backward-Compatibility

Option	Description
appctx	MAP protocol revision (2 or 3). Only MAPv3 retrieves quintets, so it must be used to support EAP-AKA.
debug	Sets a debug level. Use the following: -debug 0xff

Table 191: authGateway Process Options Supported for Backward-Compatibility (continued)

Option	Description
host	Local hostname. Use the hostname associated with the IP address that the authGateway listen on, and ensure that the entry is coordinated with the radius/GWrelay.conf file. If a hostname is not specified, 0.0.0.0 is used.
invkretry	Number of invoke retry.
invktimeout	Duration of invoke timeout in seconds.
lgti	(Global Title only) Local GTI value, usually 4 for C7 and 2 for A7.
lmsisdn	(Global Title only) MSISDN of this local node.
lnai	<p>(GT only) Nature of Address Indicator. Indicates the scope of the address value, such as whether it is an international number (includes country code) or a national number (no country code).</p> <p>1 Subscriber Number—no area code (example: 5551234)</p> <p>2 unused</p> <p>3 National Significant Number—no country code (example: 2015551234)</p> <p>4 International Number—includes country code (example: 12015551234)</p>
lnp	<p>(Global Title only) Local Numbering Plan.</p> <p>Acceptable values are:</p> <p>1—ISDN/Telephony</p> <p>3—DATA</p> <p>4—TELEX</p> <p>5—Maritime Mobile</p> <p>6—Land/Mobile</p> <p>7—ISDN/Mobile</p> <p>10—British Telecom special 1</p> <p>11—British Telecom special 2</p> <p>14—Private Network</p>

Table 191: authGateway Process Options Supported for Backward-Compatibility (continued)

Option	Description
lpc	Local Point Code (PC).
lri	Routing indicator - 0 for GT (Global Title), 1 for PC/SSN.
lssn	Local Subsystem Number (SSN) (required).
ltn	(Global Title only) Local Translation Type. Generally in a live network TT is always 0.
max_requests	The maximum number of simultaneous MAP dialogs.
monitor	Activates Message Activity Monitor.
no rst	Disables automatic restart of process.
node	Node name.
port	Port number used by the SCTP association with the client.
prot	Variant used (C7, A7, or CH7).
rssn	Subsystem number of HLR.
trace	<p>We recommend setting this to 0xff; this enables debug tracing and displays the trace information about the console. (Consists of a trace of all MAP messages that are formatted and sent down the stack.)</p> <p>Use the tracefile option to capture the trace information to a file.</p>
tracefile	Captures the trace information to a file. The filename follows the -tracefile switch. Include the directory in the filename.

Example—Creating and Starting the authGateway Process

We recommend that you specify an absolute (full) path in the **EXEC** command. The following configuration example explains how to create and start three authGateway instances:

(Lines are wrapped.)

```
CREATE-PROCESS:NAME="GMT", CE="sbr-lnx-perf", EXEC="/opt/JNPRsbr/radius/authGateway
-name GMT -conf /opt/JNPRsbr/radius/conf/authGateway.conf -port 2003"
```

```
START-PROCESS:NAME="GMT",CE="sbr-lnx-perf";

CREATE-PROCESS:NAME="GMT1", CE="sbr-lnx-perf", EXEC="/opt/JNPRsbr/radius/authGateway

-name GMT1 -conf /opt/JNPRsbr/radius/conf/authGateway.conf -port 2005"

START-PROCESS:NAME="GMT1",CE="sbr-lnx-perf";

CREATE-PROCESS:NAME="GMT2", CE="sbr-lnx-perf", EXEC="/opt/JNPRsbr/radius/authGateway

-name GMT2 -conf /opt/JNPRsbr/radius/conf/authGateway.conf -port 2007"

START-PROCESS:NAME="GMT2",CE="sbr-lnx-perf";
```

NOTE: MML commands are saved in MML files that can be loaded into Signalware. See the *SBR Carrier Installation Guide* for more information.

5

PART

Optional Mobility Module Configuration Files

WiMAX Mobility Module Configuration File | 534

WiMAX Mobility Module Configuration File

IN THIS CHAPTER

- [wimax.ini File](#) | 534

This chapter describes the configuration file used by Steel-Belted Radius Carrier to configure the optional WiMAX mobility module. The following topics are included in this chapter:

wimax.ini File

The **wimax.ini** configuration file contains parameters that control basic behavior of the WiMAX mobility module. You must configure the **wimax.ini** file for the WiMAX features you want Steel-Belted Radius Carrier to support. The features described in the **wimax.ini** file require a WiMAX mobility module license key, which is entered during installation. For details about installing Steel-Belted Radius Carrier, see the *SBR Carrier Installation Guide*.

NOTE: The **wimax.ini** file is read whenever Steel-Belted Radius Carrier restarts or receives a SIGHUP (1) signal. All other parameters in the [Settings] section can be updated by a SIGHUP (1) signal. See the **UpdateWiMAX** parameter in the “[update.ini File](#)” on page 120.

The **wimax.dct** file shipped with Steel-Belted Radius Carrier has been configured with the attributes necessary for supporting WiMAX in compliance with the WiMAX Forum Network Working Group standards.

[Settings] Section

The [Settings] section ([Table 192 on page 535](#)) contains the settings that control the basic operation of the WiMAX mobility module.

Table 192: wimax.ini [Settings] Syntax

Parameter	Function
Add-Diagnostic-Reply-Message-To-Access-Reject	<p>When an Access-Reject is rejected, a programmatically-generated Reply-Message attribute can be added to the Access-Reject. The Reply-Message contents may be used for diagnostic purposes.</p> <ul style="list-style-type: none"> • If set to 0, do not add a Reply-Message to the Access-Reject. • If set to 1, add a Reply-Message to the Access-Reject. <p>Default value is 0.</p>
Add-Funk- WiMAX-Client-Type-To-Request	<p>The Funk-WiMAX-Client-Type attribute contains an integer value that specifies the type of RADIUS client sending the Access-Request or Accounting-Request. This information may be of use with scripts or stored procedures.</p> <ul style="list-style-type: none"> • If set to 0, do not attach Funk-WiMAX-Client-Type attribute • If set to 1, attach Funk-WiMAX-Client-Type attribute. <p>Default value is 0.</p>
Allow-Zero-WiMAX-MN-hHA-MIP4-SPI	<p>Specifies whether SBR Carrier should honor a request for an MIP4 key if the request contains a WiMAX-MN-hHA-MIP4-SPI attribute with a value of 0.</p> <ul style="list-style-type: none"> • If set to 1, SBR Carrier returns the corresponding key, if present in the session database, in Access-Accept. • If set to 0, SBR Carrier does not return the key, even if it is present in the session database. <p>Default value is 0.</p>

Table 192: wimax.ini [Settings] Syntax (continued)

Parameter	Function
Chargeable-User-Identity-Type	<p>Specifies the value of the Chargeable-User-Identity (CUI) attribute to attach to the Access-Accept. This value can be programmatically-generated or configured. Possible values are:</p> <ul style="list-style-type: none"> • Session-Id • Return-List-Attr • True-Identity <p>Default value is Return-List-Attr.</p> <p>The same CUI value is sent to both the ASN-GW and home agent. Because the CUI is attached to all Accounting-Requests, it can be used to match the accounting records associated with the ASN-GW and home agent, and for a single Mobile IP (MIP) session.</p> <p>If you want to return a specific value for the CUI, you need to set this parameter to Return-List-Attr, and configure the attribute in a return list in either the User or Profile entry. The CUI attribute is attached to the Access-Accept message.</p> <p>If you want the true identity of the user to be sent in the CUI, select True-Identity. For EAP-TTLS, the true identity is the inner identity. For EAP-AKA, the true identity is the Permanent Identity. These identify the actual username used for authentication by Steel-Belted Radius Carrier not a pseudo-identity or alternate identity.</p> <p>If you want each MIP session to be uniquely identified, select Session-Id. The AAA-Session-Id sent to the ASN-GW is used as a unique identifier of the MIP session.</p> <p>NOTE: This setting is applicable only for SBR acting as HAAA.</p>

Table 192: wimax.ini [Settings] Syntax (continued)

Parameter	Function
Check-CN-In-TTLS-Client-Certificate	<p>Enables and disables checking of the Common Name (CN) field of a client certificate in TTLS authentication.</p> <p>If enabled, the MAC Address field of the client certificate is verified against the Calling-Station-Id in the outer Access-Request; if they do not match, the request is rejected.</p> <ul style="list-style-type: none"> • If set to 0, checking of the CN field is disabled. • If set to 1, the CN is required to start with the 12 character hex representation of the MAC address, which must match the Calling-Station-Id request attribute (according to WiMAX specifications). Non-hex characters in the Calling-Station-Id are skipped in the check.
DHCP-RK-Lifetime-Secs	<p>Specifies the DHCP-RK (root key) lifetime for all DHCP servers in seconds. The DHCP-RK key is cryptographic key and is a random number generated by the AAA server.</p> <p>Default value is 86400 seconds (24 hours).</p>
DisableHaPhantom	<p>Enables or disables the creation of phantom records for authentication messages from the home agent.</p> <p>If this parameter is enabled, Steel-Belted Radius Carrier does not create phantom sessions for authentication requests from the home agent.</p> <ul style="list-style-type: none"> • If set to 0, phantom session records are enabled for authentication messages from the home agent. • If set to 1, phantom session records are disabled for authentication messages from the home agent. <p>Default value is 0.</p>
Enable	<p>Specifies whether the WiMAX mobility module is enabled.</p> <ul style="list-style-type: none"> • If set to 0, WiMAX is disabled. • If set to 1, WiMAX is enabled. <p>Default value is 0.</p>

Table 192: wimax.ini [Settings] Syntax (continued)

Parameter	Function
Encrypt-Chargeable-User-Identity	<p>Specifies whether to salt-encrypt the value of the Chargeable-User-Identity attribute attached to the Access-Accept.</p> <ul style="list-style-type: none"> • If set to 0, do not salt-encrypt the Chargeable-User-Identity attribute. • If set to 1, salt-encrypt the Chargeable-User-Identity attribute. <p>Default value is 1.</p> <p>Setting this value to 1 ensures the user identity is uniquely encrypted for each session. In WiMAX, even when an identity is encrypted, if it is encrypted in the same way each time (encryption of the identity results in the same cipher text each time), then the user's network traffic can be identified and tracked, even if the true identity of the user is not known. When Steel-Belted Radius Carrier salt-encrypts the CUI, the cipher text value is different for each encryption. Encryption is especially important when the CUI contains the true identity of the user.</p> <p>NOTE: This setting is applicable only for SBR acting as HAAA.</p>
HA-Dynamic -Addr-Weight -File = <path/filename>	<p>Specifies the path and filename of the dynamically updated file used by the smart dynamic home agent assignment feature. This file contains pairs of IP addresses and weights, and is read by SBR Carrier upon the receipt of a signal (either SIGHUP (1) or SIGUSR2 (17), as defined in update.ini file. For more information about the smart dynamic home agent assignment feature, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p>
HARK-Lifetime-Secs	<p>Specifies the HA-RK (root key) lifetime for all home agents in seconds. The HA-RK key is cryptographic key and is a random number generated by the AAA server.</p> <p>Default value is 86400 seconds (24 hours).</p>

[ASNGW-Requests] Section

The [ASNGW-Requests] section ([Table 193 on page 539](#)) contains the settings that control the processing of ASN-GW (Access Server Network-Gateway) requests.

Table 193: wimax.ini [ASNGW-Requests] Syntax

Parameter	Function
Accept-ASNGW-Requests	<p>Specifies whether ASN-GW request processing is enabled.</p> <ul style="list-style-type: none">• If set to 0, ASN-GW request processing is disabled. If an Access-Request is received from an ASN-GW, the request is rejected.• If set to 1, ASN-GW request processing is enabled. If an Access-Request is received from an ASN-GW, the request is processed. <p>Default value is 0.</p>

Table 193: wimax.ini [ASNGW-Requests] Syntax (*continued*)

Parameter	Function
Add-Funk-WiMAX-Auth-Mode-To-Access-Request	<p>Specifies whether to attach the Funk-WiMAX-Auth-Mode attribute to the Access-Request. The Funk-WiMAX-Auth-Mode attribute contains the numeric value to the right of the equal sign in any {am=} decoration prepended to the User-Name. For example, if the User-Name contains {am=2} joe@bigco.com, then the Funk-WiMAX-Auth-Mode attribute value is 2. This information is useful for scripts and stored procedures.</p> <ul style="list-style-type: none"> • If set to 0, do not attach the Funk-WiMAX-Auth-Mode attribute to the Access-Request. • If set to 1, attach the Funk-WiMAX-Auth-Mode attribute to the Access-Request. <p>Default value is 0.</p> <p>NOTE: For authentication methods with both inner and outer authentication such as EAP-PEAP and EAP TTLS, the Funk-WiMAX-Auth-Mode attribute is set in the outer authentication method. To transfer it to the inner authentication method, the EAP method must be configured to pass the outer attributes to the inner request by setting the Transfer_Outer_Attribs_to_New parameter. This is set either in the .aut configuration file, or on the Request Filters tab in the EAP Methods List page in Web GUI.</p> <p>NOTE: For more details about the Funk-WiMAX-Auth-Mode attribute, see the radius.dct dictionary that is shipped with Steel-Belted Radius Carrier.</p>
Add-Generated-PMIP-Auth-Id-To-Access-Accept	<p>Steel-Belted Radius Carrier can optionally generate the PMIP-Authenticated-Identity. This parameter specifies whether to add the value for the PMIP-Authenticated-Identity to the Access-Accept.</p> <ul style="list-style-type: none"> • If set to 0, do not attach the PMIP-Authenticated-Identity to the Access-Accept. • If set to 1, attach the PMIP-Authenticated-Identity to the Access-Accept. <p>Default value is 0.</p>

Table 193: wimax.ini [ASNGW-Requests] Syntax (*continued*)

Parameter	Function
Add-MSK-To-Access-Accept	<p>Specifies whether to add the WiMAX-MSK to the Access-Accept.</p> <p>MPPE keys will go out if configured in EAP-TLS, EAP-TTLS, EAP-SIM or EAP-AKA plug-in.</p> <ul style="list-style-type: none"> • If set to 1, WiMAX-MSK is added to the Access-Accept. • If set to 0, WiMAX-MSK is not added to the Access-Accept. <p>The default value is 0.</p>
Allow-VAAA-To-Assign-Home-Agent-And-DHCP-Server	<p>Specifies whether or not to allow the VAAA server to assign the home agent and DHCP server IP addresses. If the VAAA server can assign the home agent and DHCP server IP addresses, it attaches the vHA-IP-MIP4 attribute to the Access-Request it proxies to the Home Authentication, Authorization, and Accounting (HAAA) server. If the HAAA server is configured to allow the VAAA server to assign the home agent and DHCP server IP addresses, then the HAAA server attaches that same vHA-IP-MIP4 attribute to the Access-Accept returned to the VAAA server. For more information about configuring the home agent and DHCP server when using WiMAX, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <ul style="list-style-type: none"> • If set to 0, do not allow the VAAA server to assign the home agent and DHCP server IP addresses. • If set to 1, allow the VAAA server to assign the home agent and DHCP server IP addresses. If this parameter is set to 1 and the VAAA server attaches the vHA-IP-MIP4 attribute to the Access-Request, then the HAAA server attaches the following additional attributes to the Access-Accept: vHA-IP-MIP4, MN-vHA-MIP4-KEY, and MN-vHA-MIP4-SPI. <p>Default value is 0.</p>

Table 193: wimax.ini [ASNGW-Requests] Syntax (*continued*)

Parameter	Function
ASNGW-Accept-Filter	<p>Specifies the name of the attribute filter to be applied to the ASN-GW Access-Accept parameter before the session is recorded. You can use this parameter to specify regular or scripted filters. If no filter is specified, all attributes are returned unchanged.</p> <p>If no filter is specified, all attributes are returned unchanged.</p> <p>NOTE: You must define all filters using the Web GUI. Do not edit the filter.ini file manually. For more information, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is no filter.</p>
ASNGW-PostSession-Filter	<p>Specifies the name of the attribute filter to be applied to the ASN-GW Access-Accept parameter after the session is recorded. You can use this parameter to specify regular or scripted filters. If no filter is specified, all attributes are returned unchanged.</p> <p>NOTE: You must define all filters using the Web GUI. Do not edit the filter.ini file manually. For more information, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is no filter.</p>

[ASNGW-Requests/<name>] Section

Multiple sections with names of the style [ASNGW-Requests/< **name** >] can also exist in the **wimax.ini** file. These sections are only referenced when a proxy realm's configuration file (**.pro**) contains an ASNGW-Requests-Section setting in its [WiMAX] section.

Specifying this option in a realm's configuration file puts the options in the matching ASNGW user authentication request processing section of the **wimax.ini** file in effect for all ASNGW user authentication request transactions that are processed by the proxy realm. As a result, the settings in this section are used instead of the settings in the [ASNGW-Requests] section for transactions processed against the proxy realm.

The options in these sections are identical to those documented for the [ASNGW-Requests] section, which is described on “[ASNGW-Requests] Section” on page 539.

NOTE: This section only applies to the WiMAX VAAA configuration.

[Home-Agent-Requests] Section

The [Home-Agent-Requests] section ([Table 194 on page 543](#)) contains the settings that control the processing of the home agent requests.

Table 194: wimax.ini [Home-Agent-Requests] Syntax

Parameter	Function
Accept-Home-Agent-Requests	<p>Specifies whether home agent Access-Request processing is enabled.</p> <ul style="list-style-type: none"> • If set to 0, home agent request processing is disabled. If an Access-Request is received from a home agent, the request is rejected. • If set to 1, home agent request processing is enabled. If an Access-Request is received from a home agent, the request is processed. <p>Default value is 0.</p>
Add-Funk-Full-User-Name-To-Access-Request	<p>Specifies whether to attach the Funk-Full-User-Name attribute to the Access-Request. The Funk-Full-User-Name attribute contains the true identity of the user. For the EAP-TTLS method, this is the inner identity, for the EAP-TLS method, this is the identity obtained from the certificate, for the EAP-AKA method, this is the permanent identity.</p> <ul style="list-style-type: none"> • If set to 0, do not attach the Funk-Full-User-Name attribute to the Access-Request. • If set to 1, attach the Funk-Full-User-Name attribute to the Access-Request. <p>Default value is 0.</p> <p>NOTE: For more details about the Funk-Full-User-Name attribute, see the radius.dct dictionary that is shipped with Steel-Belted Radius Carrier.</p>

Table 194: wimax.ini [Home-Agent-Requests] Syntax (*continued*)

Parameter	Function
Check-Rcvd-HA-IP-MIP-Same-As- Assigned-By-HAAA	<p>Specifies whether to check if the home HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW specifies the IP address of the assigned home agent in the home network. The home HA-IP-MIP4 attribute received from a home agent identifies that particular home agent.</p> <ul style="list-style-type: none"> • If set to 0, do not check the home HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW. • If set to 1, check the home HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW. If the received home HA-IP-MIP4 attribute is not the assigned home HA-IP-MIP4 attribute, then the home agent request is rejected. <p>Default value is 0.</p> <p>NOTE: For this parameter to work, you need to uncomment the Sbr_HaType column in the WimaxTables.sql script and re-create the database. For the standalone version of SBR, the Sbr_HaType column is available and this parameter works by default. The Sbr_HaType column in the WiMAX table is optional.</p>
Check-Rcvd-HA-IP-MIP-Same-As- Assigned-By-VAAA	<p>Specifies whether to check if the visited HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW specifies the IP address of the assigned home agent in the visited network. The visited HA-IP-MIP4 attribute received from a home agent identifies that particular home agent.</p> <ul style="list-style-type: none"> • If set to 0, do not check the visited HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW. • If set to 1, check the visited HA-IP-MIP4 attribute sent from the home AAA server to the ASN-GW. If the received visited HA-IP-MIP4 attribute is not the assigned visited HA-IP-MIP4 attribute, then the home agent request is rejected. <p>Default value is 0.</p>

Table 194: wimax.ini [Home-Agent-Requests] Syntax (*continued*)

Parameter	Function
Home-Agent-Accept-Filter	<p>Specifies the name of the attribute filter to be applied to the home agent Access-Accept. You can use this parameter to specify regular or scripted filters.</p> <p>If no filter is specified, all attributes are returned unchanged.</p> <p>NOTE: You must define all filters using the Web GUI. Do not edit the filter.ini file manually. For more information, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is no filter.</p>

[DHCP-Server-Requests] Section

The [DHCP-Server-Requests] section ([Table 195 on page 545](#)) contains the settings that control the processing of the DHCP server requests.

Table 195: wimax.ini [DHCP-Server-Requests] Syntax

Parameter	Function
Accept-DHCP-Server-Requests	<p>Specifies whether DHCP server request processing is enabled.</p> <ul style="list-style-type: none"> • If set to 0, any DHCP server request is rejected. • If set to 1, DHCP server request processing is enabled. <p>Default value is 0.</p>

Table 195: wimax.ini [DHCP-Server-Requests] Syntax (*continued*)

Parameter	Function
DHCP-Server-Accept-Filter	<p>Specifies the name of the attribute filter to be applied to the DHCP server Access-Accept. You can use this parameter to specify regular or scripted filters.</p> <p>If no filter is specified, all attributes are returned unchanged.</p> <p>NOTE: You must define all filters using the Web GUI. Do not edit the filter.ini file manually. For more information, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is no filter.</p>

[Other-Requests] Section

The [Other-Requests] section ([Table 196 on page 546](#)) specifies how other Accept-Requests (ones that do not fit in any of the other categories of ASN-GW, home agent, or DHCP server) from a client are handled.

Table 196: wimax.ini [Other-Requests] Syntax

Parameter	Function
Other-Accept-Filter	<p>Specifies the name of the filter to be applied to attributes in an Access-Accept in response to all requests of type Other. You can use this parameter to specify regular or scripted filters.</p> <p>If no filter is specified, all attributes are returned unchanged.</p> <p>NOTE: You must define all filters using the Web GUI. Do not edit the filter.ini file manually. For more information, see the <i>SBR Carrier Administration and Configuration Guide</i>.</p> <p>Default value is no filter.</p>

Table 196: wimax.ini [Other-Requests] Syntax (*continued*)

Parameter	Function
Pass-On-Other-Requests	<p>Specifies whether Access-Request processing is enabled from a RADIUS client that is not an ASN-GW, home agent, or DHCP server.</p> <ul style="list-style-type: none"> • If set to 0 (disabled) and an Access-Request is received from such a client, then the request is rejected. • If set to 1 (enabled), then the WiMAX mobility modules apply the filter (specified in the Other-Accept-Filter parameter). <p>Default value is 0.</p>

[HAs] Section

The [HAs] section lists the NAS-Identifier (for example, **homeAgent.bigco.com**) of each home agent from which an Access-Request is processed.

- If the list is not empty and the received NAS-Identifier is not in the list, then the Access-Request is rejected.
- If the list is empty, then Access-Requests from all home agents are processed.

[DHCPServers] Section

The [DHCPServers] section lists the NAS-Identifier (for example, **dhcpServer.bigco.com**) of each DHCP server from which an Access-Request is processed.

- If the list is not empty and the received NAS-Identifier is not in the list, then the Access-Request is rejected.
- If the list is empty, then Access-Requests from all DHCP servers are processed.

[RADIUS client-Access-Request-Required-Attributes] Sections

These sections list the attributes that must be present in an Access-Request to classify the RADIUS client as a WiMAX ASN-GW, home agent, DHCP server, or something else (Other).

- [ASNGW-Access-Request-Categorization-Attributes] section
If all attributes in the [ASNGW-Access-Request-Categorization-Attributes] section are present on the Access-Request, then the client is classified as ASN-GW; if not, check the attributes in the [Home-Agent-Access-Request-Categorization-Attributes] section.

```
[ASNGW-Access-Request-Categorization-Attributes]
User-Name
Service-Type
EAP-Message
WiMAX-Capability
NAS-Identifier
NAS-Port-Type
Calling-Station-Id
GMT-Time-Zone-Offset
```

- [Home-Agent-Access-Request-Categorization-Attributes] section

If all attributes in the [Home-Agent-Access-Request-Categorization-Attributes] section are present on the Access-Request, then the client is classified as a home agent; if not, check the attributes in the [DHCP-Server-Access-Request-Required-Attributes] section.

```
[Home-Agent-Access-Request-Categorization-Attributes]
User-Name
NAS-Identifier
WiMAX-Capability
MN-HA-MIP4-SPI
```

- [DHCP-Server-Access-Request-Categorization-Attributes] section

If all attributes in the [DHCP-Server-Access-Request-Categorization-Attributes] section are present on the Access-Request, then the client is classified as a DHCP server; if not, then the client is classified as other.

```
[DHCP-Server-Access-Request-Categorization-Attributes]
NAS-Identifier
DHCP-RK-Key-ID
```

For more information about how Steel-Belted Radius Carrier categorizes Access-Request messages when using WiMAX, see the *SBR Carrier Administration and Configuration Guide*.

Example wimax.ini File

```
[Settings]
;Enable = 0
;HARK-Lifetime-Secs=86400
;DHCP-RK-Lifetime-Secs=86400
;Add-Diagnostic-Reply-Message-To-Access-Reject = 0
;Chargeable-User-Identity-Type = Return-List-Attr
```

```

;Encrypt-Chargeable-User-Identity = 1
;Add-Funk-WiMAX-Client-Type-To-Request = 0
;DisableHaPhantom=0

[ASNGW-Requests]
;Accept-ASNGW-Requests = 0
;Allow-VAAA-To-Assign-Home-Agent-And-DHCP-Server = 0
;Add-Generated-PMIP-Auth-Id-To-Access-Accept= 0
;Add-Funk-WiMAX-Auth-Mode-To-Access-Request = 0
;ASNGW-Accept-Filter =

[Home-Agent-Requests]
;Accept-Home-Agent-Requests = 0
;Add-Funk-Full-User-Name-To-Access-Request = 0 ;Contains true identity
;Check-Rcvd-HA-IP-MIP-Same-As-Assigned-By-HAAA = 0
;Check-Rcvd-HA-IP-MIP-Same-As-Assigned-By-VAAA = 0
;Home-Agent-Accept-Filter =

[DHCP-Server-Requests]
;Accept-DHCP-Server-Requests = 0
;DHCP-Server-Accept-Filter =

[Other-Requests]
;Pass-On-Other-Requests = 0
;Other-Accept-Filter =

[HAs]
;homeAgent.bigco.com

[DHCPServers]
;dhcpServer.bigco.com

[ASNGW-Access-Request-Categorization-Attributes]
User-Name
Service-Type
EAP-Message
WiMAX-Capability
NAS-Identifier
NAS-Port-Type
Calling-Station-Id
WiMAX-GMT-Time-Zone-Offset

[Home-Agent-Access-Request-Categorization-Attributes]
User-Name

```

NAS-Identifier

WiMAX-Capability

WiMAX-MN-HA-MIP4-SPI

[DHCP-Server-Access-Request-Categorization-Attributes]

NAS-Identifier

WiMAX-DHCP-RK-Key-ID

6

PART

SNMP Configuration Files

[SNMP Configuration Overview | 552](#)

[SNMP Traps and Statistics Overview | 565](#)

SNMP Configuration Overview

IN THIS CHAPTER

- [SNMP Overview | 552](#)
- [jnprsnmpd.conf File Overview | 556](#)
- [testagent.sh Script Overview and Syntax | 563](#)

This chapter describes how to configure the Steel-Belted Radius Carrier server Simple Network Management Protocol (SNMP) package. These topics are included in this chapter:

See the *SBR Carrier Administration and Configuration Guide* for information about configuring and running SNMP, MIBs, and troubleshooting SNMP.

