

Chapter 3

Configuring the SAE with SDX Configuration Editor

This chapter describes how to use SDX Configuration Editor to configure general SAE properties. You can use SDX Configuration Editor on a Solaris platform.

You can also use the SRC CLI to configure and SAE on a Solaris platform or on a C-series platform. See *Chapter 2, Configuring the SAE with the SRC CLI*.

Topics in this chapter include:

- Overview of Configuring the SAE on page 29
- Configuring LDAP Access to Directory Data on page 29
- Storing Subscriber and Service Session Data on page 41
- Configuring the Session Store Feature on page 42
- Configuring the Number of Threads for Sessions on page 50

Overview of Configuring the SAE

The SAE property file contains SAE configuration data that is stored in the directory. You can modify the SAE property file with SDX Configuration Editor, SDX Admin, or a standard text editor. This chapter shows how to configure SAE properties with SDX Configuration Editor. Each field description includes a property name, which is used if you modify the properties with SDX Admin or a text editor. For information about modifying the property file with SDX Admin, see *SRC-PE Subscribers and Subscriptions Guide, Chapter 5, Configuring Subscriber-Related Properties on the SAE on a Solaris Platform*.

Configuring LDAP Access to Directory Data

The SRC software stores subscriber, service, persistent login, policy, router, and cached subscriber profiles and session data in a directory. The SAE uses LDAP to store and retrieve the data. You can configure the LDAP connections to the directories in which this data is stored. You can also select the filter that the SAE uses to search for subscriptions in the directory and directory eventing parameters for data stored in the directory.

The tasks to configure LDAP access to directory data are:

- Configuring Access to Subscriber Data on page 30
- Configuring Access to Service Data on page 33
- Configuring Access to Policy Data on page 35
- Configuring Access to the Persistent Login Cache on page 37
- Configuring the Location of Router, Persistent Login, and Persistent Session Data on page 39
- Enabling Automatic Discovery of Changes in SAE Configuration Data on page 40

Configuring Access to Subscriber Data

To use SDX Configuration Editor to configure the LDAP connection from the SAE to the directory in which subscriber data is stored:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab, and expand the **User Data** section.

User Data	
Server Address	127.0.0.1 Disable
Search Base	o=Users, <base>
Authentication DN	cn=ssp,ou=Components,o=Operators,<base> Disable
Password	*** Show Disable
Enable Directory Eventing	Yes ▼
Directory Polling Interval [s]	30
Secured LDAP protocol	LDAPS ▼ Disable
Filter for loading subscriptions	Subscription Objectclass Filter ▼

3. Edit or accept the default values in the fields.
See *User Data Fields* on page 31.
4. Select **File** > **Save**.
5. Right-click the configuration file, select **SDX System Configuration** > **Export to LDAP Directory**.

User Data Fields

In SDX Configuration Editor, you can modify the following fields in the User Data section of the LDAP pane in an SAE configuration file.

Server Address

- Disables or enables and identifies the directory server that stores subscriber information.
- Value—IP address or hostname; use a space to separate addresses for multiple directory servers: 127.153.27.1 192.168.0.1
- Default—Disabled
- Property name—`UserDataSource.repository.ldap.server.address`

Search Base

- Subtree in the directory in which subscriber information is stored. When a subscriber logs in to a residential portal, the SAE searches subscriber profiles by mapping the realm of the login name to a retailer object found below the search base.
- Value—`<DN>`
You can use the special value `<base>` to refer to the globally configured base distinguished name (DN).
- Guidelines—Sensible values include `o = Users`, `o = umc` for multidomain support and `retailerName = Retailer`, `o = Users`, `o = umc` for single domain support.
- Default—`o = Users`, `<base>`
- Property name—`UserDataSource.repository.ldap.server.base.dir`

Authentication DN

- Disables or enables and sets the DN that the SAE uses to authenticate access to the directory server. The specified directory entry must exist and have read access to all attributes. The entry must have write access if subscribers are allowed to customize their subscription profiles.
- Value—`<DN>`
You can use the special value `<base>` to refer to the globally configured base DN.
- Default—Disabled, which means that the value configured for the directory is used
- Property name—`UserDataSource.repository.ldap.server.authDN`

Password

- Disables or enables and sets the password used to authenticate access to the directory server. You must configure the password in the directory to authenticate read-access to the directory.
- Value—Text string or base64 string that matches the value of the `userPassword` attribute of the authentication DN

- Default—Disabled, which means that the value configured for the directory is used
- Property name—`UserDataSource.repository.ldap.server.password`

Enable Directory Eventing

- Enables or disables automatic discovery of changes in subscriber profiles.
- Value
 - Yes—Changes in the subscriber profile or subscriptions take effect automatically while the subscriber is logged in.
 - No—Changes in the subscriber profile or subscriptions do not take effect until the next time the subscriber logs in.
- Default—Yes
- Property name—`UserDataSource.repository.ldap.server.des.enable_eventing`

Directory Polling Interval [s]

- Sets the frequency for checking the directory for updates.
- Value—Number of seconds in the range 15–86400
- Default—30
- Property name—`UserDataSource.repository.ldap.server.des.pollinginterval`

Secured LDAP protocol

- Enables or disables LDAPS as the secure protocol for connections to the server that stores subscriber data.
- Value—Enable or Disable
- Default—Disable
- Property name—`UserDataSource.repository.ldap.server.security.protocol`

Filter for loading subscriptions

- Selects the filter that the SAE uses to search for subscriptions in the directory when the SAE loads a subscription.
- Value—Select one of the following values from the drop-down menu:
 - Subscriber reference filter—The SAE runs a search based on the `subscriberRef` attribute in the `umcServiceProfile` object, which is the base object class of the service profile hierarchy. The `subscriberRef` attribute contains a DN that points to the parent of the subscriber object.
 - Subscription Objectclass filter—The SAE performs a one-level search with the directory entry, which represents the subscriber folder as the base DN. The search filter is `(objectClass = sspServiceProfile)`. This method can be slow if you have a large number of subscription entries within the subscriber folder subtree.
- Default—Subscription Objectclass filter
- Property name—`UserDataSource.repository.ldap.server.loadSubscriptionFilter`

Configuring Access to Service Data

To use SDX Configuration Editor to configure the LDAP connection from the SAE to the directory in which service data is stored:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab, and expand the **Service Data** section.

3. Edit or accept the default values in the fields.
See *Service Data Fields* on page 33.
4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Service Data Fields

In SDX Configuration Editor, you can modify the following fields in the Service Data section of the LDAP pane in an SAE configuration file.

Server Address

- Disables or enables and identifies the directory server that stores service data.
- Value—IP address or hostname; use a space to separate addresses for multiple directory servers: 127.153.27.1 192.168.0.1
- Default—Disabled, which means that the value configured for the directory is used
- Property name—ServiceDataSource.repository.ldap.server.address

Search Base

- Subtree in the directory in which service information is stored. The SAE loads service definitions on startup and when service reloading is requested.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default— < base >
- Property name—ServiceDataSource.repository.ldap.server.base.dir

Authentication DN

- Disables or enables and sets the DN that the SAE uses to authenticate access to the directory server. The specified directory entry must exist and have read access to all attributes.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—Disabled, which means that the value configured for the directory is used
- Property name—ServiceDataSource.repository.ldap.server.authDN

Password

- Disables or enables and sets the password used to authenticate access to the directory server. You must configure the password in the directory to authenticate read access to the directory.
- Value—Text string or base64 string
- Default—Disabled, which means that the value configured for the directory is used
- Property name—ServiceDataSource.repository.ldap.server.password

Enable Directory Eventing

- Enables or disables automatic discovery of changes in service definitions.
- Value
 - Yes—Changes in service definitions take effect automatically. If a changed service is in use, all service instances are deactivated and then reactivated with the modified settings. Consequently, service may be affected for subscribers who are logged in at the time of the modification.
 - No—Changes in service definitions do not take effect until the SAE is restarted.
- Default—Yes
- Property name—ServiceDataSource.repository.ldap.server.des.enable_eventing

Directory Polling Interval [s]

- Sets the frequency for checking the directory for updates.
- Value—Number of seconds in the range 15–86400
- Default—30
- Property name—ServiceDataSource.repository.ldap.server.des.pollinginterval

Secured LDAP protocol

- Enables or disables LDAPS as the secure protocol for connections to the server that stores service data.
- Value—Enable or Disable
- Default—Disable
- Property name—ServiceDataSource.repository.ldap.server.security.protocol

Configuring Access to Policy Data

To use SDX Configuration Editor to configure the LDAP connection from the SAE to the directory in which policy data is stored:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab, and expand the **Policy Data** section.