See “[SNMP Traps and Statistics Overview](#)” on [page 565](#) in this guide for a list of supported SNMP traps and statistics, and see “[events.ini File](#)” on [page 18](#) in this guide for information about trap dilution settings.

SNMP Overview

SNMP is an IETF standard protocol that enables an administrator to set configuration parameters and monitor operating statistics and status for a managed device, such as a server or router, from a remote location.

In addition to supporting routine monitoring of the server as a device, the Steel-Belted Radius Carrier software includes proprietary MIBs (Management Information Bases) that support monitoring and interaction with the SBR Carrier software applications.

SNMP Network Management Architecture Overview

The SNMP network management architecture consists of *managed devices*, *SNMP agents*, and *network management stations* (NMS).

- A managed device is any host or hardware on a network that runs an SNMP agent. The Steel-Belted Radius Carrier server is a managed device after you install and configure the optional SNMP agent.

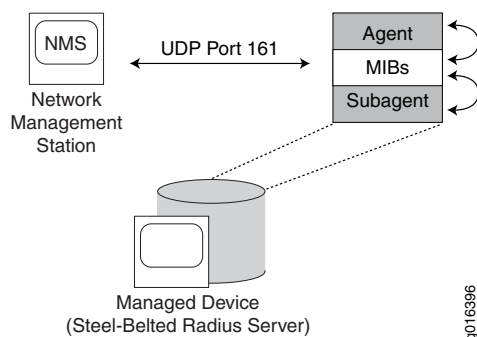
- An SNMP agent is a software module running on a managed device that is responsible for recording performance statistics and events in a MIB and for communicating that information with the NMS. The SNMP agent for Steel-Belted Radius Carrier is called **jnpnsnmpd**. When an NMS requests information, the SNMP agent processes the request, acquires information from the management database, and forwards the information to the NMS. The SNMP agent can also accept control information from the NMS.

An SNMP subagent may be responsible for gathering information about network activity relating to a particular service running on the managed device. Steel-Belted Radius Carrier runs an SNMP subagent that communicates with the **jnpnsnmpd** agent transparently; you do not need to register or configure the Steel-Belted Radius Carrier subagent to work with the SNMP agent.

- A network management station (NMS) is an administration workstation that polls management agents for information and provides control information for agents. A network management station can also accept trap messages when an asynchronous event occurs on a managed device.

Figure 16 on page 553 illustrates the default SNMP management architecture.

Figure 16: SNMP Architecture



SNMP Versions

Steel-Belted Radius Carrier supports SNMP versions 1 (SNMPv1) and 2c (SNMPv2c).

- SNMPv1 is the original implementation of SNMP, as defined in RFC 1157, "Simple Network Management Protocol (SNMP)."
- SNMPv2c is an enhanced version of the SNMP standard that includes improvements to SNMPv1 in the areas of protocol packet types, transport mappings, and MIB structure elements. SNMPv2c uses the SNMPv1 "community based" administration structure and RFC 1901, "Introduction to Community-based SNMPv2;" RFC 1905, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2);" and RFC 1906, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)."

NOTE: Steel-Belted Radius Carrier does not support SNMP version 3 (SNMPv3).

MIBs Overview

A management information base (MIB) is a hierarchical collection of information that resides on a managed device. A MIB defines the types of information (objects) that can be controlled and collected by an NMS and includes thresholds, counters, tables, lists, and values. Managed objects consist of one or more object instances.

MIB objects can be read-only or read-write:

- A read-only object is a variable that can be read but not set from an NMS. For example, an NMS can read (but not increment) the value of a counter showing the number of packets received on the accounting port.
- A read-write object is a variable that can be set from an NMS. For example, an NMS can set the device name or IP address for an SNMP client.

For the list of MIBs supported by Steel-Belted Radius Carrier, see the *SBR Carrier Administration and Configuration Guide*.

SNMP Messages

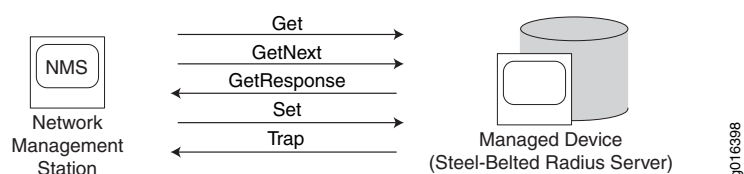
SNMP uses different types of messages to send and retrieve information.

- A Get message requests the value of an object from a table or list maintained by a managed device. For example, a Get message might request the number of users since a device was restarted or the number of authentication requests that a server has received.
- A GetNext message requests the value of the next object instance from a table or list maintained by a managed device. GetNext messages enable the NMS to walk a list or table to retrieve MIB object values sequentially.
- A Get-Response message returns the information requested by a Get or GetNext message.
- A Set message sets the value of an object instance within a managed device. Sets are not supported in Steel-Belted Radius Carrier.
- A Trap message notifies the NMS asynchronously when an important event, such as a change in state or a device or component failure, has occurred. For example, a managed device might send a trap message if the amount of space on the RADIUS server falls below a specified threshold or if the server cannot access its authentication database.

The SNMP traps supported by Steel-Belted Radius Carrier are **fnkradtr.mib** (for SNMP v1) and **fnkradtr-v2** (for SNMP v2). Traps are divided into three types:

- *Informational* traps are sent to report important RADIUS information that is not an error or a warning, such as when the RADIUS server daemon is loaded or unloaded, when a threshold of some kind has been exceeded, or when the system has recovered from a previous error or warning condition.
- *Warning* traps are sent to report RADIUS behavior that indicates a problem has occurred or may occur, such as when the RADIUS server is unable to connect to an external SQL database or when the file system is almost full. Many of these warning traps can be diluted or have configurable thresholds.
- *Error* traps are sent to report RADIUS problems that have occurred, such as when the RADIUS server is unable to initialize one or more critical components on startup. Most Error traps indicate that the RADIUS server failed to start properly for some reason, such as the inability to allocate memory from the system. Most of these traps cannot be diluted.

Figure 17: SNMP Messages



Dilution and Threshold

Trap event dilution means you can configure Steel-Belted Radius Carrier so that a particular trap is sent to the NMS once for every n occurrences of the condition that generated that trap. This allows for a fine degree of control with respect to trap generation for certain warning and error conditions. Many of the traps defined in **fnkradtr.mib** and **jnx-diameter-base-protocol.mib** can be diluted.

Some traps have configurable thresholds so you can set lower and upper limits of acceptable behavior, and to generate different types of traps depending on the condition. For example, you can configure SBR Carrier to send a warning trap message when if the count of available threads (for authentication and accounting) falls below 10, and to send an informational trap message when the count of available threads rises above 20.

SNMP trap event dilutions and thresholds are configured in the **events.ini** and **events.xml** files, which reside in the RADIUS server directory. If you anticipate using SNMP traps, review **fnkradtr.mib**, **jnx-diameter-base-protocol.mib**, **events.ini**, and **events.xml** files to understand the options available to you. For more information about the **events.ini** and **events.xml** files, see [“events.ini File” on page 18](#) and [“events.xml File” on page 22](#).

SNMP Community Overview

An SNMP community defines an administrative relationship between a managed device and one or more management stations on your network. Each community has a name called the *community string*. The

community string provides access control for SNMP objects. When an NMS sends a Get message or Set message to a managed device that belongs to an SNMP community, it includes the appropriate community string in the request.

- If the community string in the request is correct, the managed device sends back the requested information.
- If the community string is incorrect, the managed device discards the request without responding.

Rate Statistic Overview

Rate statistics variables defined in the **fnkrate.mib** file are derived from existing counter statistics by taking time into consideration.

The **funkSbrRatesSecondsPerInterval** read-only variable gives the duration in seconds of the interval over which the rate statistics are gathered.

For overall server specific rate statistics, three types of rate values are calculated for each of these counter statistics:

- Current-rate—The rate measured over the most recent rate interval
- Average-rate—The rate measured since startup, or the most recent statistics reset command
- Peak-rate—The highest current rate observed either since startup (not the highest average rate), or the most recent statistics reset command

For rate statistics per NAD client and per Called-Station-ID, only the current-rate and peak-rate values are calculated for each of the counter statistics.

NOTE: NAD client and Called-Station-ID specific rate statistics are calculated only if you set the **EnhancedRateStats** parameter in the **radius.ini** file to 1.

jnbrsnmpd.conf File Overview

The **jnbrsnmpd.conf** configuration file stores settings for the SNMP agent. After you install the SNMP agent for Steel-Belted Radius Carrier on a server, you can modify the **jnbrsnmpd.conf** configuration file to reflect your network environment.

NOTE: When you install Steel-Belted Radius Carrier, you are prompted to enter your SNMP settings by the installation script. The installation script updates the **jnp_{rs}snmpd.conf** file based on the values you enter.

If SNMP was not set up during installation, we recommend you run the configuration script again and enable SNMP. Take care not to make any changes to the existing configuration.

The "clientaddr" needs to be placed under [snmp] in the **jnp_{rs}snmpd.conf** file; otherwise, an error is logged in the **jnp_{rs}snmpd.log**. To avoid this issue, place "clientaddr" under the [snmp] header in the **jnp_{rs}snmpd.conf** file.

Access Control Overview and Syntax

jnp_{rs}snmpd supports the View-Based Access Control Model (VACM) described in RFC 2575. To configure access control, you must map community names to security names, map security names to groups, and specify access rights and views for groups.

Use the **com2sec** keyword to map each source/community pair to a security name. The **com2sec** entry is used to determine a security name from the traditional community string, taking into account where a request has come from.

The syntax for the **com2sec** keyword is:

```
com2sec security_name source community
```

where:

- *security_name* identifies the security name you want to create.
- *source* can be a hostname, a subnet, or the word default. You can specify a subnet as an IP address and mask (*nnn.nnn.nnn.nnn/nnn.nnn.nnn.nnn*) or as an IP address and Classless Inter-Domain Routing (CIDR) bits (*nnn.nnn.nnn.nnn/nn*).
- *community* is an SNMP community string, which acts as a password to authenticate SNMP communications.

NOTE: If you use a CIDR address to identify a subnet, the host portion of the CIDR address must be 0. For example, if you are using the equivalent of a Class C subnet such as 192.168.1.x, you must enter the network address as 192.168.1.0/24 (which is the equivalent of 192.168.1.0/255.255.255.0).

The first source/community combination that matches an incoming packet is selected.

For example, the following creates two security names (**local** and **mynetwork**) and maps them to two different subnet/community name pairs:

	sec.name	source	community
com2sec	local	localhost	local_community
com2sec	mynetwork	192.168.10/24	remote_community

Security Names Overview and Syntax

Use the group keyword to map security names into group names. The **group** keyword gives general control by mapping between a security name (for a particular protocol version), and the internal name used in the access line.

The syntax for the group keyword is:

```
group name model security
```

where:

- *name* is the name of an access group
- *model* identifies the security model you want to use: v1 or v2c.
- *security* is a security name.

For example, these lines map the two security names to four group/model pairs:

#	sec.model	sec.name	
group	LocalGroup	v1	local
group	LocalGroup	v2c	local
group	LANGroup	v1	mynetwork
group	LANGroup	v2c	mynetwork

Access View Overview and Syntax

Use the **view** keyword to specify what portions of the MIB tree a specified group can view or modify. The syntax for the **view** keyword is:

```
view name {include | excluded} subtree [mask]
```

where:

- *name* is the identifier used for the view.
- include | exclude lets you include or exclude specific portions of the MIB tree from the view.
- *subtree* identifies the portion of the MIB tree that this name refers to in numeric or named form.
- *mask* specifies what elements of the MIB subtree are relevant. The mask argument enables you to control access to specific rows in a table. When the entire MIB can be viewed, you can omit the mask field or enter **ff**.

Group Access Overview and Syntax

Use the **access** keyword to specify who has access to part or all of the MIB tree. The syntax for the **access** keyword is:

```
access name context model level prefix read write notify
```

where:

- *name* is the name of a group.
- *context* specifies the context for the view. For SNMPv1 or SNMPv2c, *context* is empty.
- *model* is the security model: any, v1, or v2c.
- *level* can be used to ensure that the request is authenticated or encrypted. For SNMPv1 or SNMPv2c, *level* is noauth.
- *prefix* specifies how the *context* setting is matched against the context of the incoming PDU. Enter exact or prefix.
- *read* specifies the view to be used for READ access.
- *write* specifies the view to be used for WRITE access.
- *notify* specifies the view to be used for NOTIFY access.

For example, these settings specify that the LocalGroup uses the all view for READ, WRITE, and NOTIFY access:

#			sec	sec				
#		context	model	level	prefix	read	write	notify
access	LocalGroup	""	any	noauth	exact	all	all	all
access	LANGroup	""	any	noauth	exact	all	none	none

System Contact Overview and Syntax

You can specify your system contact information in the **jnpnsnmpd.conf** file or in the MIB. If you configure your system contact information in the **jnpnsnmpd.conf** file, the objects are locked and cannot be modified by means of SNMP.

System contact information consists of the following:

- **syslocation**—The physical location of the managed device.
- **syscontact**—The person or department responsible for maintaining the managed device.
- **sysname**—The name of the managed device.

This information is stored in the system group of the MIB-II tree.

The syntax for specifying system contact information is:

```
syslocation string
syscontact string
sysname string
```

Traps Overview and Syntax

Traps can be used by network entities to signal abnormal conditions to management stations. Identify the NMS that receives trap messages generated by Steel-Belted Radius Carrier.

NOTE: You can configure Steel-Belted Radius Carrier to use either SNMPv1 or SNMPv2c traps. You cannot configure Steel-Belted Radius Carrier to generate both types of traps simultaneously.

- Use the **trapcommunity** keyword to specify the default community string to be used when sending traps. Syntax for the **trapcommunity** keyword is:


```
trapcommunity string
```

The **trapcommunity** keyword must precede the **trap2sink** keyword in the **jnprsnmpd.conf** file.

- Use the **trapsink** and **trap2sink** keywords to specify whether you want Steel-Belted Radius Carrier to use either SNMPv1 traps or SNMPv2c traps. Do not enable both types of traps at the same time.
- The **trapsink** keyword specifies the host or hosts to which the Steel-Belted Radius Carrier server is to send SNMPv1 trap messages. If you want to use SNMPv1 traps, uncomment the **trapsink** keyword and specify the host or hosts to which you want Steel-Belted Radius Carrier to send SNMPv1 trap messages.

Syntax for the **trapsink** keyword is:

```
trapsink host[:port] [community]
```

where:

- *host* specifies the hostname or IP address of the NMS.
- *community* specifies the community string the NMS accepts.
- *port* specifies the UDP port on which the NMS is listening for SNMPv1 trap messages. Default is UDP port 162.

For example:

```
# send v1 traps
trapsink nms.system.com secret
```

- The **trap2sink** keyword specifies the host or hosts to which you want Steel-Belted Radius Carrier server to send SNMPv2c trap (notification) messages. If you want to use SNMPv2c traps, uncomment the **trap2sink** keyword to specify the host or hosts to which you want Steel-Belted Radius Carrier to send SNMPv2c trap (notification) messages.

The syntax for the **trap2sink** keyword is:

```
trap2sink host[:port] [community]
```

where:

- *host* specifies the hostname or IP address of the NMS.
- *community* specifies the community string the NMS accepts.
- *port* specifies the port on which the NMS is listening for SNMPv2c trap messages.

For example:

```
# send v2 traps
trap2sink nms.system.com secret
```

[snmp] Overview and Syntax

The SNMP agent uses the **jnbrsnmpd.conf** file to store static agent configuration information, such as community strings. The SNMP agent uses the **persist** directory to store information set during the running of the agent, which needs to be persistent from one run to the next.

The **persistDir** keyword in the [snmp] section of **jnbrsnmpd.conf** specifies the location of the **persist** directory. By default, the **persist** directory is located in the **/snmp** directory within *radiusdir* on your server.

The syntax for specifying the location of the **persist** directory is:

```
[snmp]
persistDir radiusdir/sntp/persist
```

init.jnbrsnmpd Overview and Syntax

By default, **jnbrsnmpd** listens for incoming SNMP requests on UDP port 161 on all IP interfaces. You can specify a different UDP port in the **init.jnbrsnmpd** file. The syntax for specifying a listening port is:

```
SNMPDPORT=port_number
```

NOTE: If you change the SNMP port number in **init.jnbrsnmpd**, you must also enter the same port number in **testagent.sh**.

NOTE: If you run more than one SNMP agent on your server, each agent must use a unique UDP port number.

Subagent Overview and Syntax

By default, the SNMP subagent in Steel-Belted Radius Carrier communicates with the SNMP agent on TCP port 6669. If you change the port used for subagent-agent communication, you must modify

`jnprsnmpd.conf` to uncomment the `a3s_admin_parameters` keyword and specify host, port, and interval values.

The syntax for the `a3s_admin_parameters` keyword is:

```
a3s_admin_parameters host=localhost port=port tryinterval=interval
```

where:

- *port* identifies the TCP port the Steel-Belted Radius Carrier server uses for SNMP communication. The default value is 6669.
- *interval* specifies the number of seconds the subagent waits to reconnect to the server if the connection breaks. If your SNMP management station issues queries intermittently, set the **tryinterval** value to a small number (1-5) to ensure timely information. If your SNMP management station polls the server periodically, set the **tryinterval** value to a larger number to avoid flooding the server with queries. The default is 10 seconds.
- *lifetime* specifies the number of seconds information can remain in the SNMP subagent cache. If your SNMP management station issues queries intermittently, set the **tryinterval** value to a small number (1-5) to ensure timely information. If your SNMP management station polls the server periodically, set the **tryinterval** value to a larger number to avoid flooding the server with queries. The default is 10 seconds.

testagent.sh Script Overview and Syntax

You can run the `testagent.sh` script to verify that the `jnprsnmpd` SNMP agent is functioning. Before you do so, you must configure the `testagent.sh` file with the community string for your network.

The syntax for the `testagent.sh` file ([Table 197 on page 563](#)) is:

```
snmpget_path -M mib_location -c community port sysDescr
```

Table 197: testagent.sh Syntax

Keyword	Function
<code>snmpget_path</code>	Specifies the path for the <code>snmpget</code> utility. Default value is <code>radiusdir/snmp/bin/snmpget</code> .
<code>-M mib_location</code>	Specifies the path for the MIBs used by the SNMP agent. Default value is <code>/radiusdir/snmp/mibs</code> .

Table 197: testagent.sh Syntax (*continued*)

Keyword	Function
<i>-c community</i>	Specifies the community string for your network. Default value is COMMUNITY .
<i>port</i>	Specifies the default port for SNMP traffic. Default value is localhost:161 .
<i>sysDescr</i>	Specifies the MIB variable to be retrieved. Default value is system.sysDescr.0 .

SNMP Traps and Statistics Overview

IN THIS CHAPTER

- Trap Variables Overview | 566
- Trap Definitions | 569
- Rate Statistics | 587

Steel-Belted Radius Carrier supports retrieving configuration information with standard SNMP utilities. This chapter summarizes the proprietary SBR Carrier MIBs, each one's traps, and rate statistics generated by SBR Carrier.

The seven proprietary MIBs are:

- The **fnkradtr.mib** defines the content of the traps that are generated by the Steel-Belted Radius Carrier server for SNMPv1.
- The **fnkradtr-v2.mib** defines the content of the traps that are generated by the Steel-Belted Radius Carrier server for SNMPv2.
- The **fnkrate.mib** defines the peak, current, and average rate statistics maintained by Steel-Belted Radius Carrier.
- The **jnx-smi.mib** defines Juniper Networks overall MIB hierarchy.
- The **jnx-aaa.mib** defines Juniper Networks AAA specific MIB hierarchy.
- The **jnx-diameter-base-protocol.mib** defines the content of Diameter traps that are generated by the SBR Carrier server.
- The **jnx-diameter-nas-application.mib** maintains Diameter NASREQ application specific counters.

Reporting characteristics of a number of traps in the proprietary MIBs can be altered by diluting or suppressing them or altering reporting thresholds. These settings are controlled by the *radiusdir/events.ini* file and the *events.xml* file. For more information about these files, see [“events.ini File” on page 18](#) and [“events.xml File” on page 22](#).

NOTE: The SNMP subagent in Steel-Belted Radius Carrier may generate traps that do not reference Juniper Networks (Funk) enterprise IDs. For information about generic SNMP traps specified by IETF-specified MIBs, refer to the appropriate RFC. For information about generic netSnmp traps specified by netSnmp-specific MIBs, refer to the netSnmp documentation at www.net-snmp.org.

These topics are included in this chapter:

Trap Variables Overview

Table 198 on page 566 lists the trap variables for the proprietary SNMP traps used by Steel-Belted Radius Carrier.

Table 198: Trap Variables

Variable Name	Identifies
funkSbrTrapVarComp	<p>The component within the SBR server that issued the trap.</p> <ul style="list-style-type: none"> • 1—Core • 2—Accounting • 3—Authentication
funkSbrTrapVarSev	<p>The severity of the event that caused the trap.</p> <ul style="list-style-type: none"> • 1—Informational • 2—Warning • 3—Error
funkSbrTrapVarCurrentSessions	<p>Total number of concurrent active sessions in the server.</p>
funkSbrTrapVarLicensedSessions	<p>The maximum number of concurrent active sessions allowed by the license.</p>
funkSbrTrapVarSWName	<p>The identity of the software that is the RADIUS server.</p>
funkSbrTrapVarThreadsAvail	<p>The number of threads available in the thread worker pool.</p>

Table 198: Trap Variables (*continued*)

Variable Name	Identifies
funkSbrTrapVarBytesAvail	The number of bytes available in the file system.
funkSbrTrapVarPrivateDir	The file system path to the private directory used by the RADIUS server.
funkSbrTrapVarNumberOfOccurrences	The dilution factor for the trap. The trap is sent on once for every 'n' occurrences of this event.
funkSbrTrapVarSQLConnects	The number of connection attempts to a SQL database.
funkSbrTrapVarSQLDisconnects	The number of disconnects from a SQL database (due to an error encountered during an operation).
funkSbrTrapVarSQLTimeouts	The number of timeouts encountered when trying to perform a transaction against a SQL database.
funkSbrTrapVarIniString	The .ini file setting used to specify a configuration value.
funkSbrTrapVarDbType	The type of database being employed by the RADIUS server.
funkSbrTrapVarFailedSystemName	The name of the remote system failing connectivity from the RADIUS server.
funkSbrTrapVarUserName	The name of the user to whom the trap refers.
funkSbrTrapVarPersistStoreName	The name of the persistent storage to which the trap refers.
funkSbrTrapVarDiagnosticMessage	A generic diagnostic message that may be helpful in determining and addressing the possible root causes of the trap.
funkSbrTrapVarIPAddrPoolName	The name of the IP address pool to which the trap refers.

Table 198: Trap Variables (*continued*)

Variable Name	Identifies
funkSbrTrapVarIPAddrAvail	The number of addresses available in the IP address pool.
funkSbrTrapVarConnectedSystemName	The name of the remote system with which the RADIUS server has established a connection.
funkSbrTrapVarQueueName	The name of a queue in the RADIUS server.
jnxAAATrapVarSeverity	The severity of the event that caused the trap: <ul style="list-style-type: none"> • 1—Informational • 2—Warning • 3—Error
jnxAAATrapVarNumberOfOccurrences	The dilution factor for the trap. The trap is sent once for every 'n' occurrences of this event.
jnxAAATrapVarAdminUserName	The name of the admin user whose request is denied.
jnxAAATrapVarAdminRequestUri	The uri of the request that is denied.
jnxAAATrapVarAdminRequestErrorReason	The textual reason why an admin request is denied.
funkSbrTrapVarPortNumber	The socket port number to which the trap refers.
funkSbrTrapVarIPAddr	The socket IP address to which the trap refers.
funkSbrTrapVarLogDir	The name of the directory where log files are stored.
funkSbrTrapVarTransactionRateLimit	The textual information that specifies whether the transaction rate limit is applied or cleared.

Table 198: Trap Variables (*continued*)

Variable Name	Identifies
funkSbrTrapVarCSTSwitchOverWhat	<p>The store from which the persistence switchover was made:</p> <ul style="list-style-type: none"> • from NDB to LOCAL—The switchover happened from SSR cluster to local CST. • from LOCAL to NDB—The switchover happened from local CST to SSR cluster.
funkSbrTrapVarCSTSwitchOverWhen	<p>The time when the persistence switchover occurred:</p> <ul style="list-style-type: none"> • during start-up—The switchover occurred during SBR startup. • during transaction—The switchover occurred during SBR transaction.
jnxDbpPeerId	The server identifier of the Diameter peer.
jnxAAATrapVarResultCode	Diameter Result-Code attribute values returned by SBR Carrier.

Trap Definitions

Table 199 on page 570 lists proprietary SNMP traps generated by Steel-Belted Radius Carrier. The columns in Table 199 on page 570 consist of the following:

- **OID Suffix**—Identifies the OID suffix for the trap. To identify the OID number for an alarm, append the OID suffix to the Juniper Networks (Funk) OID prefix (1.3.6.1.4.1.1411). For example, the ASN.1 number for the **funkSbrTrapServiceStarted** trap is 1.3.6.1.4.1.1411.1.1.0.100.
- **Trap**—Identifies the name of the proprietary trap.
- **Description**—Describes when the trap is generated.
- **Type**—Indicates whether the trap is informational, warning, or error.

NOTE: Some Steel-Belted Radius Carrier traps may be diluted so that one message is generated when some number of events occur. For example, when five events occur, one message can be sent instead of five discrete messages, so frequently occurring traps do not overwhelm the network with unnecessary messages.

NOTE: Traps for Service Level Manager (SLM) client events are Early Field Trial features and are not yet fully qualified. If you use any of these, it is your responsibility to ensure that the feature operates correctly in your targeted configuration. These traps are marked with an asterisk (*) in the OID Suffix column.

Table 199: fnkradtr.mib Trap Definitions

OID Suffix	Trap Name	Description	Type
100	funkSbrTrapServiceStarted	<p>Sent when the RADIUS server is started.</p> <p>Cause: Trap indicates that the server itself has started. This does not mean that all of the various configured features have loaded successfully.</p> <p>If there is an issue with another component, traps specific to it indicate so. This trap shows that a valid license is installed. You can now interact with the server through the Web GUI or LDAP Configuration Interface.</p> <p>Severity: If unexpected, this can be the result of the radius process crashing, which produces a core dump file.</p>	Info
101	funkSbrTrapServiceStopped	<p>Sent when the RADIUS server is stopped.</p> <p>Cause: The server completed its shutdown operation and is no longer running. Server does not respond to any operation from the Web GUI, LCI, or from any inbound RADIUS data.</p> <p>Severity: If unexpected, this can be the result of a core in the RADIUS process.</p>	Info

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
102	funkSbrTrapThreadsNormal	<p>Sent when the number of available threads in the accounting or authentication server has risen above a specified threshold.</p> <p>You can set this threshold value in the [Thresholds] section of the events.ini file.</p> <p>Cause: The number of available threads on the system has risen above the threshold configured in the events.ini file.</p>	Info
103	funkSbrTrapFSNormal	<p>Sent when the number of bytes available in the file system from which the server is running has risen above a specified threshold.</p> <p>You can set this threshold value in the [Thresholds] section of the events.ini file.</p> <p>Cause: The number of bytes available in the free disk space has increased above the threshold configured in the events.ini file.</p> <p>Severity: This can cause the system to become inoperable.</p>	Info
104 *	funkSbrTrapConcurrencyReconnect	<p>Sent when RADIUS reconnects to the Service Level Manager server after it has sent a ConcurrencyFailure, ConcurrencyTimeout, or ConcurrencyLocalProxyFailure trap.</p> <p>Cause: If a failure to communicate with the Concurrency Server has occurred, this trap is sent when communications have been re-established and the SLM server is responding again.</p> <p>Severity: Users may have either been rejected or been able to exceed their configured concurrent login policy during the time interval when communications with the CS were down. This depends on the settings in the forward.aut configuration file which resides on any of the SBR servers acting as clients to the CS. Check the RejectIfUnreachable setting if you are uncertain as to the expected behavior of Steel-Belted Radius Carrier in the event that the CS is unreachable.</p>	Info

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
105	funkSbrTrapSQLReconnect	<p>Sent when RADIUS reconnects to the SQL database after it has sent a SQLConnectFail trap.</p> <p>Cause: If a failure to communicate with the SQL server database has occurred, this trap is sent when communications have been re-established and the SQL server is responding again.</p> <p>Severity: Users may have either been rejected or been allowed onto the network without proper verification of credentials during the time interval when the SQL server was unreachable. This depends on the settings in the radsql.aut configuration file. Check the [Failure] section settings if you are uncertain as to the expected behavior of Steel-Belted Radius Carrier in the event that the SQL server is unreachable.</p>	Info
106	funkSbrTrapLDAPReconnect	<p>Sent when RADIUS reconnects to the LDAP server after it has sent a LDAPConnectFail trap.</p> <p>Cause: If a failure to communicate with the LDAP database has occurred, this trap is sent when communications have been re-established and the LDAP server is responding again.</p> <p>Severity: Users may have either been rejected or been allowed onto the network without proper verification of credentials during the time interval when the LDAP server was unreachable. This depends on the settings in the ldapauth.aut configuration file. Check the [Failure] section settings if you are uncertain as to the expected behavior of SBR in the event that the LDAP database is unreachable.</p>	Info

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
107	funkSbrTrapUserAccountLocked	<p>Sent when a user's account becomes locked out due to an excessive number of rejected authentication attempts within a defined period of time.</p> <p>Cause: A user's account is locked, disallowing access to the network, after an excessive number of rejected authentication attempts. This functionality is configured in the lockout.ini file.</p> <p>Severity: The user is not able to access the network until the account is unlocked.</p>	Info
108	funkSbrTrapUserAccountReleased	<p>Sent when a user's account, previously locked due to an excessive number of rejected authentication attempts, becomes unlocked.</p>	Info
109	funkSbrTrapProxySpoolReconnect	<p>Sent when the proxy accounting spooler reconnects to the target realm after it has sent a ProxySpoolTimeout trap.</p> <p>Cause: Issues affecting transmission of spooled accounting proxy data to one or more configured downstream targets have been resolved (possibly a restoration of the network link, or the downstream proxy accounting server has become available again).</p> <p>Severity: If unexpected, then the accounting target system (possibly a billing server) was not receiving data from the AAA server for some time interval. During that time, data was written to the local disk for temporary storage until the accounting target became available again.</p>	Info
110	funkSbrTrapIPAddrPoolNormal	<p>Sent when the number of available IP addresses in any pool rises above a specified threshold. IP pool thresholds can be configured in events.ini file.</p> <p>Severity: Users may have been rejected if threshold warning trap 5027 was ignored.</p>	Info
111	funkSbrTrapSQLConnect	<p>Sent only once, when RADIUS initially connects to the SQL database.</p>	Info

Table 199: fnkradtr.mib Trap Definitions (continued)

OID Suffix	Trap Name	Description	Type
112	funkSbrTrapLDAPConnect	Sent only once, when RADIUS initially connects to the LDAP server.	Info
113	funkSbrTrapWatchdogStarted	Sent when the radiusd watchdog is started.	Info
114	funkSbrTrapWatchdogStopped	Sent when the radiusd watchdog is stopped.	Info
115	funkSbrTrapWatchdogRadiusStarted	Sent whenever the radiusd watchdog attempts to start or restart the RADIUS server.	Info
116	funkSbrTrapUserAccountRedirected	Sent when a user account has been redirected due to an excessive number of rejected authentication attempts.	Info
117	funkSbrTrapSS7CommunicationOK	Sent when SS7 communications are successful after a funkSbrTrapSS7CommunicationError trap has been sent.	Info
118	funkSbrTrapSS7CDRGenerationOK	Sent when CDR generation is successfully written.	Info
119	funkSbrTrapSS7AuthDatabaseOK	Sent when access to the Authorization databases are successful after a funkSbrTrapSS7AuthDatabaseError trap has been sent.	Info
120	funkSbrTrapSS7ProvDatabaseOK	Sent when access to the SMS Provisioning database is successful after a funkSbrTrapSS7ProvDatabaseError trap has been sent.	Info
121	funkSbrTrapProxyFastFailOK	Sent when a proxy target has come out of fast-fail mode and responded after a funkSbrTrapProxyFastFail trap was sent.	Info
122	funkSbrTrapProxyOutOfServiceOK	Sent when at least one target in a proxy realm has responded, come out of fast-fail mode, and returned into service after a funkSbrRTrapProxyOutOfService trap was sent.	Info
123	funkSbrTrapSSRCommunicationOK	Sent when SBRC is reconnected to SSR/CST.	Info

Table 199: fnkradtr.mib Trap Definitions (continued)

OID Suffix	Trap Name	Description	Type
124	funkSbrTrapResumeLogging	Sent when the logging mechanism resumes logging after the amount of free space in the logging partition exceeds the threshold limit.	Info
128	funkSbrTrapCstSwitchOver	Sent when the session store switches from SSR cluster to local CST, or vice versa.	Info
5000	funkSbrTrapCmdArgBadPrivDir	Sent when an invalid private directory is specified on the command line used to launch the RADIUS server. The command line option is ignored.	Warning
5001	funkSbrTrapLowThreads	Sent when the count of threads available for the accounting or authentication server drops below a configurable threshold. An informational trap is sent when the count of available threads (at some future point) rises to an acceptable level.	Warning
5002 *	funkSbrTrapConcurrencyFailure	Sent when communications with the RADIUS concurrency server fails. Trap can be diluted.	Warning
5003 *	funkSbrTrapConcurrencyTimeout	Sent when communications with the RADIUS concurrency server times out. Trap can be diluted.	Warning
5004 *	funkSbrTrapConcurrencyLocalProxy Failure	Sent when a local error prevents the RADIUS server from sending a proxy request to the RADIUS concurrency server. Trap can be diluted.	Warning
5005	funkSbrTrapStaticAcctProxyTimeout	Sent when the RADIUS server times out in an attempt to forward an accounting request to the location specified by the static proxy option. Trap can be diluted.	Warning

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
5006	funkSbrTrapStaticAcctProxyLocal Failure	Sent when the RADIUS server encounters a local failure in an attempt to forward an accounting request to the location specified by the static proxy option. Trap can be diluted.	Warning
5007	funkSbrTrapLowFSSpace	Sent when the amount of space available in the file system in which the server's private directory resides falls below a configurable threshold. An informational trap is sent when the amount of available space (at some future point) rises to an acceptable level.	Warning
5008	funkSbrTrapSQLConnectFail	Sent when the connection to a SQL database has failed. Trap can be diluted.	Warning
5009	funkSbrTrapSQLDisconnect	Sent when a disconnect to a SQL database occurs. Trap can be diluted.	Warning
5010	funkSbrTrapSQLTimeout	Sent when a timeout occurs during an attempt to perform transactions to a SQL database. Trap can be diluted.	Warning
5015	funkSbrTrapLDAPConnectFailure	Sent when a connect failure to an LDAP server occurs.	Warning
5016	funkSbrTrapLDAPConnectFailures	Sent when an attempt to communicate with the LDAP Server has failed. Trap can be diluted.	Warning
5017	funkSbrTrapLDAPDisconnects	Sent when the LDAP Server has disconnected. Trap can be diluted.	Warning
5018	funkSbrTrapLDAPRequestTimeouts	Sent when a request sent to the LDAP Server has timed out. Trap can be diluted.	Warning
5019	funkSbrTrapLDAPDisconnect	Sent when a disconnect to a LDAP server occurs.	Warning

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
5020	funkSbrTrapLDAPRequestTimeout	Sent when a request sent to the LDAP Server has timed out.	Warning
5021	funkSbrTrapProxySpoolTimeout	Sent when a request forwarded by the proxy accounting spooler has timed out. This trap is paired with 109 so that the name of the realm that has failed is reported.	Warning
5022	funkSbrTrapProxySpoolTimeouts	Sent when a request forwarded by the proxy accounting spooler has timed out. Reports the number of occurrences of failure. Trap can be diluted.	Warning
5023 *	funkSbrTrapSoftLimitViolation	Sent when accepting a concurrency request exceeds a realm's soft limit. Trap can be diluted.	Warning
5024 *	funkSbrTrapHardLimitViolation	Sent when a concurrency request is rejected because a realm's hard limit has been reached. Trap can be diluted.	Warning
5025 *	funkSbrTrapConcurrencyServer Misconfiguration	Sent when a PAS realm has been misconfigured. All authentication requests to the named realm are rejected.	Warning
5026	funkSbrTrapACCTWriteFailure	Sent when the server is unable to commit accounting data to a persistent store such as the file system, database, and so on. Trap can be diluted.	Warning
5027	funkSbrTrapIPAddrPoolLow	Sent when the number of available IP addresses in any pool falls below a configurable threshold. An informational trap is sent when the number of available IP addresses (at some future point) rises to an acceptable level.	Warning
5028	funkSbrTrapWatchdogRadiusTerm	Sent whenever the radiusd watchdog attempts to send a TERM signal to terminate the RADIUS server.	Warning

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
5029	funkSbrTrapWatchdogRadiusKill	Sent whenever the radiusd watchdog attempts to send a KILL signal to terminate the RADIUS server.	Warning
5030	funkSbrTrapFloodQueueOverflow	Sent whenever a flood queue drops a packet. Trap can be diluted.	Warning
5033 *	funkSbrTrapMappingFailure	Sent when rejecting a concurrency request that fails to resolve a realm or region name Trap can be diluted.	Warning
5035	funkSbrTrapProxyFastFail	Sent when a proxy target does not respond and enters fast-fail mode.	Warning
5036	funkSbrTrapRealmOutOfService	Sent when all the targets for a proxy realm go out of service.	Warning
5037	funkSbrTrapProxySpoolRecordSkipped	Sent when all targets within a proxy realm do not respond and enter into fast-fail mode. Trap is issued only when moving to next record after exceeding the configured retry value.	Warning
5038	funkSbrTrapProxyTransactionLimitSet	Sent when the transaction rate limit is applied by transaction-based licensing.	Warning
5039	funkSbrTrapProxyTransactionLimitCleared	Sent when the transaction rate limit is cleared by transaction-based licensing.	Warning
5040	funkSbrTrapSuspendLogging	Sent when the logging mechanism is temporarily suspended when the amount of free space in the logging partition is below the threshold limit.	Warning
10003	funkSbrTrapFailedThreadCreate	Sent when an attempt to create a thread at server startup encounters a failure. The server fails to start.	Error
10004	funkSbrTrapFailedMutexCreate	Sent when an attempt to create a mutual exclusion lock (mutex) at server startup encounters a failure. A mutex prevents multiple threads from executing critical sections of code simultaneously. The server fails to start.	Error

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10005	funkSbrTrapFailedSignalInit	Sent when an attempt to initialize signal handling at server startup encounters a failure. The server fails to start.	Error
10007	funkSbrTrapFailedLogFile	Sent when an attempt to open or create a log file at server startup encounters a failure. The server fails to start.	Error
10008	funkSbrTrapFailedLDAPAdminInit	Sent when an attempt to initialize the LDAP administration interface at server startup encounters a failure. The server fails to start.	Error
10010	funkSbrTrapFailedIPInit	Sent when an attempt to initialize basic TCP/IP services at server startup encounters a failure. The server fails to start.	Error
10011	funkSbrTrapFailedCurrentSessionsInit	Sent when an attempt to initialize current sessions table processing at server startup encounters a failure. The server fails to start.	Error
10012	funkSbrTrapFailedChallCacheInit	Sent when an attempt to initialize the RADIUS challenge continuation cache at server startup encounters a failure. The server fails to start.	Error
10013	funkSbrTrapFailedActiveRASInit	Sent when an attempt to initialize the network access server activity monitor at server startup encounters a failure. The server fails to start.	Error
10014	funkSbrTrapFailedDictionaryInit	Sent when an attempt to initialize the dictionary processing at server startup encounters a failure. The server fails to start.	Error
10015	funkSbrTrapFailedVendorInit	Sent when an attempt to process the vendor.ini file at server startup encounters a failure. The server fails to start.	Error
10016	funkSbrTrapFailedDBInit	Sent when an attempt to initialize the internal database at server startup encounters a failure. The server fails to start.	Error

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10017	funkSbrTrapFailedUnixUserInit	Sent when an attempt to initialize the UNIX user browsing component at server startup encounters a failure. The server fails to start.	Error
10018	funkSbrTrapFailedAdminRightsInit	Sent when an attempt to initialize the administration user rights component at server startup encounters a failure. The server fails to start.	Error
10019	funkSbrTrapFailedDbOpen	Sent when an attempt to open the internal database at server startup encounters a failure. The server fails to start.	Error
10020	funkSbrTrapFailedDNISLookupInit	Sent when an attempt to initialize the tunnel DNIS lookup component at server startup encounters a failure. The server fails to start.	Error
10021	funkSbrTrapFailedConfigCacheInit	Sent when an attempt to initialize the configuration caching component at server startup encounters a failure. The server fails to start.	Error
10022	funkSbrTrapFailedDbCacheInit	Sent when an attempt to initialize the database caching component at server startup encounters a failure. The server fails to start.	Error
10023	funkSbrTrapFailedLicenseInit	Sent when an attempt to initialize the licensing component at server startup encounters a failure. The server fails to start.	Error
10025	funkSbrTrapFailedHostLookupInit	Sent when an attempt to initialize host lookup processing on NetWare at server startup encounters a failure. The server fails to start.	Error
10026	funkSbrTrapFailedAddrPoolInit	Sent when an attempt to initialize IP address pool resource management at server startup encounters a failure. The server fails to start.	Error
10027	funkSbrTrapFailedLoginLimitInit	Sent when an attempt to initialize user login count tracking at server startup encounters a failure. The server fails to start.	Error

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10028	funkSbrTrapFailedPersistStoreCreate	Sent when an attempt to create the persistent store for current session list processing at server startup encounters a failure. The server fails to start.	Error
10029	funkSbrTrapFailedPersistStoreInit	Sent when an attempt to initialize the persistent store for current session list processing at server startup encounters a failure. The server fails to start.	Error
10031	funkSbrTrapFailedLockInit	Sent when an attempt to initialize admin locking component at server startup encounters a failure. The server fails to start.	Error
10032	funkSbrTrapFailedPluginInit	Sent when an attempt to initialize the plug-in support component at server startup encounters a failure. The server fails to start.	Error
10033	funkSbrTrapFailedPacketCacheInit	Sent when an attempt to initialize duplicate packet request cache at server startup encounters a failure. The server fails to start.	Error
10034	funkSbrTrapFailedNameMangleInit	Sent when an attempt to initialize name mangling support at server startup encounters a failure. The server fails to start.	Error
10035	funkSbrTrapFailedNameStripInit	Sent when an attempt to initialize name stripping support at server startup encounters a failure. The server fails to start.	Error
10036	funkSbrTrapFailedFSSpaceChecking	Sent when an attempt to determine the amount of free file system space fails. File system space checking is disabled until the server is restarted.	Error
10037	funkSbrTrapFailedNameValidateInit	Sent when an attempt to initialize name validation support at server startup encounters a failure. The server fails to start.	Error
10038	funkSbrTrapFailedResourceCheckInit	Sent when an attempt to initialize system resource checking at server startup encounters a failure. The server fails to start.	Error

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10039	funkSbrTrapFailedSystemStatsInit	Sent when an attempt to initialize statistic collection at server startup encounters a failure. The server fails to start.	Error
10040	funkSbrTrapSQLConnectFailure	Sent when a connection attempt from the SQL authentication or accounting plug-in to the specified system has failed.	Error
10041	funkSbrTrapSQLDiscon	Sent when a disconnect from a SQL database has occurred.	Error
10042		(reserved for internal use) individual SQL timeout (not sent as a trap)	Error
10043	funkSbrTrapFailedReserveMemoryAlloc	Sent when an attempt to allocate reserved memory based on a setting in the radius.ini file fails. The server starts without reserved memory, but is unable to warn of low memory conditions.	Error
10044	funkSbrTrapReserveMemoryFreed	Sent when an attempt to allocate memory during runtime fails and the block of memory reserved at system startup is freed in an attempt to alleviate the low memory condition.	Error
10045	funkSbrTrapMemoryAllocFail	Sent when an attempt to allocate memory has failed. Trap can be diluted.	Error
10048	funkSbrTrapFailedMibInfoCollectInit	Sent when an attempt to initialize MIB information collection at server startup encounters a failure. The server fails to start.	Error
10049	funkSbrTrapFailedMibInfoAccessInit	Sent when an attempt to initialize MIB access at server startup encounters a failure. The server fails to start.	Error
10050	funkSbrTrapFailedCommonIPInit	Sent when an attempt to initialize common IP services at server startup encounters a failure. The server fails to start.	Error

Table 199: fnkradtr.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10051	funkSbrTrapWatchdogAborted	Sent whenever the radiusd watchdog terminates. This is typically due to a prolonged inability to control or communicate with the RADIUS server, or some fatal error that has occurred within the watchdog itself.	Error
10052	funkSbrTrapWatchdogFailedInit	Sent whenever the radiusd watchdog is unable to initialize itself. This is typically due to insufficient or invalid command line parameters given to the watchdog itself.	Error
10053	funkSbrTrapAdminAuthFailedInit	Sent whenever the server is unable to initialize administrative authentication and authorization. The server fails to start.	Error
10054	funkSbrTrapServiceFailedInit	Sent when the server has failed to start. This trap is sent in addition to a specific failure trap.	Error
10055	funkSbrTrapSS7MapGatewayFailedInit	Sent when the MAP Gateway has failed to initialize. The server starts but SS7 functions are not available.	Error
10056	funkSbrTrapSS7CommunicationError	Sent when Signalware communication has failed.	Error
10057	funkSbrTrapSS7CDRGenerationError	Sent once when Steel-Belted Radius Carrier fails to create CDR files.	Error
10058	funkSbrTrapSS7AuthDatabaseError	Sent once when Steel-Belted Radius Carrier fails to retrieve authorization information from any database.	Error
10059	funkSbrTrapSS7ProvDatabaseError	Sent when access to the Provisioning database has failed.	Error
10060	funkSbrTrapSSRCommunicationError	Sent when the connection to the SSR/CST has been lost.	Error
10061	funkSbrTrapMaxConcurrentSessionsExceeded	Trap is sent when the number of concurrent active sessions in Steel-Belted Radius Carrier exceeds the count set by the license.	Error

Table 200 on page 584 lists additional proprietary SNMP traps generated by Steel-Belted Radius Carrier for the jnx-aaa.mib.

The columns in Table 200 on page 584 consist of the following:

- **OID Suffix**—Identifies the OID suffix for the trap. To identify the OID number for an alarm, append the OID suffix to the Juniper Networks AAA mib OID prefix (1.3.6.1.4.1.2636.8.1). For example, the ASN.1 number for the **jnxAAATrapUnauthorizedAdminRequest** trap is 1.3.6.1.4.1.2636.8.1.0.5000.
- **Trap**—Identifies the name of the proprietary trap.
- **Description**—Describes when the trap is generated.
- **Type**—Indicates whether the trap is informational, warning, or error.

Table 200: jnx-aaa.mib Trap Definitions

OID Suffix	Trap Name	Description	Type
100	jnxAAATrapServerStartup	This trap is sent when the Diameter server is started. Parameters: <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences 	Info
101	jnxAAATrapServerShutdown	This trap is sent when the Diameter server is stopped. Parameters: <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences 	Info
5000	jnxAAATrapUnauthorizedAdminRequest	This trap is sent whenever a request from the administrator interface is denied. Parameters: <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarAdminUserName • jnxAAATrapVarAdminRequestUri • jnxAAATrapVarAdminRequestErrorReason This trap can be diluted.	Warning

Table 200: jnx-aaa.mib Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
10000	jnxAAATrapInternalError	<p>This trap is sent when the Diameter Result-Code attribute value indicating internal error is set. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarResultCode <p>This trap can be diluted.</p>	Error
10001	jnxAAATrapLicenseCheckFailure	<p>This trap is sent when a disposition indicating license check failure is set. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarDisposition • jnxAAATrapVarLicenseCheckFailureReason <p>This trap can be diluted.</p>	Error
10002	jnxAAATrapResourceFailure	<p>This trap is sent when a disposition indicating resource failure is set. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarDisposition <p>This trap can be diluted.</p>	Error
10003	jnxAAATrapLogFileFailure	<p>This trap is sent when an attempt to open, create, or write a log file encounters a failure. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences <p>This trap can be diluted.</p>	Error

Table 201 on page 586 lists additional proprietary SNMP traps related to the Diameter protocol. These traps are available in the **jnx-aaa.mib** and **jnx-diameter-base-protocol.mib**. Table 201 on page 586 consists of the following columns:

- **OID Suffix**—Identifies the OID suffix for the trap. To identify the OID number for an alarm, append the OID suffix to the Juniper Networks Diameter mib OID prefix (1.3.6.1.4.1.2636.8.1.2.1.0.0). For example, the ASN.1 number for the **jnxDbpProtocolError** trap is 1.3.6.1.4.1.2636.8.1.2.1.0.0.1.
- **Trap Name**—Identifies the name of the proprietary trap.
- **Description**—Describes when the trap is generated.
- **Type**—Indicates whether the trap is informational, warning, or error.

Table 201: Diameter Trap Definitions

OID Suffix	Trap Name	Description	Type
1	jnxDbpProtocolError	<p>Sent when the Diameter Result-Code attribute value is within the range of 3000 through 3999. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarResultCode • jnxDbpPeerId <p>This trap can be diluted.</p>	Error
2	jnxDbpTransientFailure	<p>Sent when the Diameter Result-Code attribute value is within the range of 4000 through 4999. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarResultCode • jnxDbpPeerId <p>This trap can be diluted.</p>	Error
3	jnxDbpPermanentFailure	<p>Sent when the Diameter Result-Code attribute value is within the range of 5000 through 5999. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxAAATrapVarResultCode • jnxDbpPeerId <p>This trap can be diluted.</p>	Error

Table 201: Diameter Trap Definitions (*continued*)

OID Suffix	Trap Name	Description	Type
4	jnxDbpPeerConnectionDown	<p>Sent when the connection state of the Diameter peer is down. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxDbpPeerId <p>This trap can be diluted.</p>	Error
5	jnxDbpPeerConnectionUp	<p>Sent when a responder transport connection or initiator transport connection is established for communication between the Diameter peer and SBR Carrier. Parameters:</p> <ul style="list-style-type: none"> • jnxAAATrapVarSeverity • jnxAAATrapVarNumberOfOccurrences • jnxDbpPeerId <p>This trap can be diluted.</p>	Error

Rate Statistics

The **fnkrate.mib** describes the structure of the rate statistics SNMP data, including per-NAD and per-Called-Station-ID rate statistics.

For overall server specific rate statistics, Steel-Belted Radius Carrier maintains three types of values for these types of statistics:

- The current-rate statistics specify the rate measured over the most recent rate interval.
- The average-rate statistics specify the rate measured since startup, or the most recent **statistics reset** command.
- The peak-rate statistics specify the highest rate observed since startup, or the most recent **statistics reset** command.

For rate statistics per NAD client and per Called-Station-ID, Steel-Belted Radius Carrier maintains only the current-rate and peak-rate values.

NOTE: Dropped current and peak rates are calculated based on the counters which are incremented when there is any error during building or sending the response to NAD, and do not include packets dropped due to insufficient threads.

Table 202 on page 588 presents an overview of the SNMP accessible overall server specific rate statistics maintained by Steel-Belted Radius Carrier.

Table 202: Server Rate Statistics

OID Suffix	Statistic	Function
1	funkSbrRatesSecondsPerInterval	Specifies the duration (in seconds) of the interval over which the rate statistics are gathered.
2	funkSbrRatesAuthRequestCurrentRate	AuthRequest Current Rate
3	funkSbrRatesAuthRequestAverageRate	AuthRequest Average Rate
4	funkSbrRatesAuthRequestPeakRate	AuthRequest Peak Rate
5	funkSbrRatesAuthAcceptCurrentRate	AuthAccept Current Rate
6	funkSbrRatesAuthAcceptAverageRate	AuthAccept Average Rate
7	funkSbrRatesAuthAcceptPeakRate	AuthAccept Peak Rate
8	funkSbrRatesAuthRejectCurrentRate	AuthReject Current Rate
9	funkSbrRatesAuthRejectAverageRate	AuthReject Average Rate
10	funkSbrRatesAuthRejectPeakRate	AuthReject Peak Rate
11	funkSbrRatesAcctStartCurrentRate	AcctStart Current Rate
12	funkSbrRatesAcctStartAverageRate	AcctStart Average Rate
13	funkSbrRatesAcctStartPeakRate	AcctStart Peak Rate
14	funkSbrRatesAcctStopCurrentRate	AcctStop Current Rate
15	funkSbrRatesAcctStopAverageRate	AcctStop Average Rate

Table 202: Server Rate Statistics (continued)

OID Suffix	Statistic	Function
16	funkSbrRatesAcctStopPeakRate	AcctStop Peak Rate
17	funkSbrRatesProxyAuthRequestCurrentRate	ProxyAuthRequest Current Rate
18	funkSbrRatesProxyAuthRequestAverageRate	ProxyAuthRequest Average Rate
19	funkSbrRatesProxyAuthRequestPeakRate	ProxyAuthRequest Peak Rate
20	funkSbrRatesProxyAcctRequestCurrentRate	ProxyAcctRequest Current Rate
21	funkSbrRatesProxyAcctRequestAverageRate	ProxyAcctRequest Average Rate
22	funkSbrRatesProxyAcctRequestPeakRate	ProxyAcctRequest Peak Rate
23	funkSbrRatesProxyFailTimeoutCurrentRate	ProxyFailTimeout Current Rate
24	funkSbrRatesProxyFailTimeoutAverageRate	ProxyFailTimeout Average Rate
25	funkSbrRatesProxyFailTimeoutPeakRate	ProxyFailTimeout Peak Rate
26	funkSbrRatesProxyFailBadrespCurrentRate	ProxyFailBadresp Current Rate
27	funkSbrRatesProxyFailBadrespAverageRate	ProxyFailBadresp Average Rate
28	funkSbrRatesProxyFailBadrespPeakRate	ProxyFailBadresp Peak Rate
29	funkSbrRatesProxyFailBadsecretCurrentRate	ProxyFailBadsecret Current Rate
30	funkSbrRatesProxyFailBadsecretAverageRate	ProxyFailBadsecret Average Rate
31	funkSbrRatesProxyFailBadsecretPeakRate	ProxyFailBadsecret Peak Rate
32	funkSbrRatesProxyFailMissingresrCurrentRate	ProxyFailMissingresr Current Rate
33	funkSbrRatesProxyFailMissingresrAverageRate	ProxyFailMissingresr Average Rate
34	funkSbrRatesProxyFailMissingresrPeakRate	ProxyFailMissingresr Peak Rate
35	funkSbrRatesProxyRetriesCurrentRate	ProxyRetries Current Rate

Table 202: Server Rate Statistics (continued)

OID Suffix	Statistic	Function
36	funkSbrRatesProxyRetriesAverageRate	ProxyRetries Average Rate
37	funkSbrRatesProxyRetriesPeakRate	ProxyRetries Peak Rate
38	funkSbrRatesProxyAuthRejProxyCurrentRate	ProxyAuthRejProxy Current Rate
39	funkSbrRatesProxyAuthRejProxyAverageRate	ProxyAuthRejProxy Average Rate
40	funkSbrRatesProxyAuthRejProxyPeakRate	ProxyAuthRejProxy Peak Rate
41	funkSbrRatesProxyAcctFailProxCurrentRate	ProxyAcctFailProx Current Rate
42	funkSbrRatesProxyAcctFailProxAverageRate	ProxyAcctFailProx Average Rate
43	funkSbrRatesProxyAcctFailProxPeakRate	ProxyAcctFailProx Peak Rate
44	funkSbrRatesProxyAuthRejProxyErrorCurrentRate	ProxyAuthRejProxyError Current Rate
45	funkSbrRatesProxyAuthRejProxyErrorAverageRate	ProxyAuthRejProxyError Average Rate
46	funkSbrRatesProxyAuthRejProxyErrorPeakRate	ProxyAuthRejProxyError Peak Rate
47	funkSbrRatesAcctInterimCurrentRate	AcctInterim Current Rate
48	funkSbrRatesAcctInterimAverageRate	AcctInterim Average Rate
49	funkSbrRatesAcctInterimPeakRate	AcctInterim Peak Rate

Table 203 on page 590 presents an overview of the SNMP accessible NAD client specific rate statistics maintained by Steel-Belted Radius Carrier.

Table 203: NAD Client Rate Statistics

OID Suffix	Statistic	Description
17	funkSbrClientAuthRequestCurrentRate	Current rate of Access-Requests received from the specific NAD

Table 203: NAD Client Rate Statistics (*continued*)

OID Suffix	Statistic	Description
18	funkSbrClientAuthAcceptCurrentRate	Current rate of Access-Accepts sent to the specific NAD
19	funkSbrClientAuthRejectCurrentRate	Current rate of Access-Rejects sent to the specific NAD
20	funkSbrClientAuthDroppedCurrentRate	Current rate of Authentication-Requests which are dropped due to failure while processing the response to a specific NAD
21	funkSbrClientAcctStartCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Start received from the specific NAD
22	funkSbrClientAcctStopCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Stop received from the specific NAD
23	funkSbrClientAcctInterimCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Interim-update received from the specific NAD
24	funkSbrClientAcctRequestCurrentRate	Current rate of Accounting-Requests received from the specific NAD
25	funkSbrClientAcctDroppedCurrentRate	Current rate of Accounting-Requests which are dropped due to failure while processing the response to a specific NAD
26	funkSbrClientProxyAuthRequestCurrentRate	Current rate of Authentication-Requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")

Table 203: NAD Client Rate Statistics (*continued*)

OID Suffix	Statistic	Description
27	funkSbrClientProxyAcctRequestCurrentRate	Current rate of Accounting-Requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")
29	funkSbrClientProxyTotalRequestCurrentRate	Current rate of all RADIUS requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")
30	funkSbrClientProxyFailTimeoutCurrentRate	Current timeout rate for all RADIUS requests forwarded to the specific NAD
31	funkSbrClientAuthRequestPeakRate	Peak rate of Access-Requests received from the specific NAD
32	funkSbrClientAuthAcceptPeakRate	Peak rate of Access-Accepts sent to the specific NAD
33	funkSbrClientAuthRejectPeakRate	Peak rate of Access-Rejects sent to the specific NAD
34	funkSbrClientAuthDroppedPeakRate	Peak rate of Authentication-Requests which are dropped due to failure while processing the response to a specific NAD
35	funkSbrClientAcctStartPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Start received from the specific NAD
36	funkSbrClientAcctStopPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Stop received from the specific NAD
37	funkSbrClientAcctInterimPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Interim-update received from the specific NAD

Table 203: NAD Client Rate Statistics (*continued*)

OID Suffix	Statistic	Description
38	funkSbrClientAcctRequestPeakRate	Peak rate of Accounting-Requests received from the specific NAD
39	funkSbrClientAcctDroppedPeakRate	Peak rate of Accounting-Requests which are dropped due to failure while processing the response to a specific NAD
40	funkSbrClientProxyAuthRequestPeakRate	Peak rate of Authentication-Requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")
41	funkSbrClientProxyAcctRequestPeakRate	Peak rate of Accounting-Requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")
42	funkSbrClientProxyTotalRequestPeakRate	Peak rate of all RADIUS requests received from the specific NAD and forwarded to other RADIUS servers ("proxied")
43	funkSbrClientProxyFailTimeoutPeakRate	Peak timeout rate for all RADIUS requests forwarded to the specific NAD

[Table 204 on page 593](#) presents an overview of the SNMP accessible Called-Station-ID specific rate statistics maintained by Steel-Belted Radius Carrier.

Table 204: Called-Station-ID Rate Statistics

OID Suffix	Statistic	Description
17	funkSbrCalledSIdAuthRequestCurrentRate	Current rate of Access-Requests containing the specified Called-Station-ID

Table 204: Called-Station-ID Rate Statistics (continued)

OID Suffix	Statistic	Description
18	funkSbrCalledSIdAuthAcceptCurrentRate	Current rate of Access-Accepts in response to requests containing the specified Called-Station-ID
19	funkSbrCalledSIdAuthRejectCurrentRate	Current rate of Access-Rejects in response to requests containing the specified Called-Station-ID
20	funkSbrCalledSIdAuthDroppedCurrentRate	Current rate of Authentication-Requests which are dropped due to failure while processing the response to a request which contains the specified Called-Station-ID
21	funkSbrCalledSIdAcctStartCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Start and containing the specified Called-Station-ID
22	funkSbrCalledSIdAcctStopCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Stop and containing the specified Called-Station-ID
23	funkSbrCalledSIdAcctInterimCurrentRate	Current rate of Accounting-Requests with Acct-Status-Type set to Interim-update and containing the specified Called-Station-ID
24	funkSbrCalledSIdAcctRequestCurrentRate	Current rate of all Accounting-Requests containing the specified Called-Station-ID
25	funkSbrCalledSIdAcctDroppedCurrentRate	Current rate of Accounting-Requests which are dropped due to failure while processing the response containing the specified Called-Station-ID

Table 204: Called-Station-ID Rate Statistics (continued)

OID Suffix	Statistic	Description
26	funkSbrCalledSIdProxyAuthRequestCurrentRate	Current rate of Authentication-Requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")
27	funkSbrCalledSIdProxyAcctRequestCurrentRate	Current rate of Accounting-Requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")
28	funkSbrCalledSIdProxyTotalRequestCurrentRate	Current rate of all RADIUS requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")
29	funkSbrCalledSIdProxyFailTimeoutCurrentRate	Current timeout rate for all RADIUS requests containing the specified Called-Station-ID forwarded to other RADIUS servers (proxied)
30	funkSbrCalledSIdAuthRequestPeakRate	Peak rate of Access-Requests in response to requests containing the specified Called-Station-ID
31	funkSbrCalledSIdAuthAcceptPeakRate	Peak rate of Access-Accepts in response to requests containing the specified Called-Station-ID
32	funkSbrCalledSIdAuthRejectPeakRate	Peak rate of Access-Rejects containing the specified Called-Station-ID
33	funkSbrCalledSIdAuthDroppedPeakRate	Peak rate of Authentication-Requests containing the specified Called-Station-Id which are dropped due to failure while processing the response

Table 204: Called-Station-ID Rate Statistics (continued)

OID Suffix	Statistic	Description
34	funkSbrCalledSIdAcctStartPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Start and containing the specified Called-Station-ID
35	funkSbrCalledSIdAcctStopPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Stop and containing the specified Called-Station-ID
36	funkSbrCalledSIdAcctInterimPeakRate	Peak rate of Accounting-Requests with Acct-Status-Type set to Interim-update and containing the specified Called-Station-ID
37	funkSbrCalledSIdAcctRequestPeakRate	Peak rate of all Accounting-Requests containing the specified Called-Station-ID
38	funkSbrCalledSIdAcctDroppedPeakRate	Peak rate of Accounting-Requests which are dropped due to failure while processing the response to a request which contains the specified Called-Station-ID
39	funkSbrCalledSIdProxyAuthRequestPeakRate	Peak rate of Authentication-Requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")
40	funkSbrCalledSIdProxyAcctRequestPeakRate	Peak rate of Accounting-Requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")
41	funkSbrCalledSIdProxyTotalRequestPeakRate	Peak rate of all RADIUS requests containing the specified Called-Station-ID forwarded to other RADIUS servers ("proxied")

Table 204: Called-Station-ID Rate Statistics (*continued*)

OID Suffix	Statistic	Description
42	funkSbrCalledSldProxyFailTimeoutPeakRate	Peak timeout rate for all RADIUS requests containing the specified Called-Station-ID forwarded to other RADIUS servers (proxied)

PART 7

Appendixes

This part contains these appendixes:

APPENDIX A

Authentication Protocols

This appendix contains a matrix of authentication methods and their supported authentication protocols.

Table 205: Authentication Protocols

Method	PAP	CHAP	MS-CHAP	MS-CHAP-V2	EAP-MSCHAP-V2	EAP-MD5
Microsoft PEAP available inner authentication protocols	No	No	No	No	Yes	No
TTLS available inner authentication protocols	Yes	Yes	Yes	Yes	Yes	Yes
Native	Yes	Yes	Yes	Yes	Yes	Yes
Native (password saved as Allow PAP only, {SHA} or {crypt}).	Yes	No	No	No	No	No
UNIX authentication methods						
Solaris User	Yes	No	No	No	No	No
Solaris Group	Yes	No	No	No	No	No
LDAP						
BIND (this includes AD and eDirectory/NDS)	Yes	No	No	No	No	No

Table 205: Authentication Protocols (*continued*)

Method	PAP	CHAP	MS-CHAP	MS-CHAP-V2	EAP-MSCHAP-V2	EAP-MD5
BINDNAME (password stored in clear-text)	Yes	Yes	Yes	Yes	Yes	Yes
BINDNAME (password stored in SHA/Solaris Crypt text)	Yes	No	No	No	No	No
BINDNAME (password Stored as MD4 hash of Unicode value text)	Yes	No	No	Yes	Yes	No
BINDNAME (password Stored as enc-md5)	Yes	Yes	No	No	No	No
SQL						
Password stored in clear-text	Yes	Yes	Yes	Yes	Yes	Yes
Password password stored in SHA/Solaris Crypt text	Yes	No	No	No	No	No
Password stored as {MD4} hash of Unicode value text	Yes	No	No	Yes	Yes	No
Password stored as {enc-md5}	Yes	Yes	No	No	No	No

APPENDIX B

Vendor-Specific Attributes

This appendix describes the Juniper Networks vendor-specific attributes used with Steel-Belted Radius Carrier.

Table 206: Steel-Belted Radius Carrier Vendor-Specific Attributes

Attribute Name	Purpose
Funk-Allowed-Access-Hours	<p>May be placed in the check list for a user or profile entry to control the exact time periods during which a user may be allowed access.</p> <p>Funk-Allowed-Access-Hours is a variable-length string that identifies time periods in a 7-day week of 24-hour days. This string consists of one or more day specifiers (each of which may list one or more days and/or ranges of days) followed by one or more ranges of 24-hour times, in minutes.</p>
Funk-Concurrent-Login-Limit	Reserved for future use.
Funk-Full-User-Name	Reserved for future use.
Funk-Location-Group-Id	<p>Added to an inbound authentication or accounting request when the request is matched to a location group and AddFunkLocationGroupIdToRequest is set to 1 in the radius.ini file.</p> <p>The value of the attribute is the name of the location group.</p>

Table 206: Steel-Belted Radius Carrier Vendor-Specific Attributes (*continued*)

Attribute Name	Purpose
Funk-Peer-Cert-Hash	<p>Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1.</p> <p>The value of the attribute is the hexadecimal ASCII representation of the SHA1 hash of the client's certificate.</p>
Funk-Peer-Cert-Issuer	<p>Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1.</p> <p>The value of the attribute is the contents of the Issuer attribute of the client's certificate.</p>
Funk-Peer-Cert-Principal	<p>Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1.</p> <p>The value of the attribute is the contents of the Subject Alternate Name or Other Name attribute of the client's certificate.</p>
Funk-Peer-Cert-Subject	<p>Added to the list of attributes available for secondary authorization processing when EAP-TLS is loaded as an automatic EAP helper. The attribute is added to the request only if Include_Certificate_Info in the [Secondary_Authorization] section of tlsauth.eap is set to 1.</p> <p>The value of the attribute is the contents of the Subject attribute of the client's certificate.</p>

Table 206: Steel-Belted Radius Carrier Vendor-Specific Attributes (*continued*)

Attribute Name	Purpose
Funk-Round-Robin-Group	<p>May be placed in the return list for a user or profile entry to dynamically assign an attribute set from an Attribute Value Pool at login time.</p> <p>The value of this attribute must be set to the .rr file name which defines the Attribute Value Pool.</p>
Funk-Source-IP-Address	<p>Added to the list of attributes available for request processing if AddSourceIPAddressAttrToRequest is set to 1 in the [Configuration] section of the radius.ini file.</p> <p>The value of the attribute is the IP address from which the packet containing the request was received.</p>
Funk-Source-IPv6-Address	Reserved for future use.
Funk-Tribe-Name	Reserved for future use.

APPENDIX C

Configuration Examples

IN THIS SECTION

- [Steel-Belted Radius Carrier: 3G-to-Wi-Fi Offload Solution Using the SBR MAP Gateway with EAP-SIM or EAP-AKA | 604](#)

Steel-Belted Radius Carrier: 3G-to-Wi-Fi Offload Solution Using the SBR MAP Gateway with EAP-SIM or EAP-AKA

IN THIS SECTION

- [Requirements | 605](#)
- [Overview | 605](#)
- [Configuration | 607](#)

This example explains the Signalware Man Machine Language (MML) configurations to support the 3G-to-Wi-Fi offload solution using the SBR MAP Gateway with EAP-SIM or EAP-AKA.

Requirements

This example uses the following hardware and software components:

- Standalone SBR Carrier server
- Signalware
- SBR software licenses:
 - SBR-CAR-AAA—Base server license
 - SBR-CAR-SIM—SIM authentication module
 - SBR-HLR-SIG—SBR Carrier HLR Gateway - Signalware SIGTRAN stack (includes two SIGTRAN associations)
- A client with EAP-SIM or EAP-AKA enabled with an HLR
- 802.1X-capable Wi-Fi infrastructure (applicable to Wi-Fi networks only)

Overview

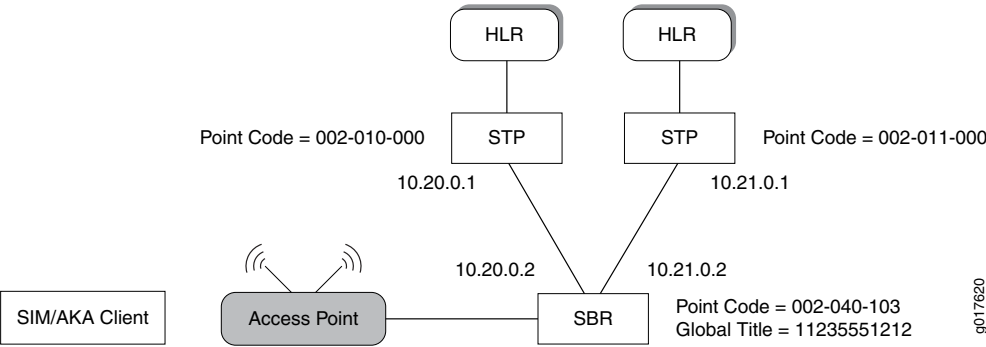
With the increase in the number of smartphones and other mobile devices in the 3G network, the mobile data traffic volume increases. This can result in network congestion. The 3G-to-Wi-Fi offload solution helps to alleviate the network congestion in a 3G network by offloading the mobile data traffic to a Wi-Fi hotspot. The policy to offload the mobile data traffic to a Wi-Fi hotspot can be configured by the end user or the network operator.

For example, when a smartphone user in a 3G network enters a Wi-Fi hotspot, the user is authenticated by SBR using the IMSI with credentials provided by the HLR. In this example, upon authentication, the user is authorized to access the Wi-Fi hotspot using encryption keys generated by EAP SIM or EAP AKA authentication.

Topology

The following topology ([Figure 18 on page 606](#)) shows the components of a typical Wi-Fi infrastructure:

Figure 18: Wi-Fi Infrastructure Topology



NOTE: The configurations described in this document are based on the information gathered prior to deployment.

You can use the following tables as a template to gather information from the customer prior to deployment:

- [Table 207 on page 606](#): SIGTRAN IP Address
- [Table 208 on page 606](#): SIGTRAN Connectivity
- [Table 209 on page 607](#): Global Title (GT)
- [Table 210 on page 607](#): Numbering Plan (NP)

Table 207: SIGTRAN IP Address Details

Component	IP Address	Subnet Mask	Gateway
SBR Carrier MAP Gateway	10.20.0.2	255.255.255.248	-
SBR Carrier MAP Gateway	10.21.0.2	255.255.255.248	-

Table 208: SIGTRAN Connectivity Details

SIGTRAN Parameters	Local IP Address	STP IP Address (active)	STP IP Address (standby)	SCTP Port (local)	SCTP Port (remote)	Routing Context	Network Appearance - NA=0		OPC SSN	DPC SSN
							Originating PC (dec)	Destination PC (dec)		
SBR Carrier MAP Gateway	10.20.0.2	10.20.0.1	-	2051	2051	-	002-040-103	002-010-000	7	6

Table 208: SIGTRAN Connectivity Details (continued)

SIGTRAN Parameters	Local IP Address	STP IP Address (active)	STP IP Address (standby)	SCTP Port (local)	SCTP Port (remote)	Routing Context	Network Appearance - NA=0		OPC SSN	DPC SSN
							Originating PC (dec)	Destination PC (dec)		
SBR Carrier MAP Gateway	10.21.0.2	10.21.0.1	-	2051	2051	-	002-040-103	002-011-000	7	6

Table 209: Global Title (GT) Details

Component	GT Address
SBR Carrier MAP Gateway	11235551212
STP	-

Table 210: Numbering Plan (NP)

Mode	NP
Transmit (SBR >>> STP)	E.164 (7)
Receive (STP >>> SBR)	E.164 (7)

Configuration

IN THIS SECTION

- [Install Signalware | 608](#)
- [Create Links, Link Sets, and Route Sets | 608](#)
- [Configure authGateway and GWrelay Applications for HLR Communication | 609](#)
- [Configure the authGateway Routing Location Information | 609](#)
- [Configure the authGateway.conf File | 610](#)
- [Configure the authGateway Startup Information | 611](#)
- [Configure the GWrelay.conf File | 612](#)

- Start the GWrelay Process | 613
- Configure the ulcmmg.conf File | 613

To configure the communication pathways, you must:

Install Signalware

Step-by-Step Procedure

To install and configure Signalware on a Steel-Belted Radius Carrier server, see the *Steel-Belted Radius Carrier Installation Guide*.

Create Links, Link Sets, and Route Sets

Step-by-Step Procedure

The following configuration is used to create links, link sets, and route sets:

1. Define the SBR's Own Signaling Point Code (OSPC). In this example, NI (Network Indicator) of NATO—National Network 0 is used.

```
CREATE-OSPC:PC=002-040-103,NI=NAT0;
```

2. Set up the M3UA link sets (LSET1 and LSET2) and use IP Signaling Point to IP Signaling Point configuration (IPSP-IPSP). See [Figure 18 on page 606](#) for addresses and point codes.

```
CREATE-M3UA-LSET:LSET=LSET1,TYPE=IPSP-IPSP,RADDR=10.20.0.1,PC=002-010-000;
CREATE-M3UA-LSET:LSET=LSET2,TYPE=IPSP-IPSP,RADDR=10.21.0.1,PC=002-011-000;
```

3. After the M3UA link sets are defined, signaling links are created using the link sets defined in [Step 2](#). In this example, the default port 2051 is used.

```
CREATE-M3UA-SLK:SLK=QFE20,LSET=LSET1,LADDR=10.20.0.2,RADDR=10.20.0.1,
MODE=CONNECT,LPORT=2051,RPORT=2051;
CREATE-M3UA-SLK:SLK=QFE21,LSET=LSET2,LADDR=10.21.0.2,RADDR=10.21.0.1,
MODE=CONNECT,LPORT=2051,RPORT=2051;
```

4. Activate the signaling links using the following command:


```
ACTIVATE-M3UA-SLK:SLK=QFE20;
ACTIVATE-M3UA-SLK:SLK=QFE21;
```

5. Define the route set (a route set is simply a collection of routes). You must also specifically allow routes to be used.

```
CREATE-RSET:RSET=STP1,PC=002-010-000,RTES=LSET1;
CREATE-RSET:RSET=STP2,PC=002-011-000,RTES=LSET2;
ALLOW-RSET:RSET=STP1;
ALLOW-RSET:RSET=STP2;
```

Configure authGateway and GWrelay Applications for HLR Communication

Step-by-Step Procedure

The authGateway application manages all communication between SBR Carrier and the HLR. The authGateway application also implements the Mobile Application Port (MAP) protocol and MAP messages that are sent through the Signalware SIGTRAN protocol stack and out to the HLR and back. Multiple authGateway instances can be used to process multiple requests for authentication and authorization information simultaneously. The GWrelay application is used to pass authentication requests between SBR Carrier and the authGateway instances in a round-robin method. The GWrelay application establishes an SCTP connection with each authGateway instance through unique source and destination ports.

Configuration of authGateway and GWrelay applications requires you to complete the activities described in the following sections:

- [Configure the authGateway Routing Location Information on page 609](#)
- [Configure the authGateway.conf File on page 610](#)
- [Configure the authGateway Startup Information on page 611](#)
- [Configure the GWrelay.conf File on page 612](#)
- [Start the GWrelay Process on page 613](#)
- [Configure the ulcmmg.conf File on page 613](#)

Configure the authGateway Routing Location Information

Step-by-Step Procedure

This section describes how to configure the local routing and the remote routing options.

- For local routing, identify one or more concerned point codes (CPCs) and the local application gateway.
- For remote routing, identify one or more point codes of the HLR and the remote application.

The following actions take place in this configuration example for local and remote routing:

1. authGateway is assigned a subsystem number (SSN) of 7 on the local host and the concerned point code on the HLR is identified as 002-010-000. The subsystem number (application) on the remote host is identified as 6.

```
CREATE-CPC:PC=002-010-000,SSN=7;
CREATE-REMSSN:PC=002-010-000,SSN=6;
```

2. authGateway is assigned a subsystem number (SSN) of 7 on the local host and the concerned point code on the HLR is identified as 002-011-000. The subsystem number (application) on the remote host is identified as 6.

```
CREATE-CPC:PC=002-011-000,SSN=7;
CREATE-REMSSN:PC=002-011-000,SSN=6;
```

3. Create one or more Global Title translations for the remote HLR (if GT routing is used).

The following commands set up the Global Title routing for both directions (outbound and inbound). Outbound GT routing using any IMSI starting with 123 uses PC 002-010-000. Inbound routing uses the GT of 11235551212 routing to the SBR point code.

```
CREATE-GT:TT=10,NP=ISDN-TEL,DIG="11235551212",PC=002-040-103,SSN=7,RI=GT;
CREATE-GT:TT=9,NP=ISDN-TEL,DIG="123",PC=002-010-000,SSN=6,RI=GT;
```

Configure the authGateway.conf File

Step-by-Step Procedure

The **authGateway.conf** file specifies remote routing and authorization options for the authGateway application.

- Remote routing options control how the remote HLR is addressed based on the incoming IMSI.
- Authorization options control whether or not a subscriber requesting an account is authorized for WLAN access, and which Steel-Belted Radius Carrier profile or native user is used.

For more information about configuring the **authGateway.conf** file for remote routing and authorization options, see the *Steel-Belted Radius Carrier Installation Guide*.

Configure the authGateway Startup Information

Step-by-Step Procedure

The **CREATE-PROCESS** and **START-PROCESS** commands start the authGateway process (by calling the **authGateway.conf** file), using options that you specify. For more information about the syntax and usage of the commands, see the *Steel-Belted Radius Carrier Installation Guide*.

Use the following configuration example to create and start three authGateway instances:

```
CREATE-PROCESS:NAME="GMT", CE="sbr-blr-vm5",
EXEC="/opt/JNPRsbr/radius/authGateway -debug 0xff -trace -name GMT -port 2003 -host
sbr-blr-vm5
-node SBRLX -prot C7 -conf /opt/JNPRsbr/radius/conf/authGateway.conf -lri 1
-lpc 12501 -lssn 252 -rssn 101 -appctx 3";
debug 0xff -trace -tracefile /opt/signalw/radius/authGateway.out

START-PROCESS:NAME="GMT",CE="sbr-blr-vm5";

CREATE-PROCESS:NAME="GMT1", CE="sbr-blr-vm5",
EXEC="/opt/JNPRsbr/radius/authGateway -debug 0xff -trace -name GMT1 -port 2005 -host
sbr-blr-vm5
-node SBRLX -prot C7 -conf /opt/JNPRsbr/radius/conf/authGateway.conf -lri 1
-lpc 12501 -lssn 252 -rssn 101 -appctx 3";
debug 0xff -trace -tracefile /opt/signalw/radius/authGateway1.out

START-PROCESS:NAME="GMT1",CE="sbr-blr-vm5";

CREATE-PROCESS:NAME="GMT2", CE="sbr-blr-vm5",
EXEC="/opt/JNPRsbr/radius/authGateway -debug 0xff -trace -name GMT2 -port 2007 -host
sbr-blr-vm5
-node SBRLX -prot C7 -conf /opt/JNPRsbr/radius/conf/authGateway.conf -lri 1
-lpc 12501 -lssn 252 -rssn 101 -appctx 3";
debug 0xff -trace -tracefile /opt/signalw/radius/authGateway2.out

START-PROCESS:NAME="GMT2",CE="sbr-blr-vm5";
```

Configure the GWrelay.conf File

Step-by-Step Procedure

The GWrelay application is used to pass authentication requests between SBR Carrier and the authGateway instances in a round-robin method. The **GWrelay.conf** file is used to define the source and destination ports through which an SCTP connection is established between the GWrelay application and the authGateway instance.

You can modify the LOCAL_HOST, REMOTE_HOST, and RELAY_SERVER lines in the **GWrelay.conf** file to define DNS names and port numbers. When you specify a DNS name for a local or remote host, you can enter the host's IP address in brackets as a backup. We recommend that you make hostname and IP address entries in the **/etc/hosts** file because it is more reliable than DNS.

The following example explains how to define source and destination ports for three authGateway instances:

```
LOCAL_HOST sbr-blr-vm5:2002
REMOTE_HOST sbr-blr-vm5:2003 [10.20.0.2]

LOCAL_HOST sbr-blr-vm5:2004
REMOTE_HOST sbr-blr-vm5:2005 [10.20.0.2]

LOCAL_HOST sbr-blr-vm5:2006
REMOTE_HOST sbr-blr-vm5:2007 [10.20.0.2]

RELAY_SERVER sbr-blr-vm5:2000
```

NOTE: The specified host-name and port parameters in the REMOTE_HOST line must match the -host and -port options in the MML CREATE-PROCESS statement, respectively.

For more information, see the *Steel-Belted Radius Carrier Installation Guide*.

Start the GWrelay Process

Step-by-Step Procedure

You can use the **sbrd** script to start and stop the GWrelay process. All **sbrd** commands can be executed only by the root user. To start the GWrelay process, execute **./sbrd start GWrelay**. To stop the GWrelay process, execute **./sbrd stop GWrelay**. To restart the GWrelay process, execute **./sbrd restart GWrelay**.

Configure the ulcmmg.conf File

Step-by-Step Procedure

The **ulcmmg.conf** file establishes the connection between the GWrelay application and SBR Carrier.

The **ulcmmg.conf** file shipped with SBR Carrier can be modified so that hostnames of LOCAL_HOST and REMOTE_HOST are same. If you specify a DNS name for a local or remote host, you can enter the host's IP address in brackets as a backup. Making an entry in the **/etc/hosts** file is recommended because it is more reliable than DNS.

The following is an example of the LOCAL_HOST and REMOTE_HOST values in the **ulcmmg.conf** file:

```
LOCAL_HOST myhost.com:2001  
REMOTE_HOST myhost.com:2000 [10.20.0.2]
```

For more information, see the *Steel-Belted Radius Carrier Installation Guide*.

APPENDIX D

Detailed Use Cases

IN THIS SECTION

- [Using SQL Accessors | 615](#)
- [Using LDAP Accessors | 617](#)
- [Assigning IP Addresses Based on APN | 618](#)
- [Adding Attributes to an Access-Accept | 622](#)
- [Interfacing with a Kineto IP Network Controller | 630](#)
- [Using Managed IPv6 Address Pools | 651](#)

This chapter explains use cases on the following:

Using SQL Accessors

IN THIS SECTION

- [Configuring gsmmap.gen for Key Field Identification | 616](#)
- [Configuring sqlaccessor.gen or sqlaccessor_jdbc.gen for Key Field Identification | 616](#)

This section explains the following:

Configuring gsmmap.gen for Key Field Identification

The choice of IMSI or MSISDN as the key field is identified in the **gsmmap.gen** file with the **KeyForAuthorization** field. (KeyForAuthorization can be MSISDN or IMSI.) In the following examples, MSISDN is identified as the key field.

Examples

```
gsmmap.gen file (Oracle or JDBC)
[SQLDatabase]
ModuleType=Database
DatabaseAccessorMethodName=SQL Accessor
KeyForAuthorization=MSISDN
gsmmap.gen file (LDAP)
[LDAPDatabase]
ModuleType=Database
DatabaseAccessorMethodName=LDAP Accessor
KeyForAuthorization=MSISDN
```

Configuring sqlaccessor.gen or sqlaccessor_jdbc.gen for Key Field Identification

For SQL databases, the SELECT statement in the [Settings] section of the **sqlaccessor.gen** file or the **sqlaccessor_jdbc.gen** file identifies the database column name of the key field. In the following example, subscriber_id is identified as the column containing the key field.

NOTE: Oracle front-end applications are not supported on a Linux platform. The **sqlaccessor.so** and **sqlaccessor.gen** files are specific to Oracle plug-ins and must not be installed on a Linux platform. You must instead use the **sqlaccessor_jdbc.gen** file.

Examples

Oracle Example: sqlaccessor.gen file


```
[Settings]
MethodName=SQL Accessor
Connect=my_user_name/password@servicename
SQL=SELECT service_type FROM table1 WHERE subscriber_id=@KeyToRecord
```

JDBC Example: sqlaccessor_jdbc.gen file

```
[Settings]
MethodName=SQL Accessor
Driver=com/provider/jdbc/sqlserver/SQLServerDriver
ConnectDelimiter=;
Connect=DSN=jdbc:provider:driver1:dsn_name;UID=db_username;PWD=db_password
SQL=SELECT service_type FROM table1 WHERE subscriber_id=@KeyToRecord
```

The corresponding database must contain a column name (as specified in the SELECT statement) containing the key field. In the following example, the column name of subscriber_id contains MSISDN values that serve as record keys.

Table 211: Example SQL Database

subscriber_id	service_type	street_address
1234	basic	10 Main Street
6889	premium	15 School Street

Using LDAP Accessors

IN THIS SECTION

- [Configuring ldapaccessor.gen for Key Field Identification | 618](#)

This section explains the following:

Configuring ldapaccessor.gen for Key Field Identification

For LDAP directories, the [Request] section and the [Search/DoLdapSearch] section identifies the key to the LDAP directory. You can set the **KeyToRecord** field in the [Request] section to either MSISDN or IMSI. The [Search/DoLdapSearch] section identifies the column name in the LDAP directory that contains the record key. In [Table 212 on page 618](#), the wlanMSISDN column contains the key MSISDN data.

ldapaccessor.gen file

```
[Request]
KeyToRecord=key

[Search/DoLdapSearch]
Base=o=bigco.com
Scope=2

Filter=wlanMSISDN=key
```

NOTE: See [“LDAP Accessor Files” on page 460](#) for more information about ldapaccessor.gen.

Table 212: Example LDAP Directory

wlanMSISDN	service_type	street_address
1234	basic	10 Main Street
6889	premium	15 School Street

Assigning IP Addresses Based on APN

Steel-Belted Radius Carrier can assign IP addresses to mobile devices based on the access point name (APN).

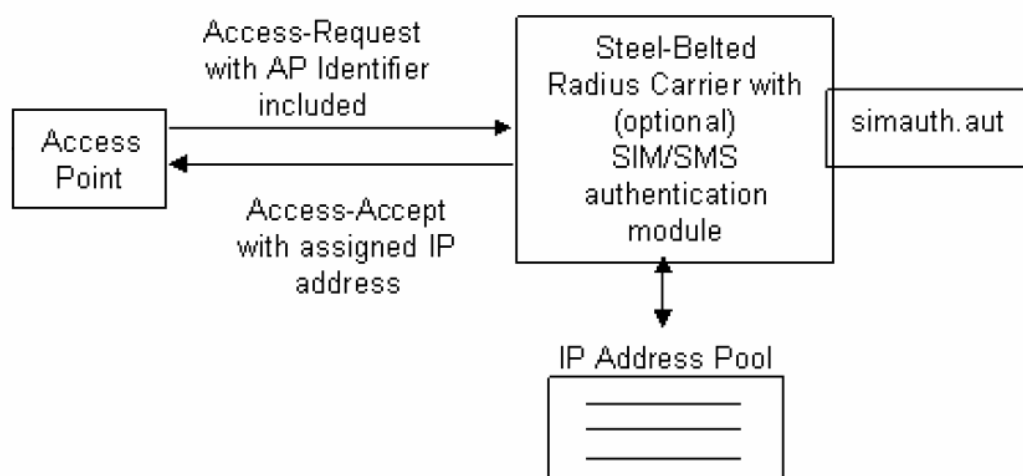
Overview

APN-based IP address assignment enables Steel-Belted Radius Carrier to perform the task of address assignment, rather than requiring the access point to assign addresses.

This feature works by configuring IP address pools, each of which consists of a set of IP addresses that can be assigned to a mobile device. You then configure an access point name (APN) to be associated with a particular pool. When an Access-Request is received, Steel-Belted Radius Carrier selects an IP address from the pool that is assigned to the APN handling the request.

Figure 19 on page 619 shows the configuration of IP address assignment based on APN.

Figure 19: IP Address Assignment Based on Access Point Name



NOTE: APN-based IP address assignment takes precedence over all other methods of IP address assignment except if an IP address (or pool name) is added to an Access-Accept as the value of the Framed-IP-Address attribute.

For information about how to add any attribute from a subscriber database (such as a SQL database), see [“Adding Attributes to an Access-Accept” on page 622](#). For example, you can retrieve the IP address from a SQL database and include it as the value of Framed-IP-Address in an Access-Accept.

Tasks for Assigning IP Address Based on Access Point

Assigning IP addresses based on access point involves the following main tasks:

- Configure **simauth.aut**
- Create an address pool

Each of these tasks is described in the following sections.

Configuring **simauth.aut** for IP Address Assignment

The **simauth.aut** configuration file retrieves the IP address from a pool to be returned with the Access-Accept.

To configure **simauth.aut** for IP address assignment based on access point:

1. In the [Settings] section of **simauth.aut**, define the attribute that identifies the access point. This attribute is usually 3GPP-WLAN-APN-Id or Called-Station-ID. Use the following format:

AssignIpPoolByAttr = *attribute*

where:

attribute is the name of the string type attribute to be used to identify the access point.

Example:

```
[Settings]
AssignIpPoolByAttr = 3GPP-WLAN-APN-Id
```

2. In the [Settings] section of **simauth.aut**, define the attribute to contain the IP address to be assigned to the mobile device. This attribute is returned with the Access-Accept. Use the following format:

```
AssignIpPoolDestAttr = attribute_for_address
```

where:

attribute_for_address is the attribute used to return the IP address in the Access-Accept. This attribute must be consistent with an IPv4 IP address. (It is usually Framed-Ip-Address.)

Example:

```
[Settings]
AssignIpPoolDestAttr = Framed-Ip-Address
```

3. In the [IpPools] section of **simauth.aut**, define the access point identifiers that associate specific pools with access points. Use the following format:

```
attribute-value = poolname
```

where:

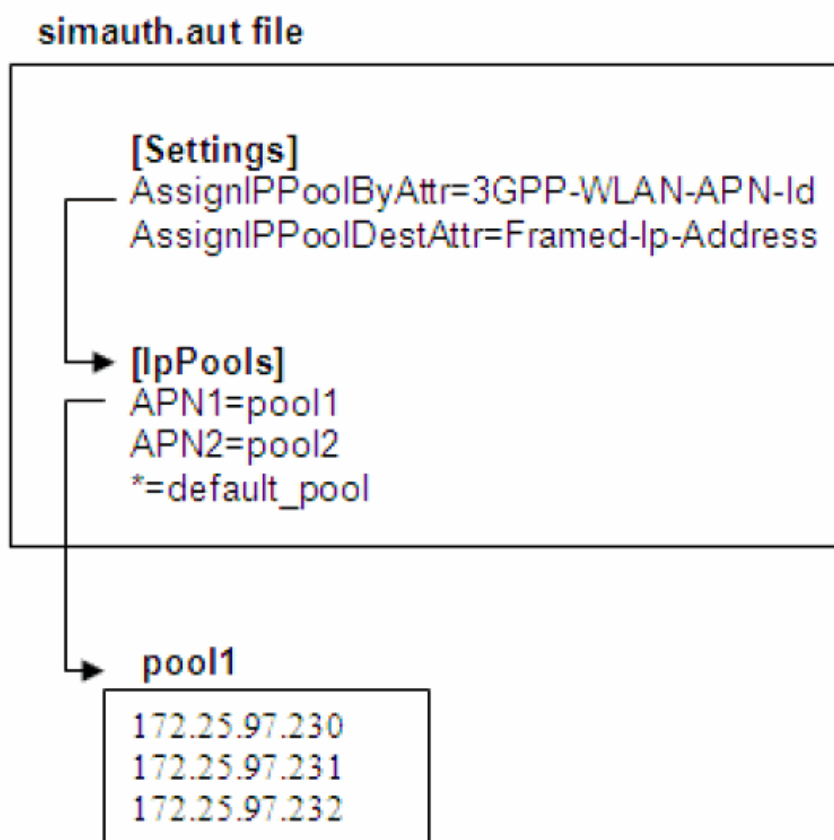
attribute-value is the access point identifier, and **poolname** is the name of an IP address pool created using the Web GUI.

Example:

```
[IpPools]
ASN1 = Pool1
ASN2 = Pool2
* = Default_Pool
```

Figure 20 on page 621 shows the configuration of **simauth.aut** for assigning IP addresses based on APN. In Figure 20 on page 621, the access point is identified by the value of the attribute 3GPP-WLAN-APN-Id. If the value of 3GPP-WLAN-APN-Id is APN1, then an IP address is taken from pool1. If the value is neither APN1 nor APN2, the address is taken from the pool named default_pool. The IP address is assigned to the attribute Framed-IP-Address, which is returned in the Access-Accept message.

Figure 20: Configuration of IP Address Assignment Based on Access Point



Create the IP Address Pool

See the *SBR Carrier Administration and Configuration Guide* for more information about administering IP address pools.

Adding Attributes to an Access-Accept

This feature allows you to add attribute values retrieved from an external subscriber database to Access-Accept message. For example, you might want to include the subscriber's level of service in the Access-Accept as the value of the attribute Reply-Message. Another example might be retrieving the IP address to be assigned to a mobile node and returning it in the Access-Accept as the value of the attribute Framed-IP-Address.

Overview

You can add additional attributes to Access-Accept messages from an external subscriber database. Two authentication plug-ins are used to accomplish the tasks of authentication and adding attributes to an Access-Accept. The authentication plug-ins are:

- The SIMAuth application (acting as the *EAP helper*)
This authenticator provides EAP authentication for the SIM authentication module.
- *Helped authenticator* (usually the SQL plug-in: **radsq1.aut** or **radsq1jdbc.aut**). This authenticator accesses the database, retrieves the specified attributes, and attaches them to the Access-Accept message. The *helped authenticator* does not perform any authentication tasks and its password-checking is suppressed. All authentication is performed by the SIMAuth application (the *EAP helper*).

Data Flow

Authentication of the Access-Request and the addition of attributes to the Access-Accept is handled according to the following flow of data:

1. The mobile device sends an Access-Request to Steel-Belted Radius Carrier.
2. SIMAuth manages the EAP negotiation (challenge, and response).

3. If SIMAuth authenticates the request, it attaches the IMSI and MSISDN of the mobile device, and sends the request to the SQL plug-in: **radsql.aut** or **radsqljdbc.aut**.
4. **radsql.aut** or **radsqljdbc.aut** can use the IMSI or MSISDN as a key to query the database and request attribute values (as a separate step from the SIMAuth authentication).
5. The *helped authenticator* (usually the SQL authentication plug-in: **radsql.aut** or **radsqljdbc.aut**), returns the Access-Accept with attribute values attached.

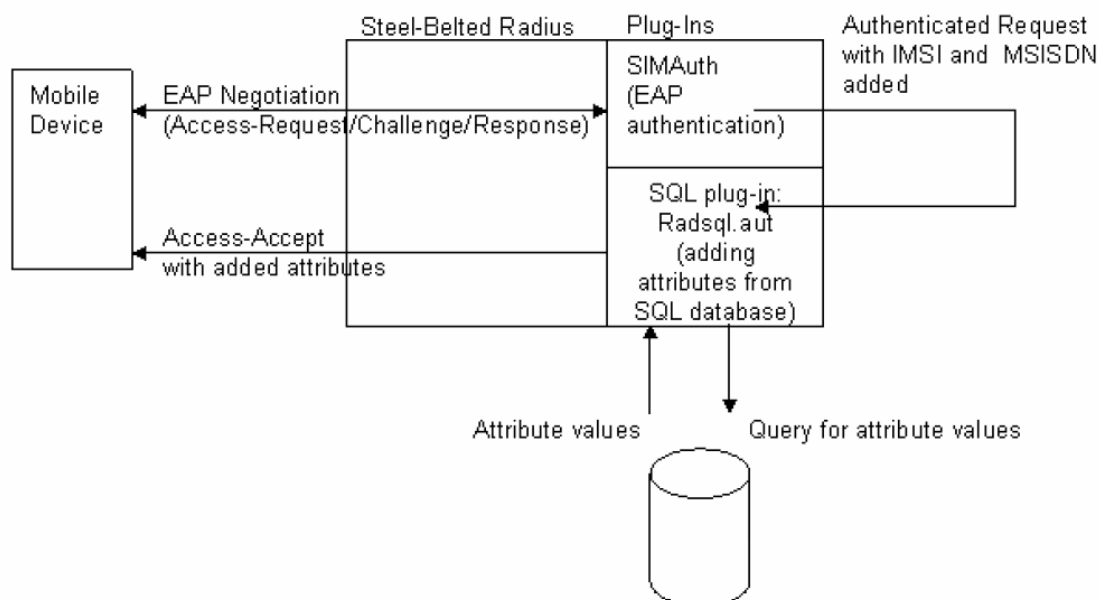
NOTE: SIMAuth is known as a Steel-Belted Radius Carrier *EAP helper* because it performs the EAP authentication for the *helped* authentication method (usually the SQL authentication plug-ins: **radsql.aut** or **radsqljdbc.aut**). Although these SQL plug-ins are usually used for authentication, in this case their function is to access the subscriber database, retrieve attributes, and return them with the Access-Accept.

For complete information about EAP helpers, see the *SBR Carrier Administration and Configuration Guide*.

Figure 21 on page 624 shows an example data flow in which Steel-Belted Radius Carrier, SIMAuth, and the SQL plug-ins (either **radsql.aut** or **radsqljdbc.aut**) work together to perform the following tasks:

- Access authentication (performed by SIMAuth)
- Addition of MSISDN and IMSI to the request (performed by SIMAuth)
- Database access and attribute retrieval (performed by **radsql.aut** in this example called *SQLAuthenticator*)
- Addition of retrieved attributes to the Access-Accept (performed by the SQL plug-in: **radsql.aut**)

Figure 21: Example Data Flow for Addition of Attribute to Access-Accept



Configuration Tasks

To add attributes to the Access-Accept, perform the following tasks:

- Configure the related files, as described in [“Configuring Files for Adding Attributes to Access-Accept” on page 624](#).
- Activate authentication as described in [“Activate the Authentication Method” on page 629](#).

Configuring Files for Adding Attributes to Access-Accept

The following files require special configuration to allow the addition of attributes to the Access-Accept:

- **simauth.aut**
- **simauth.eap**
- **radsql.aut**, **radsqljdbc.aut**, or **ldapauth.aut**
- **eap.ini**

To configure files for adding attributes to Access-Accept:

1. In the [Bootstrap] section of **simauth.aut** (for Oracle databases), set Enable=0.

Setting Enable=0 ensures that these files are disabled.

Example:

[Bootstrap]

Enable=0

2. Create a copy of **simauth.aut** and name it **simauth.eap**.

This renaming causes SIMAuth to become the EAP helper.

3. In the [Bootstrap] section of **simauth.eap**, ensure that Enable=1.

4. Open the relevant database access configuration file. This file is one of:

- **radsql.aut**
- **radsqljdbc.aut**
- **ldapauth.aut**

5. Check the [Bootstrap] section of **radsql.aut**, **radsqljdbc.aut**, or **ldapauth.aut** for the name of the specified authentication method. In the following example, the name of the specified authentication method is "SQLAuthenticator".

Example:

[Bootstrap]

Initializationstring=SQLAuthenticator

For more information about how to configure the **radsql.aut** and **radsqljdbc.aut** files, see ["SQL Authentication" on page 403](#) in this guide. For more information about configuring SQL authentication, see the *SBR Carrier Administration and Configuration Guide*.

For more information about how to configure the **ldapauth.aut** file, see ["LDAP Authentication" on page 444](#) in this guide. For more information about configuring LDAP authentication, see the *SBR Carrier Administration and Configuration Guide*.

6. Ensure that there is a section in the **eap.ini** file that includes the name of the *helped* authentication method you specified in Step 5. In this example the name is "SQLAuthenticator".

Example:

[SQLAuthenticator]

7. Ensure that the following lines are included in the helped authentication method section in **eap.ini** that you created in Step 6.

[SQLAuthenticator]

EAP-Only=1

First-Handle-Via-Auto-EAP=1

EAP-Type=SIM,AKA

Available-EAP-Only-Values=1

Available-Auto-EAP-Values=1

Available-EAP-Types=SIM|AKA

NOTE: The lines added in Step 7 configure the specified authentication method (in this case the SQL plug-in: **radsq1.aut** we named: *SQLAuthenticator*), and also prevent it from being used without the EAP helper (**SIMAuth.aut**). The use of the helped authentication method (**radsq1.aut** or **radsq1jdbc.aut**) without the EAP helper must be prevented because password checking is suppressed and the EAP helper (**SIMAuth.aut**) is needed to perform authentication.

8. Suppress database password checking in the *helped* authentication method as described for Oracle, JDBC, and LDAP databases.
 - Oracle or JDBC: Do not provide a password in the **SQL=SELECT** statement in the [Settings] section of **radsq1.aut** or **radsq1jdbc.aut**. In the [Results] section of these files, include a **PASSWORD=** statement, leaving the password blank.
Example:
[Results]
Password=
 - LDAP: Remove the %password= setting from the [Response] section.
9. Insert a query into **radsq1.aut**, **radsq1jdbc.aut**, or **ldapauth.aut** to select the attributes to be added to the Access-Accept.

The selection of attributes from the database can be based on the database key values for IMSI or MSISDN. The values for IMSI or MSISDN are added to the request by SIMAuth in the attributes 3GPP-IMSI or Funk-SS7-MSISDN so that they can be used in the database query.

Example:

SQL=SELECT subscriber-level FROM table 1 WHERE IMSI=@3GPP-IMSI

NOTE: To have the 3GPP-IMSI attribute set by Steel-Belted Radius Carrier in the request, the 3GPP dictionary must be selected in the **Make or model** field in the **RADIUS Clients List** page of Web GUI, or you must import the attribute using the @ character which indicates the dictionary file contents are to be included (see ["Include Records" on page 189](#)). You can also use the %username or %user variables in the database query. However, they do not contain the expected values if pseudonyms are active.

10. Activate the *helped* authentication method. For more information about setting up authentication policies and defining the order of authentication methods, see the *SBR Carrier Administration and Configuration Guide*.

Example Configuration for Adding Attributes to Access-Accept

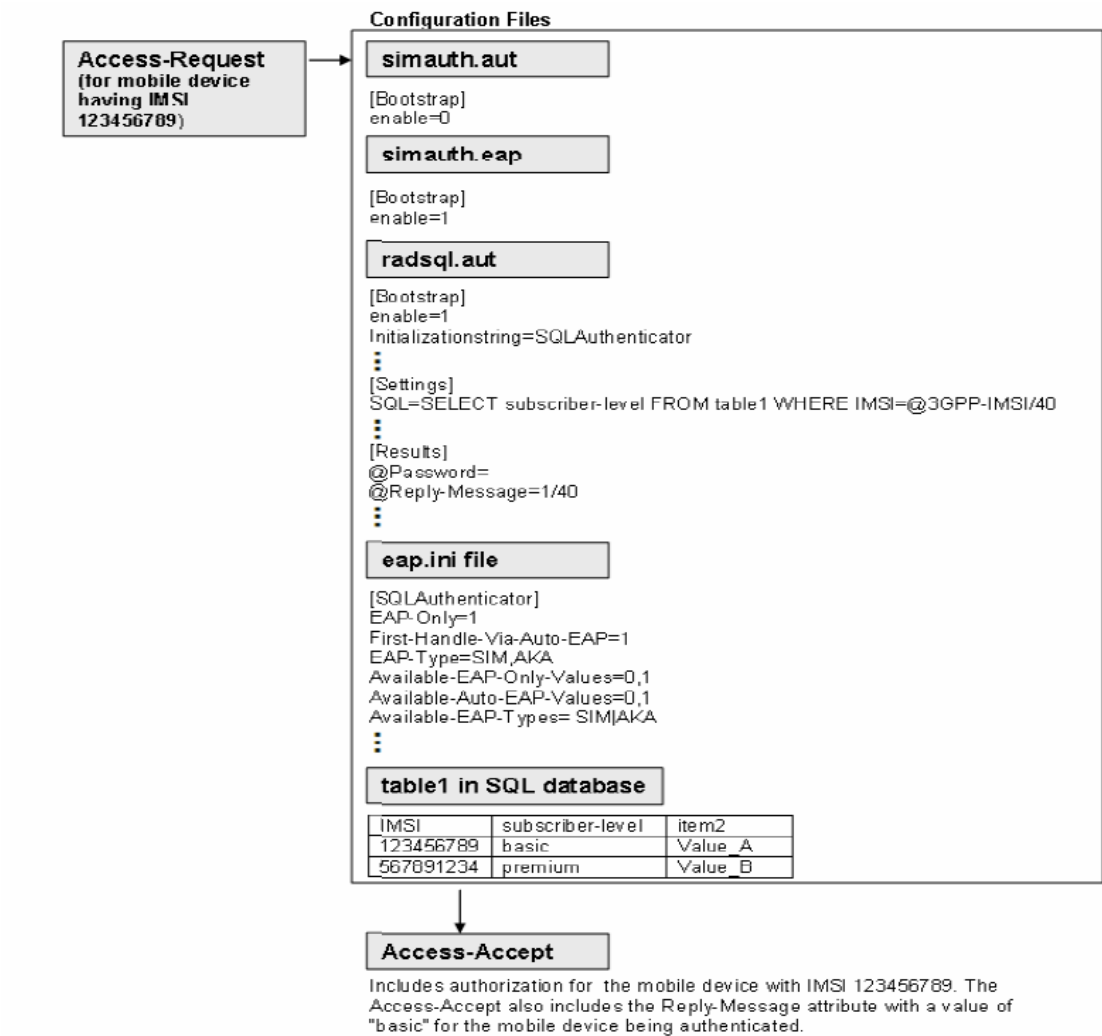
[Figure 22 on page 628](#) shows a sample configuration. The purpose of this configuration is to query the database for a subscriber-level value and return the subscriber-level value along with the Access-Accept.

Example Overview

In this example, an Access-Request is sent for a mobile device with IMSI 123456789. The value of the subscriber-level for this device is retrieved from the database, assigned to the attribute Reply-Message, and attached to the Access-Accept.

The configuration lines and syntax (shown in [Figure 22 on page 628](#)) associate all the configuration files together to attach an attribute to the Access-Accept.

Figure 22: Example Configuration for Adding Attributes to an Access-Accept



Example Notes

The sample configuration shown in [Figure 22 on page 628](#) configures the data flow in the following way:

Access-Request

An Access-Request is sent to Steel-Belted Radius Carrier for the user with an IMSI value of 123456789.

SIMAuth

simauth.eap file is enabled

simauth.aut file is disabled.

Radsql.aut

The [Bootstrap] section contains the name of the specified authentication method ("SQLAuthenticator"). You later add a [SQLAuthenticator] section to the **eap.ini** file.

Enter a SQL=SELECT statement to retrieve data from the database based on the value of the IMSI in the Access-Request. Do not include a password in the SQL SELECT statement.

The @Password= statement suppresses password checking of the database.

The @Reply-Message=1/40 field indicates the following:

- The Reply-Message attribute is added to the Access-Accept and carry the value retrieved from the database.
- The 1 in @Reply-Message=1/40 indicates that the first item in the SQL=SELECT statement (subscriber-level) is the column name of the SQL database from which the value is selected.
- The 40 in @Reply-Message=1/40 indicates that the width of the subscriber-level column is 40 characters.

Eap.ini

The **eap.ini** file must contain a section corresponding to the name of the *helped* authentication method named in the Initializationstring statement in the **radsql.aut** file. In this example the *helped* authentication method is called “SQLAuthenticator”, so the **eap.ini** must contain a section called [SQLAuthenticator].

The **eap.ini** file must contain the lines shown in [Figure 22 on page 628](#) to configure the SQL plug-in (either **radsql.aut** or **radsqljdbc.aut**). These lines, prevent either **radsql.aut** or **radsqljdbc.aut** from acting without **SIMAuth.aut**. This is necessary because password-checking by **radsql.aut** or **radsqljdbc.aut** is suppressed and the only authentication being performed would be by the EAP helper (**SIMAuth.aut**).

SQL Database Table 1

In this example, the SQL database is queried by the SQL plug-in: **radsql.aut**, and the subscriber-level for IMSI 123456789 is found to be **basic**.

Access-Accept

The value of **basic** is assigned to the attribute Reply-Message and included in the Access-Accept.

Activate the Authentication Method

For information about activating and setting up authentication policies, and defining the order of authentication methods, see the *SBR Carrier Administration and Configuration Guide*.

Interfacing with a Kineto IP Network Controller

IN THIS SECTION

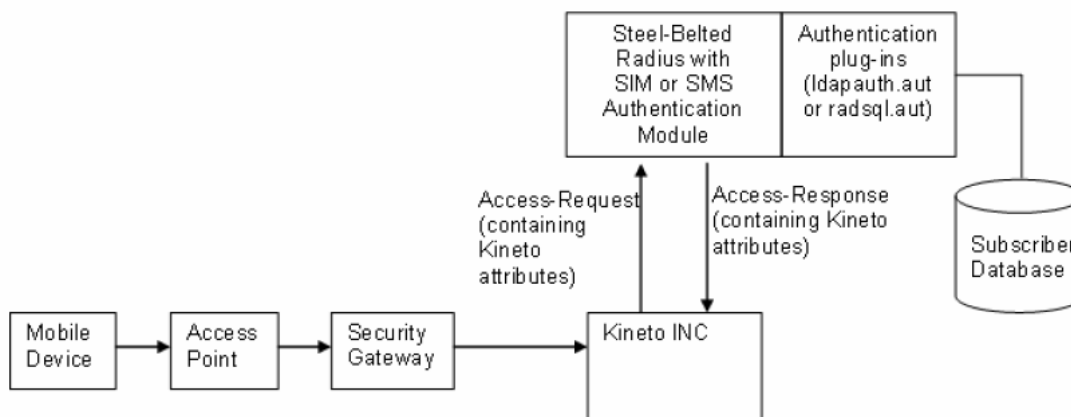
- [Attribute Handling Methods | 631](#)
- [Access-Request Conversion | 632](#)
- [Access-Accept Conversion | 639](#)
- [Access-Reject Conversion | 642](#)
- [Configuring the SIM Authentication Module for Handling Kineto Attributes | 642](#)

The Kineto IP Network Controller (INC) is the component of the Kineto UMA Network Controller (UNC) that manages subscriber access to voice and data mobile services. The Kineto INC-AAA (S1) interface Protocol Specification defines the interface requirements for communication between an AAA server and the Kineto INC. The following topics are included in this chapter:

The SIM authentication modules available with Steel-Belted Radius Carrier are designed to comply with the requirements for an AAA server. This chapter explains how these authentication modules interface with the Kineto INC and the tasks required for implementation.

[Figure 23 on page 630](#) illustrates the relationship between the Kineto INC and Steel-Belted Radius Carrier with one of the optional authentication modules.

Figure 23: Communication Between Kineto INC and Steel-Belted Radius Carrier



Attribute Handling Methods

The Kineto S1 Interface specification requires that certain Kineto vendor specific attributes (VSAs) be sent with Access-Requests, Access-Accepts, and Access-Rejects as defined in the following publicly available Kineto documents:

- INC-AAA (S1) Interface Protocol Specification Version 4 (Revision 0.00)
- UMA Service Access Control January 2006

The SIM authentication module available with Steel-Belted Radius Carrier are designed to comply with these specifications. Some of the Kineto VSAs contain structured attributes. Steel-Belted Radius Carrier can process these attributes in two different ways:

- Native subattribute method— Steel-Belted Radius Carrier provides native support for structured attributes which contain subattributes. This is the recommended method when using the Kineto INC. Subattributes, like normal RADIUS attribute-value pair (AVPs), consist of the raw encoding of a type field (such as 1 for WiMAX-Release, within the WiMAX-Capability VSA) followed by a length value (such as 5) followed by the value of the attribute (such as 1.2). Subattributes are values in a RADIUS packet that are not stored as a RADIUS AVP, or vendor-specific-attribute (VSA), but rather are packed with other subattributes into a RADIUS VSA. In a RADIUS packet, there may be multiple RADIUS VSAs that contain subattributes. The RADIUS VSA, which consists of multiple subattributes, is sometimes referred to as a *structured attribute* because it contains structured data.

Refer to structured attributes, using the “.” notation. For example:

“A.b.c”

Where attribute “A” is a group attribute containing a sequence subattribute “b”, which contains a simple attribute “c”.

Structured attributes are addressable only by their full *pathname*, which must include all interim group or sequence attributes.

You do not need to add every subattribute in a tree. Steel-Belted Radius Carrier handles partially defined trees by automatically supplying subattributes that have a default value specified in the **.jdict** definition files. Additionally, if a subattribute is added that does not have a suitable parent or group defined, Steel-Belted Radius Carrier automatically creates one.

For complete details on subattributes, see [“Structured Attributes” on page 199](#).

- Flattening/Unflattening method— An Access-Request is processed by *flattening* the structured attribute into a single-payload. The subattributes contained in the structured attribute are copied out of their containing AVPs and represented in the Access-Request as if they had been received as separate AVPs. In the returned Access-Accept or Access-Reject messages, the structured Kineto VSAs are reassembled or *unflattened* before returning them to the Kineto INC device. Before release 7.0, this was the only method supported in Steel-Belted Radius Carrier. This flattening/unflattening method has been deprecated, and it is recommended that you migrate to the native subattribute method.

Table 213 on page 632, Table 214 on page 639, and Table 215 on page 642 list the Kineto VSAs and their converted flattened/unflattened form, as well as their equivalent subattribute form. If you previously used the flatten/unflatten attribute conversion method, we recommend that you migrate to the subattribute method. All other Kineto attributes, not listed in the tables that follow, are passed between the Kineto INC and Steel-Belted Radius Carrier unchanged. If your processing requires handling of Kineto compound VSAs, use these conversion tables to identify the attributes to be passed between the Steel-Belted Radius Carrier AAA server and the Kineto INC.

Access-Request Conversion

Table 213 on page 632 lists the Kineto Access-Request compound attributes that are flattened by Steel-Belted Radius Carrier with either the SIM authentication module.

Table 213: Conversion from Kineto VSAs to Flattened Attributes

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UP-Client-Remote-Address	IPv4=8 IPv6=20	Vendor-Type=2 octets Vendor-Length=1 octet IP Address Type=1 octet IP Address: 4 octets for IPv4 16 octets for IPv6	Converted to one of the following two flattened attributes: Kineto-UP-Client-Remote-IPv4-Addr (type: ipaddr) or Kineto-UP-Client-Remote-IPv6-Addr (type: ipv6addr) Equivalent subattributes: Kineto-UP-Client-Remote-Address.IPv4-Address or Kineto-UP-Client-Remote-Address.IPv6-Address

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UP-Client-Public-Address	IPv4=8 IPv6=20	Vendor-Type=2 octets Vendor-Length=1 octet IP Address Type=1 octet IP Address: 4 octets for IPv4 16 octets for IPv6	Converted to one of the following two flattened attributes: Kineto-UP-Clnt-Public-IPv4-Addr (type: ipaddr) or Kineto-UP-Clnt-Public-IPv6-Addr (type: ipv6addr) Equivalent subattributes: Kineto-UP-Client-Public-Address.IPv4-Address or Kineto-UP-Client-Public-Address.IPv6-Address
Kineto-UMA-Classmark	2	Consists of multiple enumerated values within 2 octets.	Converted to all of the following four flattened attributes: Kineto-UMA-Classmark-TURA (type: integer) Kineto-UMA-Classmark-UC (type:integer) Kineto-UMA-Classmark-GC (type:integer) Kineto-UMA-Classmark-RRS (type:integer) Equivalent subattributes: Kineto-UMA-Classmark.TURA Kineto-UMA-Classmark.UC Kineto-UMA-Classmark.GC Kineto-UMA-Classmark.RTP-Redundancy-Supported

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UMA-AP-Radio-Identity	7	Discriminator octet (always 0x1) followed by 6-octet MAC address.	<p>Converted to the following flattened attribute:</p> <p>Kineto-UMA-AP-Radio-MAC (type:string)</p> <p>Equivalent subattributes:</p> <p>Kineto-UMA-AP-Radio-Identity.Value</p>
Kineto-UMA-MS-Radio-Identity	7	Discriminator octet (always 0x1) followed by 6-octet MAC address.	<p>Converted to the following flattened attribute:</p> <p>Kineto-UMA-MS-Radio-MAC (type:string)</p> <p>Equivalent subattributes:</p> <p>Kineto-UMA-MS-Radio-Identity.Value</p>

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UMA-Location-Area-ID	5 if LAC is not present; 7 if LAC is present	<p>Contains MCC (Mobile Country Code), MNC (Mobile Network Code) and LAC (Location Area Code).</p> <p>If the LAC is not present in the original VSA sent in the Access-Request, the attribute is not converted.</p> <p>The digits of MCC and MNC are encoded as BCD. MNC digit 3 may not be present, in which case its value is 0xF.</p> <p>See Figure 24 on page 639 for an illustration of the encoding.</p>	<p>Converted to the following flattened attributes:</p> <p>Kineto-UMA-Location-Area-MCC (type:string)</p> <p>Kineto-UMA-Location-Area-MNC (type:string)</p> <p>Kineto-UMA-Location-Area-LAC (type:string)</p> <p>Equivalent subattributes:</p> <p>NOTE: The MCC and MNC (3 digit numbers) in Kineto-UMA-Location-Area-MCC and Kineto-UMA-Location-Area-MNC are now available as three separate integer subattributes (.MCC1, .MCC2, and so on.).</p> <p>Kineto-UMA-Location-Area-ID.MCC1</p> <p>Kineto-UMA-Location-Area-ID.MCC2</p> <p>Kineto-UMA-Location-Area-ID.MCC3</p> <p>Kineto-UMA-Location-Area-ID.MNC1</p> <p>Kineto-UMA-Location-Area-ID.MNC2</p> <p>Kineto-UMA-Location-Area-ID.MNC3</p> <p>Kineto-UMA-Location-Area-ID.LAC</p>

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UMA-Geographical-Loc		<p>Two geographical location types can be generated:</p> <ul style="list-style-type: none"> • ellipsoid point (discriminator=0x00, length=7) for a latitude and longitude without any uncertainty. • ellipsoid point with uncertainty circle (discriminator=0x10, length=8) if Kineto-UMA-GeogLoc-Uncert-Circ is present. <p>For both geographical location types, a latitude and longitude are generated from Kineto-UMA-GeogLoc-Latitude and Kineto-UMA-GeogLoc-Longitude.</p>	<p>Converted to the following flattened attributes:</p> <p>Kineto-UMA-GeogLoc-Latitude</p> <p>Kineto-UMA-GeogLoc-Longitude</p> <p>Kineto-UMA-GeogLoc-Uncert-Circ (<i>optional</i>)</p> <p>Equivalent subattributes:</p> <p>NOTE: The subattribute equivalents for Kineto-UMA-Geographical-Loc have two possible structures: lat-long with uncertainty, or a lat-long alone.</p> <p>NOTE: The flattener method converts the latitude and longitude into decimal values to make them more readable and useful. The subattribute software does not process the attributes, it simply provides the Kineto-encoded values, which must be converted with reference to the Kineto specification if decimal degree values are desired.</p>
		<p>The latitude and longitude are encoded in complex ways. For more information, see</p>	<p>Kineto-UMA-Geographical-Loc.Elipsiod-Point.Latitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Latitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point.Longitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Longitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Uncertainty</p>

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
		<p>However, the converted latitude and longitude attribute formats must be encoded as ISO 6709 compliant string representations of decimal degrees, for example 48.0234.</p> <p>NOTE: Any number of decimal places for the degrees are accepted but the accuracy of the encoding depends on the format of the Kineto-UMA-Geographical-Loc attribute.</p>	
		<p>Directions are expressed:</p> <ul style="list-style-type: none"> • northern latitudes—positive numbers • southern latitudes—negative numbers • east longitudes—positive numbers • west longitudes—negative numbers 	

Table 213: Conversion from Kineto VSAs to Flattened Attributes (*continued*)

Kineto Attribute Name	Size in Octets	Format Description	Converted to: Flattened or Subattribute
Kineto-UMA-Cell-Identity	2	2-octet integer.	Before Release 7.0, the Kineto-UMA-Cell-Identity attribute was converted to a 4-octet integer, since Steel-Belted Radius Carrier did not handle 2-octet integers. In Steel-Belted Radius Carrier Release 7.0 and later, 2-octet integers are handled natively, so there is no need for using the flattening or subattribute method for this attribute.
Kineto-UMA-AP-Service-Name	Number of octets in the string plus one	<p>Consists of an octet discriminator followed by a string value of the PAN or SSID.</p> <p>PAN: Discriminator octet has a value of 0x02 and the PAN is the following string.</p> <p>IPv6: Discriminator octet has a value of 0x01 and the SSID is the following string.</p>	<p>Converted to the following flattened attributes:</p> <p>Kineto-UMA-AP-SSID (type:string)</p> <p>or</p> <p>Kineto-UMA-AP-PAN (type:string)</p> <p>Equivalent subattributes:</p> <p>Kineto-UMA-AP-Service-Name.SSID</p> <p>Kineto-UMA-AP-Service-Name.PAN</p>

Figure 24 on page 639 illustrates the encoding for the Kineto-UMA-Location-Area-Identification attribute. This attribute contains the MCC (Mobile Country Code), MNC (Mobile Network Code) and LAC (Location Area Code). For more information see “Kineto-UMA-Location-Area-ID” in Table 213 on page 632.

Figure 24: Format of Kineto-UMA-Location-Area-Identification Attribute

bits	8	7	6	5	4	3	2	1	
	MCC digit 2				MCC digit 1				octet 1
	MNC digit 3				MCC digit 3				octet 2
	MNC digit 2				MNC digit 1				octet 3
	LAC								octet 4
	LAC (continued)								octet 5

Access-Accept Conversion

Table 214 on page 639 describes the compound Kineto VSAs that are returned with Access-Accepts.

Table 214: Kineto Attributes Returned with the Access-Accepts

Converted From: Flattened or Subattributes	Returned Kineto Attribute (Unflattened)	Format Description
Converted from one of the following flattened attributes: Kineto-UMA-Service-Zone-Icon-Ind Kineto-UMA-Service-Zone-Name Subattribute equivalent: Kineto-UMA-Service-Zone-Info.Icon-Indicator Kineto-UMA-Service-Zone-Info.Name Kineto-UMA-Service-Zone-Info.Name-Length	Kineto-UMA-Service-Zone-Info	Consists of the Kineto-UMA-Service-Zone-Icon, followed by one octet containing the string length, followed by a string, extracted from Kineto-UMA-Service-Zone-Name.

Table 214: Kineto Attributes Returned with the Access-Accepts (*continued*)

Converted From: Flattened or Subattributes	Returned Kineto Attribute (Unflattened)	Format Description
<p>Converted from one of the following flattened attributes:</p> <p>Kineto-UMA-Location-Area-MCC Kineto-UMA-Location-Area-MNC Kineto-UMA-Location-Area-LAC</p> <p>Subattribute equivalent:</p> <p>NOTE: The MCC and MNC (3 digit numbers) in Kineto-UMA-Location-Area-MCC and Kineto-UMA-Location-Area-MNC are now available as three separate integer subattributes (.MCC1, .MCC2, and so on).</p> <p>Kineto-UMA-Location-Area-ID.MCC1 Kineto-UMA-Location-Area-ID.MCC2 Kineto-UMA-Location-Area-ID.MCC3 Kineto-UMA-Location-Area-ID.MNC1 Kineto-UMA-Location-Area-ID.MNC2 Kineto-UMA-Location-Area-ID.MNC3 Kineto-UMA-Location-Area-ID.LAC</p>	Kineto-UMA-Location-Area-ID	<p>Encoded values of Kineto-UMA-Location-Area-MCC, Kineto-UMA-Location-Area-MNC, and Kineto-UMA-Location-Area-LAC as shown in Figure 24 on page 639.</p>
<p>Converted from one of the following flattened attributes:</p> <p>Kineto-UMA-GeogLoc-Latitude Kineto-UMA-GeogLoc-Longitude Kineto-UMA-GeogLoc-Uncert-Circ(<i>optional</i>)</p> <p>Equivalent subattributes:</p> <p>NOTE: The subattribute equivalents for Kineto-UMA-Geographical-Loc have two possible structures: lat-long with uncertainty, or a lat-long alone.</p>	Kineto-UMA-Geographical-Loc	<p>Two geographical location types can be generated:</p> <ul style="list-style-type: none"> • ellipsoid point (discriminator=0x00, length=7) for a latitude and longitude without any uncertainty. • ellipsoid point with uncertainty circle (discriminator=0x10, length=8) if Kineto-UMA-GeogLoc-Uncert-Circ is present.

Table 214: Kineto Attributes Returned with the Access-Accepts (*continued*)

Converted From: Flattened or Subattributes	Returned Kineto Attribute (Unflattened)	Format Description
<p>NOTE: The flattener method converts the latitude and longitude into decimal values to make them more readable and useful. The subattribute software does not process the attributes; it simply provides the Kineto-encoded values, which must be converted with reference to the Kineto specification if decimal degree values are desired.</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point.Latitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Latitude</p>		<p>For both geographical location types, a latitude and longitude are generated from Kineto-UMA- GeogLoc-Latitude and Kineto-UMA -GeogLoc-Longitude.</p> <p>The latitude and longitude are encoded in complex ways. For more information, see 3GPP TS 23.032, <i>Sections 5 (Shapes), 6 (Coding), and 7 (General message format and information element)</i>. However, the converted latitude and longitude attribute formats must be encoded as ISO 6709 compliant string representations of decimal degrees DD.DDDD, for example 48.0234.</p>
<p>Kineto-UMA-Geographical-Loc.Elipsiod-Point.Longitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Longitude</p> <p>Kineto-UMA-Geographical-Loc.Elipsiod-Point-With-Uncertainty.Uncertainty</p>		<p>NOTE: Any number of decimal places for the degrees are accepted but the accuracy of the encoding depends on the format of the Kineto-UMA-Geographical-Loc attribute.</p> <p>Directions are expressed:</p> <ul style="list-style-type: none"> • northern latitudes—positive numbers • southern latitudes—negative numbers • east longitudes—positive numbers • west longitudes—negative numbers <p>For the second geographical type, the expected format of the uncertainty circle is a string that contains a decimal number of meters.</p> <p>For a full description of the uncertainty encoding and its forward translation, see 3GPP TS 23.032, <i>Section 6.2 (Uncertainty)</i>.</p>

Access-Reject Conversion

Table 215 on page 642 describes the Kineto compound VSA that is returned with an Access-Reject.

Table 215: Kineto Attributes Returned with the Access-Response

Converted From: Flattened or Subattributes	Returned Kineto Attribute	Format Description
<p>Converted from one of the following flattened attributes:</p> <p>Kineto-UMA-Service-Zone-Icon</p> <p>Kineto-UMA-Service-Zone-Name</p> <p>Equivalent subattributes:</p> <p>Kineto-UMA- Service -Zone-Info. Icon-Indicator</p> <p>Kineto-UMA-Service-Zone-Info.Name-Length</p>	Kineto-UMA-Service-Zone-Information	Consists of the Kineto-UMA-Service-Zone-Icon-Ind, followed by one octet containing the string length, followed by a string, extracted from Kineto-UMA-Service-Zone-Name.

Configuring the SIM Authentication Module for Handling Kineto Attributes

The following configuration activities are required to activate Kineto attribute handling:

- Configure the kinetoUMAAAttrHandler.ctrl file (*required only if using the flattening/unflattening attribute handling method*)
- Configure the controlpoints.ini file (*required only if using the flattening/unflattening attribute handling method*)
- Configure Steel-Belted Radius Carrier to recognize the Kineto attributes
- Develop applications for the S1 interface

Each of these configuration activities are described in the sections that follow.

Configuring the kinetoUMAAAttrHandler.ctrl File

The **kinetoUMAAAttrHandler.ctrl** file (located in the **Radius** directory) calls the appropriate library, enables use of the Kineto attribute handling features, and controls related settings.

NOTE: Configuration of the `kinetoUMAAttrHandler.ctrl` file is only required if using the flattening/unflattening attribute handling method. We recommend that you migrate to using the native subattribute handling method.

To configure the `kinetoUMAAttrHandler.ctrl` file:

1. Open the `kinetoUMAAttrHandler.ctrl` file located in the Radius directory.
2. In the [Bootstrap] section of the `kinetoUMAAttrHandler.ctrl` file, set `Enable=1`.
3. In the [Bootstrap] section of the `kinetoUMAAttrHandler.ctrl` file, make sure the following lines exist and are not commented out:

LibraryName=kinetoUMAAttrHandler.so
InitializationString= kinetoUMAAttrHandler

4. In the [Settings] section of the `kinetoUMAAttrHandler.ctrl` file, make sure the following line exists and is not commented out:

RemoveTranslatedAttributes=true

Example kinetoUMAAttrHandler.ctrl file

```
[Bootstrap]
Enable=1
LibraryName=kinetoUMAAttrHandler.so
InitializationString= kinetoUMAAttrHandler
[Settings]
RemoveTranslatedAttributes=true
```

[Table 216 on page 643](#) explains the settings required in the `kinetoUMAAttrHandler.ctrl` file to allow Kineto attribute handling.

Table 216: kinetoUMAAttrHandler.ctrl Fields

Section	Field	Description
[Bootstrap]	LibraryName	Specifies the name of the executable binary. Set to kinetoUMAAttrHandler.so
[Bootstrap]	Enable	<ul style="list-style-type: none"> • Set to 1 to enable this file. • Set to 0 to disable this file. Set to 1.

Table 216: kinetoUMAAAttrHandler.ctrl Fields (*continued*)

Section	Field	Description
[Bootstrap]	InitializationString	Specifies the name of the initialization file for the library. Set to kinetoUMAAAttrHandler
[Settings]	Remove Translated Attributes	<ul style="list-style-type: none"> • Set to true to enable this file. • Set to false to disable this file. Set to true.

Configuring the controlpoints.ini File

The **controlpoints.ini** file (located in the **Radius** directory) calls the attribute handler at the appropriate processing stages.

NOTE: Configuration of the **controlpoints.ini** file is only required if using the flattening/unflattening attribute handling method. We recommend that you migrate to using the native subattribute handling method.

To configure the **controlpoints.ini** file:

1. Open the **controlpoints.ini** file located in the **Radius** directory.
2. Enter the following lines in the file:

```
[Auth-Initial-Request]
kinetoUMAAAttrHandler
[Auth-Final-Request]
kinetoUMAAAttrHandler
```

[Table 217 on page 645](#) explains the settings required in the **controlpoints.ini** file to allow Kineto attribute handling.

Table 217: controlpoint.ini File Settings

Field	Description
[Auth-Initial-Request] section	<p>Calls the attribute handler plug-in when the initial authorization request is received.</p> <p>Add the field:</p> <p>kinetoUMAAAttrHandler</p>
[Auth-Final-Request] section	<p>Calls the attribute handler plug-in when authorization is complete.</p> <p>Add the field:</p> <p>kinetoUMAAAttrHandler</p>

Configuring Kineto Attribute Recognition

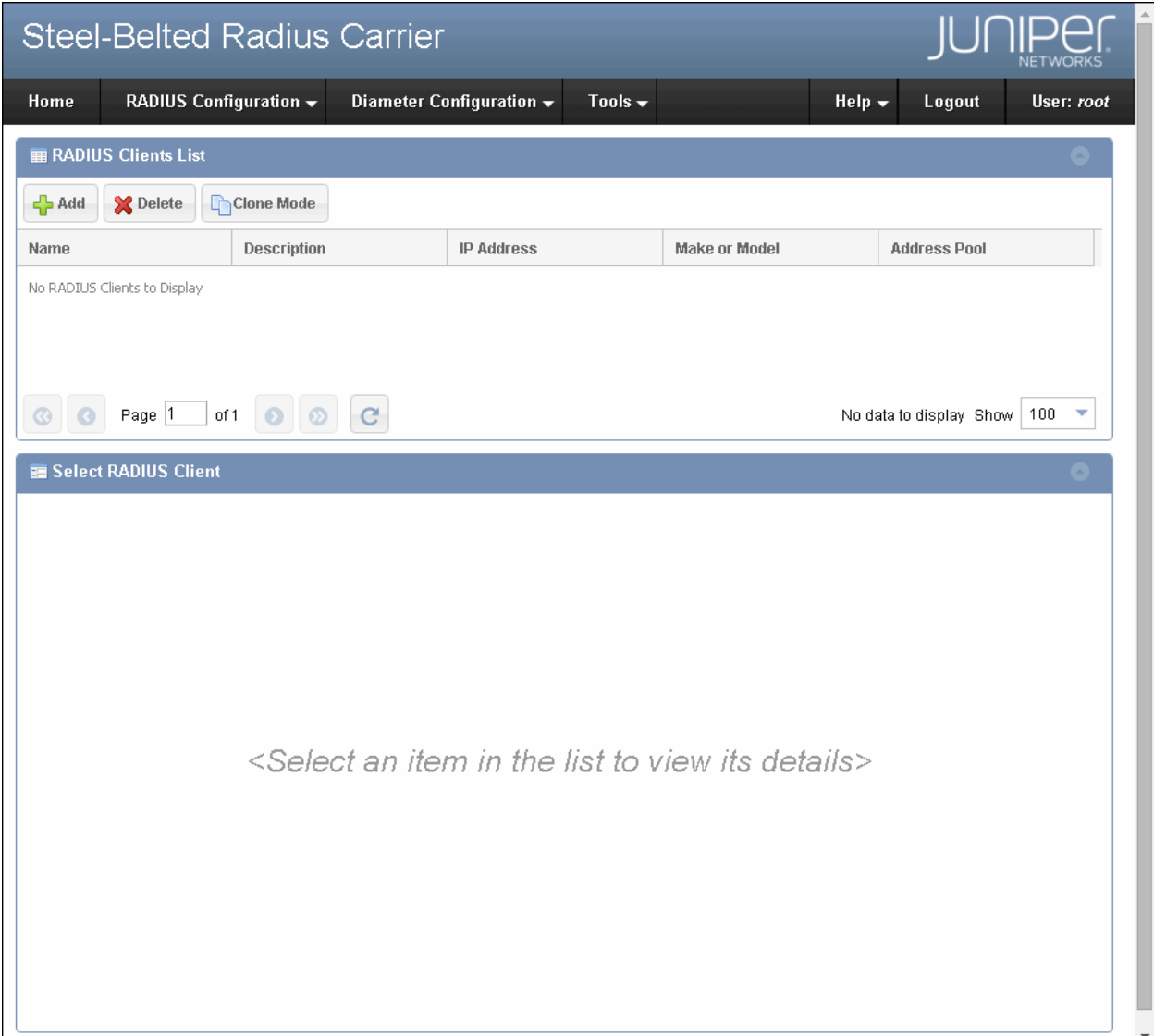
You must configure SBR Carrier to recognize the Kineto attributes by loading the Kineto dictionary file (.dct file).

To configure SBR Carrier to recognize the Kineto attributes you need to configure Kineto as a RADIUS client and activate the authentication method you want to use with Kineto using the Web GUI:

1. Launch the Web GUI and log in to your SBR Carrier server.
2. Select **RADIUS Configuration > RADIUS Clients**.

The **RADIUS Clients List** page ([Figure 25 on page 646](#)) appears.

Figure 25: RADIUS Clients List Page



3. Click **Add**.

The **Create RADIUS Client** pane (Figure 26 on page 647) appears with the **Basic Configuration** tab selected.

Figure 26: Create RADIUS Client Pane—Basic Configuration

The screenshot shows the Steel-Belted Radius Carrier web interface. The top navigation bar includes links for Home, RADIUS Configuration, Diameter Configuration, Tools, Help, Logout, and a user profile for 'root'. Below the navigation bar is a 'RADIUS Clients List' section. The main area is titled 'Create RADIUS Client' and contains four tabs: Basic Configuration, Profiles, Diameter Configuration, and Advanced Configuration. The 'Basic Configuration' tab is active, displaying a form with the following fields and options:

- Name:** A text input field.
- Description:** A text input field.
- IP Address:** A text input field.
- Range:** A checkbox followed by a numeric input field containing '1' and a small up/down arrow.
- Shared Secret:** A text input field with a 'Show' button to its right.
- Make or Model:** A dropdown menu currently showing '- Standard Radius -'.
- Address Pool:** A checkbox followed by a dropdown menu.
- Location Group:** A checkbox followed by a dropdown menu.

At the bottom of the form are three buttons: 'Save', 'Clear', and 'Cancel'.

4. Select **Kineto S1** from the **Make or Model** list and enter other details for your Kineto INC

NOTE: Selection of **Kineto S1** from the **Make or Model** list causes the Kineto dictionary file (.dct file) to be applied, which includes the Kineto attributes.

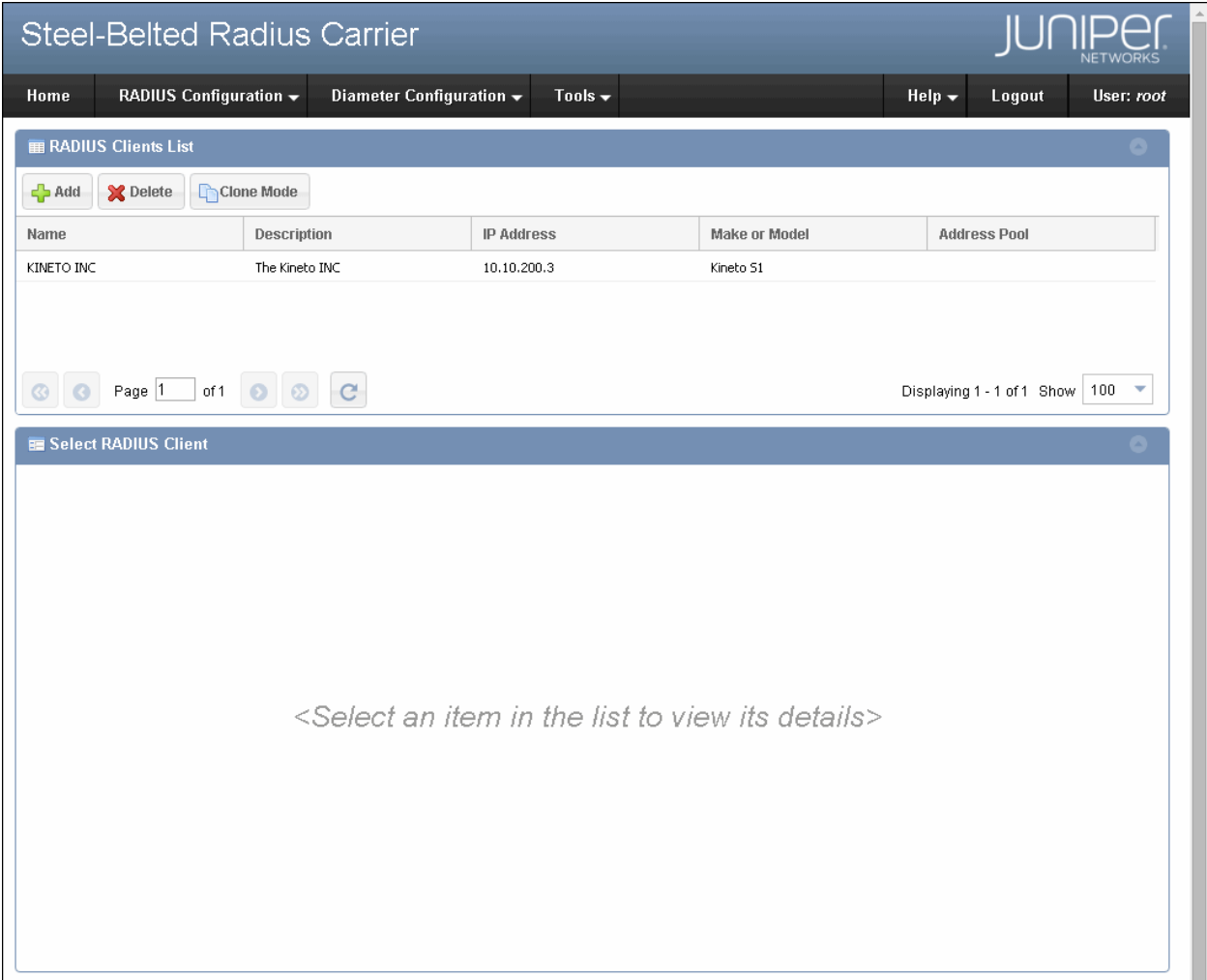
Figure 27: Selection of Kineto S1 from Make or Model List

The screenshot shows the Juniper Steel-Belted Radius Carrier configuration interface. The top navigation bar includes links for Home, RADIUS Configuration, Diameter Configuration, Tools, Help, Logout, and the user root. Below this is a tabbed interface for 'Create RADIUS Client' with tabs for Basic Configuration, Profiles, Diameter Configuration, and Advanced Configuration. The Basic Configuration tab is active, showing fields for Name (KINETO INC), Description (The Kineto INC), IP Address (10.10.200.3), Range (1), Shared Secret, and Make or Model. The Make or Model dropdown menu is open, displaying a list of device models including Kineto S1, which is highlighted. Other models listed include Iskratel Telecom Products, ITK NetBlazer, Juniper-ERX, Juniper-ERX 10.2, Juniper M/T Series, Juniper CTP Series, Kasten Chase Optiva, Lantronix LRS, Larscom 3000, LeeMah Bandwagon, Livingston PortMaster, and Meru Networks. Buttons for Save and Clear are visible at the bottom left of the form.

5. Click **Save** to save the changes.

The **RADIUS Clients List** page ([Figure 28 on page 649](#)) lists the **Kineto INC** entry.

Figure 28: RADIUS Clients List page with Kineto Entry



Activating the Authentication Method

To use either LDAP or SQL authentication, follow the procedures in the section on back-end authentication and accounting in the *SBR Carrier Administration and Configuration Guide* and see [“Back-End Authentication and Accounting” on page 385](#). To use the LDAP authentication method, you need to configure the **ldapauth.aut** file. To use the SQL authentication method you need to configure either the **radsqldb.aut** or **radsqldb.aut** file. After these files are configured, the respective authentication method becomes available to activate in Web GUI.

To activate the authentication method using the Web GUI:

1. Select **RADIUS Configuration > Authentication Policies > Order of Methods**.

The **Authentication Methods** page [Figure 29 on page 650](#) displays any configured authentication methods in the server. The **Inactive Authentication Methods** area displays a list of inactive authentication methods, while the **Active Authentication Methods** area displays a list of active authentication methods.

Figure 29: Authentication Methods Page



2. Select the authentication method from the **Inactive Authentication Methods** area and click the **Right** arrow.

You can also activate the authentication method by dragging the method to the right.

NOTE: The name of the LDAP or SQL authentication method is specified in the **InitializationString** parameter of the **.aut** file. In the example shown in [Figure 29 on page 650](#), the **MYSQL_JDBC** was defined in the **radsqljdbc.aut**. The authentication method does not appear in the list of authentication methods until you configure the associated **.aut** file.

3. To change the order in which the server tries authentication methods, select each authentication method and click the **Up** or **Down** arrow until the authentication methods are in the desired order.
4. Click **Save** to save the changes.

Developing Applications for the S1 Interface

To implement the Kineto S1interface with the authentication modules, you must:

- Write your application using SQL stored procedures or LDAP scripting to conform with the requirements in the Kineto S1 interface specification.
- Configure and enable the **ldapauth.aut**, or **radsql.aut** or **radsqljdbc.aut** to authenticate subscribers using data stored in an LDAP directory or an SQL database.

For more information about SQL stored procedures, LDAP scripting, the LDAP authentication plug-in: **ldapauth.aut**, or the SQL plug-ins: **radsql.aut** or **radsqljdbc.aut**, see [“Back-End Authentication and Accounting” on page 385](#) in this guide. Also see information about creating LDAP scripts and back-end authentication and accounting methods in the *SBR Carrier Administration and Configuration Guide*.

Using Managed IPv6 Address Pools

You can configure the **UsePools** and **Pools-IPv6-Prefix-Offset** parameters to use IPv4 address pools configured through the Web GUI for allocating IPv6 address to the client. The parameters are available in the **[IPv6]** section of the **radius.ini** file.

```
UsePools = [ "No" | "IPv4" ]
Pools-IPv6-Prefix-Offset = offset
```

The **UsePools** parameter enables the usage of IPv4 address pools to allocate IPv6 addresses to the client, and the **Pools-IPv6-Prefix-Offset** parameter specifies an offset. SBR Carrier embeds the address received

from the IPv4 address pool into the Framed-IPv6-Prefix attribute at the specified offset. The offset is specified in bits and ranges from 0 through 96. The offset must be a multiple of 8.

In addition to configuring IPv4 address pools through the Web GUI, a Framed-IPv6-Prefix attribute must be added to the reply list (for example, the attribute can be added to the list through a profile). When a Framed-IPv6-Prefix attribute is encountered in the reply list, a Framed-IP-Address attribute is in the reply list with the value set to a pool name, and the **UsePools** parameter is set to IPv4, SBR Carrier performs the following actions:

- Allocates an IPv4 address from the named pool, as is currently done for Framed-IP-Address attributes.
- Embeds the returned 4-byte address in the Framed-IPv6-Prefix attribute at the offset specified by the **Pools-IPv6-Prefix-Offset** parameter.
- Creates a Framed-IP-Address attribute for internal use in the SSR and for pool and session management purposes, but does not send the attribute on the wire.

The Framed-IPv6-Prefix attribute includes the portion where IPv4 addresses is embedded. For example, **Framed-IPv6-Prefix = fd85:4938::/64**, where /64 indicates a 64-bit prefix and :: indicates that the last 32 bits are zero.

By default, the dynamically assigned IPv4 address is embedded in the last 32 bits of the prefix (as if one had explicitly configured an optional offset value of 32 in this particular case). For example, **Framed-IPv6-Prefix = fd85:4938.a.b.c.d/64**, where *a.b.c.d* represents the IPv4 address.

If the assigned IPv4 address is 10.20.30.40, the actual hexadecimal string value of the generated RADIUS Framed-IPv6-Prefix attribute would be:

```
61 14 00 40 FD85 4938 0A14 1E28 0000 0000 0000 0000
```

where:

- 61 is the attribute type that identifies Framed-IPv6-Prefix
- 14 is the length of the attribute (20 octets)
- 00 is a reserved octet (must be zero)
- 40 is the bit length of the prefix (64 bits)
- FD85 4938 is the leading 32 bits of the prefix (as configured)
- 0A14 1E28 is the trailing 32 bits of the prefix (overwritten by the IPv4 address that is obtained from a pool)
- 00s are automatically generated padding

The offset value indicates the number of most significant bits of the prefix that will be skipped in order to determine the location of the embedded IPv4 addresses. For example, if **Pools-IPv6-Prefix-Offset =16**

and the assigned IPv4 address is 10.20.30.40, then the actual hexadecimal string value of the generated RADIUS Framed-IPv6-Prefix attribute would be:

```
61 14 00 40 FD85 0A14 1E28 0000 0000 0000 0000 0000
```

NOTE: Some non-zero bits (hex value 4938) of the original prefix are overwritten. This is not necessarily an error depending on the actual interpretation of the original prefix.

The default value for the **Pools-IPv6-Prefix-Offset** parameter is unspecified. That is, the dynamically assigned IPv4 addresses are embedded in the last 32 bits of the prefix. For example, if the Framed-IPv6-Prefix attribute is `::FFFF:0:0/128`, the offset would be set to 96 in this case resulting in IPv6 mapped addresses:

```
::ffff:a.b.c.d/128
```

If the assigned IPv4 address is 10.20.30.40, the actual hexadecimal string value of the generated RADIUS Framed-IPv6-Prefix attribute would be:

```
61 14 00 80 0000 0000 0000 0000 0000 FFFF 0A14 1E28
```

The default offset is a function of the original prefix length. The default offset is always such that the dynamically assigned IPv4 addresses will be embedded in the last 32 bits of the prefix. That is, the software must calculate the default offset using the following formula:

default offset = original prefix length - 32

For example, if you have an all-IPv6 network, only IPv6 addresses can be used. By using the default value of **Pools-IPv6-Prefix**, this feature enables the NAD to return an IPv6 address to the router, which in turn converts to IPv4 for assignment to a device which only supports IPv4.

Another use case is the emulation of IPv6 global unicast addresses by using the **Pools-IPv6-Prefix-Offset** parameter. The format for this type of IPv6 address is a 3-bit header of 001, followed by 45 bits of global routing prefix and 16 bits of subnet ID. The bit length of this type of Framed-IPv6-Prefix is always 64.

As an example, if the assigned global routing prefix for a customer is **1D854948**, one would set the leading 3-bit header to indicate an IPv6 global unicast address (the binary representation of the first hex digit 0001 becomes 0011) and the Framed-IPv6-Prefix attribute is configured as **Framed-IPv6-Prefix = 3D85:4948::/64**. If the IPv4 address 10.20.30.40 is obtained from a pool, then the following Framed-IPv6-Prefix attribute would be generated on the wire:

```
3D85:4948:0A14:1E28/64
```

or

```
61 14 00 40 3D85 4948 0A14 1E28 0000 0000 0000 0000
```

APPENDIX E

SIR.conf File

IN THIS SECTION

- [SBR_INFO] Section | 655
- [Level_One] Section | 656
- [Level_Two] Section | 663

The **SIR.conf** file is used by the **SIR.sh** script, which you can execute to collect system information requested by JTAC when there is a problem. For more information about the **SIR.sh** script, see *SBR Carrier Administration and Configuration Guide*.

NOTE: Make sure that **pkgapp**, **pkgcore.sh**, **SIR.sh**, and **SIR.conf** files are present in the same directory.

This appendix contains the following sections:

[SBR_INFO] Section

The [SBR_INFO] section ([Table 218 on page 656](#)) of the **SIR.conf** file specifies the directory where SBR Carrier is installed.

Table 218: SIR.conf [SBR_INFO] Fields

Parameter	Function
RADIUS_PRIVATE_DIR	Specifies the absolute path where SBR Carrier is installed. Default path is <code>/opt/JNPRsbr/radius</code> .

[Level_One] Section

You can use the [Level_One] section ([Table 219 on page 656](#)) to specify the first level information such as SBR configuration information, process information, system information, and package and patch information to be collected by the **SIR.sh** script. The collected information is stored in a tar file named **Level1**, under the directory named **SBR_Information_Report**.

Table 219: SIR.conf [Level_One] Fields

Parameter	Function
Enable	If set to 1, enables the collection of first level information. If set to 0, disables the collection of first level information. Default value is 1.

[SBR_Configuration_Files] Section

The [SBR_Configuration_Files] section specifies the text-based configuration files to be collected by the **SIR.sh** script. The collected files are stored in a directory named **Text_Configurations**, in the **Level1** tar file.

This section contains the **acc**, **aut**, **cnf**, **conf**, **ctrl**, **dhc**, **dir**, **gen**, **ini**, **jsi**, **pro**, **rr**, and **ses** parameters that can be used to specify the types of configuration files to be collected by the **SIR.sh** script. You can set the parameters either to 1 to enable the collection of corresponding text-based configuration files or to 0 to disable the collection of files. By default, all the parameters are set to 1.

[SBR_GUI_Configuration] Section

The [SBR_GUI_Configuration] section ([Table 220 on page 657](#)) specifies the GUI configuration files to be collected by the **SIR.sh** script. The collected files are stored in a directory named **GUI_Configurations**, in the **Level1** tar file.

Table 220: SIR.conf [SBR_GUI_Configuration] Fields

Parameter	Function
radiusdata	<p>If set to 1, enables the collection of radiusdata.* files.</p> <p>If set to 0, disables the collection of radiusdata.* files.</p> <p>Default value is 1.</p>
xdb	<p>If set to 1, enables the collection of *.xdb files.</p> <p>If set to 0, disables the collection of *.xdb files.</p> <p>Default value is 1.</p>

[SBR_Package_Information] Section

The [SBR_Package_Information] section ([Table 221 on page 657](#)) specifies the files that contains information related to SBR package versions, applied patches, and license details to be collected by the **SIR.sh** script. The collected files are stored in a directory named **SBR_Package_Patch_Info**, in the **Level1** tar file.

Table 221: SIR.conf [SBR_Package_Information] Fields

Parameter	Function
dat	<p>If set to 1, enables the collection of *.dat files (such as package.dat and configure.dat).</p> <p>If set to 0, disables the collection of *.dat files.</p> <p>Default value is 1.</p>
lic	<p>If set to 1, enables the collection of radius.lic file.</p> <p>If set to 0, disables the collection of radius.lic file.</p> <p>Default value is 1.</p>

Table 221: SIR.conf [SBR_Package_Information] Fields (*continued*)

Parameter	Function
patch_log	<p>If set to 1, enables the collection of patch log files.</p> <p>If set to 0, disables the collection of patch log files.</p> <p>Default value is 1.</p>

[SBR_Dictionary_Information] Section

The [SBR_Dictionary_Information] section ([Table 222 on page 658](#)) specifies the dictionary files to be collected by the **SIR.sh** script. The collected dictionary files are stored in a directory named **SBR_Dictionaries**, in the **Level1** tar file.

Table 222: SIR.conf [SBR_Dictionary_Information] Fields

Parameter	Function
dci	<p>If set to 1, enables the collection of *.dci files.</p> <p>If set to 0, disables the collection of *.dci files.</p> <p>Default value is 1.</p>
dct	<p>If set to 1, enables the collection of *.dct files.</p> <p>If set to 0, disables the collection of *.dct files.</p> <p>Default value is 1.</p>
dic	<p>If set to 1, enables the collection of *.dic files.</p> <p>If set to 0, disables the collection of *.dic files.</p> <p>Default value is 1.</p>
jdct	<p>If set to 1, enables the collection of *.jdct files.</p> <p>If set to 0, disables the collection of *.jdct files.</p> <p>Default value is 1.</p>

[Certificate_JAR_Files] Section

The [Certificate_JAR_Files] section ([Table 223 on page 659](#)) specifies the certificates and Java ARchive files to be collected by the **SIR.sh** script. The collected files are stored in a directory named **Certs_and_Jars**, in the **Level1** tar file.

Table 223: SIR.conf [Certificate_JAR_Files] Fields

Parameter	Function
cer	<p>If set to 1, enables the collection of *.cer files.</p> <p>If set to 0, disables the collection of *.cer files.</p> <p>Default value is 1.</p>
jar	<p>If set to 1, enables the collection of *.jar files.</p> <p>If set to 0, disables the collection of *.jar files.</p> <p>Default value is 1.</p>
pfx	<p>If set to 1, enables the collection of *.pfx files.</p> <p>If set to 0, disables the collection of *.pfx files.</p> <p>Default value is 1.</p>

[Radius_Process_Info] Section

The [Radius_Process_Info] section ([Table 224 on page 659](#)) specifies the RADIUS process specific information to be collected by the **SIR.sh** script. The collected information is stored in the files under a directory named **Radius_Process_Info**, in the **Level1** tar file.

Table 224: SIR.conf [Radius_Process_Info] Fields

Parameter	Function
cpu_usage	<p>If set to 1, enables collecting CPU usage details of RADIUS process.</p> <p>If set to 0, disables collecting CPU usage details of RADIUS process.</p> <p>Default value is 1.</p>

Table 224: SIR.conf [Radius_Process_Info] Fields (*continued*)

Parameter	Function
isof_or_pfiles	<p>If set to 1, enables collecting the list of files opened by the RADIUS process.</p> <p>If set to 0, disables collecting the list of files opened by the RADIUS process.</p> <p>Default value is 1.</p>
memory	<p>If set to 1, enables collecting memory usage details of RADIUS process.</p> <p>If set to 0, disables collecting memory usage details of RADIUS process.</p> <p>Default value is 1.</p>
no_of_threads	<p>If set to 1, enables the collection of information about the number of threads running in the RADIUS process.</p> <p>If set to 0, disables the collection of information about the number of threads running in the RADIUS process.</p> <p>Default value is 1.</p>
pmap	<p>If set to 1, enables collecting memory map details of RADIUS process.</p> <p>If set to 0, disables collecting memory map details of RADIUS process.</p> <p>Default value is 1.</p>
pstack	<p>If set to 1, enables collecting stack usage details of RADIUS process.</p> <p>If set to 0, disables collecting stack usage details of RADIUS process.</p> <p>Default value is 1.</p>
status_verbose	<p>If set to 1, enables collecting the output of <code>./sbrd status -v</code> command.</p> <p>If set to 0, disables collecting the output of <code>./sbrd status -v</code> command.</p> <p>Default value is 1.</p>

[SBR_XML] Section

The [SBR_XML] section ([Table 225 on page 661](#)) specifies the XML configuration files to be collected by the **SIR.sh** script. You can specify any **.xml** files in the [SBR_XML] section other than the files listed in

[Table 225 on page 661](#). To collect the particular **.xml** file, specify the file name and set it to 1. The collected files are stored in a directory named **SBR_XML**, in the **Level1** tar file.

Table 225: SIR.conf [SBR_XML] Fields

Parameter	Function
administration.xml	<p>If set to 1, enables the collection of administration.xml file.</p> <p>If set to 0, disables the collection of administration.xml file.</p> <p>Default value is 1.</p>
deviceModels.xml	<p>If set to 1, enables the collection of deviceModels.xml file.</p> <p>If set to 0, disables the collection of deviceModels.xml file.</p> <p>Default value is 1.</p>

[System_Info] Section

The [System_Info] section ([Table 226 on page 661](#)) specifies the system specific information to be collected by the **SIR.sh** script. The collected information is stored in the files under a directory named **System_Info**, in the **Level1** tar file.

Table 226: SIR.conf [System_Info] Fields

Parameter	Function
Disk_Usage	<p>If set to 1, enables collecting system disk space usage details.</p> <p>If set to 0, disables collecting system disk space usage details.</p> <p>Default value is 1.</p>
Environment	<p>If set to 1, enables collecting the environment variables set in the system.</p> <p>If set to 0, disables collecting the environment variables set in the system.</p> <p>Default value is 1.</p>
Hardware	<p>If set to 1, enables collecting system hardware details.</p> <p>If set to 0, disables collecting system hardware details.</p> <p>Default value is 1.</p>

Table 226: SIR.conf [System_Info] Fields (*continued*)

Parameter	Function
lostat	<p>If set to 1, enables collecting I/O statistics such as terminal, disk, and tape I/O activities.</p> <p>If set to 0, disables collecting I/O statistics.</p> <p>Default value is 1.</p>
Netstat	<p>If set to 1, enables the collection of information about listening and non-listening sockets.</p> <p>If set to 0, disables the collection of information about listening and non-listening sockets.</p> <p>Default value is 1.</p>
Opened_File	<p>If set to 1, enables collecting the list of open files in the system.</p> <p>If set to 0, disables collecting the list of open files in the system.</p> <p>Default value is 1.</p>
OpenldapVersion	<p>If set to 1, enables collecting OpenLDAP version details.</p> <p>If set to 0, disables collecting OpenLDAP version details.</p> <p>Default value is 1.</p>
OpensslVersion	<p>If set to 1, enables collecting OpenSSL version details.</p> <p>If set to 0, disables collecting OpenSSL version details.</p> <p>Default value is 1.</p>
OS_Version	<p>If set to 1, enables collecting OS type and version details.</p> <p>If set to 0, disables collecting OS type and version details.</p> <p>Default value is 1.</p>
RAM_Memory	<p>If set to 1, enables collecting system memory details.</p> <p>If set to 0, disables collecting system memory details.</p> <p>Default value is 1.</p>

Table 226: SIR.conf [System_Info] Fields (*continued*)

Parameter	Function
Running_Process	<p>If set to 1, enables the collection of information about all the processes running in the system.</p> <p>If set to 0, disables the collection of information about all the processes running in the system.</p> <p>Default value is 1.</p>
Ulimit	<p>If set to 1, enables collecting limit values set in the system.</p> <p>If set to 0, disables collecting limit values set in the system.</p> <p>Default value is 1.</p>
Uname	<p>If set to 1, enables collecting the system name.</p> <p>If set to 0, disables collecting the system name.</p> <p>Default value is 1.</p>

[Level_Two] Section

The [Level_Two] section ([Table 227 on page 663](#)) of the **SIR.conf** file specifies the core and log files to be collected by the **SIR.sh** script. The collected information is stored in a tar file named **Level2**, under the directory named **SBR_Information_Report**.

Table 227: SIR.conf [Level_Two] Fields

Parameter	Function
Enable	<p>If set to 1, enables the collection of core and log files.</p> <p>If set to 0, disables the collection of core and log files.</p> <p>Default value is 0.</p>

[Log_Files] Section

The [Log_Files] section ([Table 228 on page 664](#)) specifies the log files to be collected by the **SIR.sh** script. The collected files are stored in a directory named **Logs**, in the **Level2** tar file.

Table 228: SIR.conf [Log_Files] Fields

Parameter	Function
AcctLog	<p>If set to 1, enables the collection of accounting log files.</p> <p>If set to 0, disables the collection of accounting log files.</p> <p>Default value is 1.</p>
AcctReport	<p>If set to 1, enables the collection of accounting report log files.</p> <p>If set to 0, disables the collection of accounting report log files.</p> <p>Default value is 1.</p>
AuthLog	<p>If set to 1, enables the collection of authentication log files.</p> <p>If set to 0, disables the collection of authentication log files.</p> <p>Default value is 1.</p>
AuthReport	<p>If set to 1, enables the collection of authentication report log files.</p> <p>If set to 0, disables the collection of authentication report log files.</p> <p>Default value is 1.</p>
Configure	<p>If set to 1, enables the collection of configure.log file.</p> <p>If set to 0, disables the collection of configure.log file.</p> <p>Default value is 1.</p>
Date	Collects the log files generated by SBR Carrier on the specified date. The date must be in the DD-MM-YYYY format.
Latency	<p>If set to 1, enables the collection of latency log files.</p> <p>If set to 0, disables the collection of latency log files.</p> <p>Default value is 1.</p>

Table 228: SIR.conf [Log_Files] Fields (*continued*)

Parameter	Function
SBR	<p>If set to 1, enables the collection of RADIUS log files.</p> <p>If set to 0, disables the collection of RADIUS log files.</p> <p>Default value is 1.</p>
SNMP	<p>If set to 1, enables the collection of SNMP log files.</p> <p>If set to 0, disables the collection of SNMP log files.</p> <p>Default value is 1.</p>
Statlog	<p>If set to 1, enables the collection of statistics log files.</p> <p>If set to 0, disables the collection of statistics log files.</p> <p>Default value is 1.</p>

[Core_Files] Section

The [Core_Files] section ([Table 229 on page 665](#)) specifies the core files to be collected by the **SIR.sh** script. The collected files are stored in a directory named **Core**, in the **Level2** tar file.

Table 229: SIR.conf [Core_Files] Fields

Parameter	Function
radius	<p>If set to 1, enables the collection of RADIUS core file.</p> <p>If set to 0, disables the collection of RADIUS core file.</p> <p>When you execute the SIR.sh script, you must define the specific core file (with absolute path) to be collected by using the -r option.</p> <p>Default value is 1.</p>
snmp	<p>If set to 1, enables the collection of SNMP core file.</p> <p>If set to 0, disables the collection of SNMP core file.</p> <p>When you execute the SIR.sh script, you must define the specific core file (with absolute path) to be collected by using the -s option.</p> <p>Default value is 0.</p>

Glossary

Numerics

3GPP	Third generation Partnership Project (GSM).
3GPP2	Third generation Partnership Project 2 (CDMA).
802.1X	IEEE standard 802.1X. Standard for Local and Metropolitan Area Networks-Port-Based Network Access Control. Defines a mechanism that allows a supplicant (client) to connect to a wireless access point or wired switch so that the supplicant can provide authentication credentials that can be verified by an authentication server.

A

AAA	Authentication, Authorization, and Accounting.
AC	Access Controller.
accounting	The process of recording and aggregating resource use statistics and log files for a user, connection session, or function for billing, system diagnosis, and usage planning.
ACL	Access Control List.
agent	SNMP module on a managed device that responds to requests from a management station and sends traps to one or more recipients (trap sinks) to inform administrators of potential problems.
AKA	Authentication and Key Agreement. An extension to the EAP protocol that enables authentication and session key distribution using a mechanism based on symmetric keys and usually runs on a USIM.
AP	Access Point.
APN	Access Point Name.
attribute	RADIUS attributes that carry specific authentication, authorization, and accounting messages.
AuC	Authentication Center. The network element that provides the triplets for authenticating the subscriber.
authentication	The process of verifying the identity of a device and its user. This process is accomplished through transmission of identifying data at the time of connection.
Authentication and Key Agreement	See AKA.

authentication server A back-end server that verifies, from the credentials provided by an access client, whether the access client is authorized to use network resources.

authorization The process of controlling the access settings, such as privileges and time limits, that the user can exercise on a protected network.

autonomous server A Steel-Belted Radius Carrier server that does not use centralized configuration management.

AVP Attribute Value Pair. An attribute and its corresponding value; for example, **User-Name = admin**.

B

balun Balanced/unbalanced converter. A device used to match impedance between balanced and unbalanced lines, usually twisted-pair and coaxial cable.

BAOC Barring of All Outgoing Calls.

blocklist A profile of checklist attributes that cause Steel-Belted Radius Carrier to reject an authentication request. For example, a blocklist profile might specify calling station phone numbers or IP addresses that are blocked by Steel-Belted Radius Carrier.

BS Base Station.

C

CA Certificate Authority. A trusted entity that registers the digital identity of a site or individual and issues a digital certificate that guarantees the binding between the identity and the data items in a certificate.

CCB Customer Care and Billing system.

CCM Centralized Configuration Management. The process by which configuration information is shared between a primary RADIUS server and one or more replica RADIUS servers so that all machines operate in a similar way.

CDF Charging Data Function.

CDR Call Detail Record. Call transaction record created by an MSC to track the network resources used by subscribers in making and receiving calls, so that billing systems can compute charges based on usage.

certificate A digital file signed by a CA that guarantees the binding between an identity and the contents of the certificate.

CG Charging Gateway. Device that collects, validates, and consolidates CDRs from other network components for processing by the network billing system.

Change of Authorization See CoA.

CHAP	Challenge Handshake Authentication Protocol. An authentication protocol where a server sends a challenge to a requestor after a link has been established. The requestor responds with a value obtained by executing a hash function. The server verifies the response by calculating its own hash value. If the two hash values match, the authentication is acknowledged.
checklist	A list of attributes that must accompany a request for connection before the connection request can be authenticated.
CoA	Change of Authorization. Refers to RADIUS Change of Authorization, which is the dynamic change of the state of a previously authorized session by use of a RADIUS request sent towards the access equipment.
community	A group of devices and management stations running SNMP. An SNMP device or agent may belong to more than one SNMP community.
community string	Character string included in SNMP messages to identify valid sources for SNMP requests and to limit access to authorized devices. The read community string allows an SNMP management station to issue Get and GetNext messages. The write community string allows an SNMP management station to issue Set messages.
credentials	Data that is verified when presented to an authenticator, such as a password or a digital certificate.
CRL	Certificate Revocation List. A data structure that identifies the digital certificates that have been invalidated by the certificates' issuing CA before their expiration date.
CSCF	Call Session Control Function.

D

daemon	A program on a UNIX or Linux host that runs continuously to handle service requests.
DHCP	Dynamic Host Configuration Protocol. Protocol by which a server automatically assigns (leases) a network address and other configuration settings to a client temporarily or permanently.
dictionary	Text file that maps the attribute/value pairs supported by third-party RADIUS vendors.
Disconnect Message	See DM.
DM	Disconnect Message. Refers to RADIUS Disconnect, which is the dynamic termination of a previously authorized session by use of a RADIUS request sent towards the access equipment.
DNIS	Dialed Number Identification Service. A telephone service that identifies what number was dialed by a caller.

DNS Domain Name Service. Internet protocol for mapping hostnames, domain names, and aliases to IP addresses.

E

EAP Extensible Authentication Protocol. An industry-standard authentication protocol for network access that acts as a transport for multiple authentication methods or types. Defined by RFC 2284. The base protocol used for a variety of authentication methods with Radius and 802.1X.

EAP-AKA EAP method that allows authentication with a mobile subscriber USIM card.

EAP-SIM EAP method that allows authentication with a mobile subscriber SIM card.

EAP-TLS Authentication method that uses EAP (Extensible Authentication Protocol) and TLS (Transport Layer Security).

EAP-TTLS Authentication method that uses EAP (Extensible Authentication Protocol) and TTLS (Tunneled Transport Layer Security).

Extensible Authentication Protocol *See* EAP.

F

FMC Fixed/Mobile Convergence.

FQDN Fully Qualified Domain Name.

FTP File Transfer Protocol.

function (Specific to IMS) Any one of the identified (and named) separable components of the IMS, which communicates with other functions exclusively using reference points.

G

General Packet Radio Service *See* GPRS.

GGSN Gateway GPRS Support Node.

Global System for Mobile Communications *See* GSM.

GPRS General Packet Radio Service. Packet-based wireless communication service for wireless phones and mobile computer users.

GSM Global System for Mobile Communications. A mobile telephone system that uses a SIM for subscriber identification.

GUI Graphical User Interface.

H

HA Home Agent. Maintains connection information about the mobile station (MS) and manages a persistent IP connection on the network for the MS. (In the SBR/HA 5.5 release, HA meant “High Availability,” but that term has been deprecated in favor of Session State Register, or SSR.)

HAAA Home Authentication, Authorization and Accounting server. AAA server on the subscribers home network.

HLR Home Location Register. Contains the primary subscriber database in a GSM network using SIM or USIM credentials.

home agent *See* HA.

Home PLMN *See* HPLMN.

Home Subscriber Server *See* HSS.

Home WLAN A WLAN that interworks with the HPLMN without using a VPLMN.

hotspot A WLAN Access Point offering network connectivity to the public.

HPLMN Home Public Land Mobile Network. The mobile network that has a billing relationship with the mobile subscriber, and usually the one that authenticates the user and authorizes access.

HSS Home Subscriber Server. The IMS function that contains the primary subscriber database in IMS networks that satisfy Release 6 of the IMS reference (IMS R6).

I

identity protection Prevention of an eavesdropper from discovering the identity of a user being authenticated.

IMS IP Multimedia Subsystem. An IP multimedia and telephony core network that is defined by 3GPP and 3GPP2 standards and organizations based on IETF Internet protocols. IMS is access independent as it supports IP to IP sessions over wireline IP, 802.11, and 802.15 packet data along with GSM/EDGE/UMTS and other packet data applications. IMS is a standard reference architecture that consists of session control, connection control, and an applications services framework along with subscriber and services data.

IMSI International Mobile Subscriber Identity. A unique subscriber identifier consisting of a three-digit Mobile Country Code (MCC), a two- or three-digit Mobile Network Code (MNC), and 10-digits-or-fewer Mobile Subscriber Identification Number (MSIN).

**International Mobile
Subscriber Identity** *See* IMSI.

IP	Internet Protocol.
IP Multimedia Subsystem	<i>See</i> IMS.
IPv4	Implementation of the TCP/IP suite that uses a 32-bit addressing structure.
IPv6	Implementation of the TCP/IP suite that uses a 128-bit addressing structure.
ISP	Internet Service Provider.
J	
JavaScript	Programming language designed for use in distributed environments such as the Internet.
JDBC	Java Database Connectivity. Application programming interface for accessing a database from programs written in Java.
L	
LCI	LDAP configuration interface.
LDAP	Light-weight directory access protocol. An IETF standard protocol for updating and searching directories over TCP/IP networks.
LDIF	LDAP Data Interchange Format. The format used to represent directory server entries in text form.
M	
managed device	A device that runs an SNMP agent.
management station	Host that monitors and controls managed devices running SNMP agents.
MAP	Mobile Access Part. The SS7 protocol standard that addresses registration of roaming users and the intersystem handoff procedure in wireless mobile telephony.
MCC	Mobile Country Code. The MCC, together with the MNC, uniquely identify an operator and help identify the authentication center from which subscriber information should be retrieved.
MGW	Media Gateway.
MIB	Management Information Base. A database of objects, such as alarm status or statistics counters, that can be monitored or overwritten by an SNMP management station.
MNC	Mobile Network Code. The MNC, together with the MCC, uniquely identify an operator and help identify the authentication center from which to retrieve subscriber information.
Mobile Application Part	<i>See</i> MAP.

Mobile Country Code	<i>See</i> MCC.
Mobile Network Code	<i>See</i> MNC.
Mobile Services Switching Center	<i>See</i> MSSC.
Mobile Station	<i>See</i> MS.
Mobile Subscriber ISDN	<i>See</i> MSISDN.
MPPE	Microsoft Point-to-Point Encryption. A means of representing point-to-point packets in an RC4 encrypted format. Defined in RFC 3078.
MS	Mobile Station. Device used to attach to a mobile network.
MS-CHAP	Microsoft CHAP. Proprietary version of CHAP.
MSC	Mobile Services Switching Center. Responsible for connecting calls together by switching packets from one network path to another. MSCs also provide information to support mobile service subscribers, including user registration, authentication, and location updating.
MSISDN	Mobile Subscriber ISDN. Telephone number of the mobile user, which conforms to the dialed number formats in the subscriber's country.
MTP	Message Transfer Part.
N	
NAD	Network Access Device. Network device that accepts connection requests from remote users, authenticates users via RADIUS, and routes users onto the network.
NAI	Network Access Identifier.
NAT	Network Address Translation.
native user	A user authenticated by Steel-Belted Radius Carrier using its internal authentication database.
network element	An addressable node or cluster of nodes in an IMS network, which may host any number of IMS functions.
NGN	Next Generation Network.
NIC	Network Interface Card.

node	A node is a logical element of a Session State Register cluster, which includes SBR Carrier nodes, management nodes, and data nodes.
nonce	Random value included in data exchanges to guarantee uniqueness and protect against replay attacks.
NSP	Network Service Provider.
numbering plan	Interpretation of the digits of an IMSI.
O	
ODB	Operator-Determined Barring. An HLR authorization of service designation that specifies that a subscriber is barred from service.
offline charging	Mechanism for collecting and forwarding charging information concerning I-WLAN and core network resource usage without affecting the service rendered in real-time.
P	
PAP	Password Authentication Protocol. An authentication protocol where a requestor sends an identifier and password to a server after a link has been established. If the identifier and password match an entry in the server's database, the authentication is acknowledged.
PDA	Personal Digital Assistant.
PDSN	Packet Data Serving Node. The attachment point between the RADIUS network and the IP network. May also be known as the foreign agent (FA) when Mobile IP is used.
PEAP	Protected Extensible Authentication Protocol. A two-phase authentication protocol where (1) an authentication server is authenticated to a supplicant using a digital certificate and a secure channel is established; and (2) the supplicant is authenticated to the authentication server via the secure channel.
permanent identity	The permanent identifier of a peer, including an NAI realm portion in environments where a realm is used. The permanent identity is usually based on the IMSI. Used on full authentication only.
PLMN	Public Land Mobile Network. Refers to a mobile network.
point code	The unique identifier for each node in an SS7 network.
PPP	Point-to-Point Protocol. Network protocol defined in RFC 1661 that provides a standard method for transporting multiprotocol datagrams over point-to-point links.
provisioning	A process, possibly requiring multiple steps, that enables customers to obtain services.

proxy RADIUS Process of authenticating users whose profiles are on other RADIUS servers by forwarding access-request packets received from a RADIUS client to a remote RADIUS server (the proxy target), and then forwarding the response from the remote server back to the RADIUS client.

proxy target The remote RADIUS server that actually performs authentication in a proxy RADIUS sequence.

pseudonym identity A pseudonym identifier of a peer, including a NAI realm portion in environments where a realm is used. Used on full authentication only.

**Public Land Mobile
Network** See PLMN.

Q

quintets The authentication data formed by the UMTS values: RAND (random number), XRES (expected response), CK (cipher key), IK (integrity key), and AUTN (authentication).

R

RADIUS Remote Authentication Dial-In User Service. A client/server security administration standard that functions as an information clearinghouse, storing authentication information about users and administering multiple security systems across complex networks.

reauthentication identity The reauthentication identifier for a peer, including a NAI realm portion in environments where a realm is used. Used on reauthentication only.

**Remote Access Dial-In
User Service** See RADIUS.

return list A list of attributes that Steel-Belted Radius Carrier must return to a RADIUS client after authentication of a user succeeds. The return list usually provides additional parameters that the RADIUS client needs to complete the connection.

roaming The ability to move from one Access Point coverage area to another without interruption of service or loss of connectivity.

S

SBC Session Border Controller.

SBR Steel-Belted Radius, the product family that includes Steel-Belted Radius Carrier.

SCTP Stream Control Transmission Protocol. An Internet Protocol used by the SIGTRAN protocol stack to transport SS7 signaling commands. See IETF RFC 4166.

server In a Session State Register cluster, a computer that hosts one or more nodes.

service authorization Authorization allowing a subscriber to access the requested service based on subscription.

session ID	Session Identifier. A string of characters uniquely identifying the session.
Session State Register	<i>See</i> SSR.
SHA-1	Secure Hash Algorithm-1. A one-way cryptographic function that takes a message of any length and produces a 160-bit message digest.
Signaling System 7	<i>See</i> SS7.
Signalware	The Mavenir SIGTRAN protocol stack provided with Steel-Belted Radius Carrier.
SIGTRAN	Protocol stack supporting SS7 signaling using the SCTP Internet Protocol. <i>See</i> IETF RFC 4166.
silent discard	The process of discarding a packet without further processing and without notification to the sender.
SIM	Subscriber Identity Module.
SIM card	A SIM-based hardware SmartCard that contains the authentication keys for a GSM mobile telephone subscriber.
SIP	Session Initiation Protocol.
SmartCard	A small card containing a computer chip that can store information, including authentication information and algorithms.
SNMP	Simple Network Management Protocol.
SS7	Signaling System 7. The network and protocols used to provide out-of-band signaling (control) for telephone services to support call establishment, billing, routing, and information exchange for the public switched telephone network.
SSID	Service Set Identifier.
SSL	Secure Sockets Layer. Program layer that manages the security of messages on a network.
SSR	Session State Register, an optional module for Steel-Belted Radius Carrier that implements a multi-computer cluster to support shared databases that multiple SBR Carrier servers can access to ensure that a single set of data is used for all transactions and to implement a high-availability environment.
STP	Signaling Transfer Point.
supplicant	The client in an 802.1X-authenticated network.

T

TISPAN	Telecoms & Internet converged Services & Protocols for Advanced Networks (standardization body of ETSI).
TLS	Transport Layer Security.
TLV	Type-Length-Value. A synonym for AVP; named because the raw encoding of such a value is a type field (for example, 1 for User-Name) followed by a length value (for example, 6) followed by the value of the attribute (for example, test).
trap	An SNMP message that reports a significant event, such as a problem, error, or change in state, that occurred within a managed device.
trap sink	The destination for trap messages sent by an SNMP agent on a managed device.
TS	Teleservice. HLR authorization of service designation.
TTLS	Tunneled Transport Layer Security.

U

UE	User Equipment.
UICC	Universal Integrated Circuit Card. The chip card used in mobile terminals in GSM and UMTS networks. The UICC ensures the integrity and security of all kinds of personal data, and typically holds a few hundred kilobytes.
UMA	Unlicensed Mobile Access.
UMTS	Universal Mobile Telecommunications System. Type of mobile network (next generation after GSM) that uses the USIM card for authentication.
Universal Mobile Telecommunications System	<i>See</i> UMTS.
user database	A database where a RADIUS server keeps information about users, such as authentication information and network access permissions.
user identifier	Identifier of a user that may be used, for example, in charging functionality for billing purposes.
user profile	A record in the user database that describes how to configure a particular user or class of users during authentication and authorization.
USIM	UMTS Subscriber Identity Module.

USIM card A SIM-based hardware SmartCard that contains the authentication keys for a 3G mobile telephone subscriber.

V

VAAA Visited Authentication, Authorization and Accounting server. AAA server on the visited access network, responsible for routing authentication and accounting requests to home network.

Visited PLMN See VPLMN.

VLAN Virtual Local Area Network.

VLR Visitors Location Register.

VoIP Voice over IP.

VPLMN Visited Public Land Mobile Network. The mobile network that is providing connectivity to a roaming user.

VPN Virtual Private Network.

VSA Vendor-Specific Attributes. Usually refers to a vendor-specific attribute *and* its associated value. VSA may be used to indicate a vendor-specific attribute or vendor-specific AVP. In RADIUS, VSAs are special attributes that contain an IANA-assigned enterprise code followed by TLVs (Type Length Value) that can be defined by the vendor who owns the enterprise code. As a result, vendors can define their own RADIUS VSAs without fear of colliding with another vendor's VSA assignments.

W

W-CDMA Wideband Code Division Multiple Access.

W-CDR Wireless LAN type of CDR.

WEP Wired Equivalent Privacy. An encryption method designed to encrypt traffic between a WLAN client and an access point.

Wi-Fi Wireless local area network that uses the IEEE 802.11a, b, or g radio protocols.

WiMAX Worldwide Interoperability for Microwave Access.

WISP Wireless Internet Service Provider.

WLAN Wireless Local Area Network.