Policy Data	
Policy Search Base	<input type="text" value="o=Policies, <base>"/>
Parameter Search Base	<input type="text" value="o=Parameters, <base>"/>
Enable Directory Eventing	<input type="text" value="Yes"/>
Directory Polling Interval [s]	<input type="text" value="30"/>

3. Edit or accept the default values in the fields.
See *Policy Data Fields* on page 36.
4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Policy Data Fields

In SDX Configuration Editor, you can modify the following fields in the Policy Data section of the LDAP pane in an SAE configuration file.

Policy Search Base

- Subtree in the directory that stores policy data.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*o = Policies, < base >*
- Property name—PolicyDataSource.repository.ldap.baseDN

Parameter Search Base

- Subtree in the directory that stores policy parameter data.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*o = Parameters, < base >*
- Property name—PolicyDataSource.repository.ldap.parameterBaseDN

Enable Directory Eventing

- Enables or disables automatic discovery of changes in policy definitions and in interface classifiers.
- Value
 - Yes—Changes in policy definitions take effect automatically. If a changed policy is in use, all policy instances are deactivated and then reactivated with the modified settings. Consequently, service may be affected for subscribers who are logged in when the change is made.
 - No—Changes in policy definitions do not take effect until the SAE is restarted.
- Default—Yes
- Property name—net.juniper.smgmt.des.enable_eventing

Directory Polling Interval [s]

- Sets the frequency for checking the directory for updates.
- Value—Number of seconds in the range 15–86400
- Default—30
- Property name—net.juniper.smgmt.des.pollinginterval

Configuring Access to the Persistent Login Cache

To use SDX Configuration Editor to configure the LDAP connection from the SAE to the directory in which persistent login cache data is stored:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab, and expand the **Persistent Login Cache** section.

3. Edit or accept the default values in the fields.

See *Persistent Login Cache Data Fields* on page 37.

4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Persistent Login Cache Data Fields

In SDX Configuration Editor, you can modify the following fields in the Persistent Login Cache section of the LDAP pane in an SAE configuration file.

Server Address

- Disables or enables and identifies the directory server that stores persistent login data.
- Value—IP address or hostname; use a space to separate addresses for multiple directory servers: 127.153.27.1 192.168.0.1
- Default—Disabled, which means that the value configured for the directory is used
- Property name—UserCacheDataSource.repository.ldap.server.address

Search Base

- Subtree in the directory that stores persistent login cache data.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*ou-authCache, < base >*
- Property name—`UserCacheDataSource.repository.ldap.server.base.dir`

Authentication DN

- Disables or enables and sets the DN that the SAE uses to authenticate access to the directory server. The specified directory entry must exist and have read access to all attributes.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—Disabled
- Property name—`UserCacheDataSource.repository.ldap.server.authDN`

Password

- Disables or enables and sets the password used to authenticate access to the directory server. You must configure the password in the directory to authenticate read access to the directory.
- Value—Text string or base64
- Default—ssp
- Property name—`UserCacheDataSource.repository.ldap.server.password`

Enable Directory Eventing

- Enables or disables automatic discovery of changes to the persistent login cache.
- Value—Yes or No
- Default—No
- Property name—`UserCacheDataSource.repository.ldap.server.des.enable_eventing`

Directory Polling Interval [s]

- Sets the frequency for checking the directory for updates.
- Value—Number of seconds in the range 15–86400
- Default—30
- Property name—`UserCacheDataSource.repository.ldap.server.des.pollinginterval`

Secured LDAP protocol

- Enables or disables LDAPS as the secure protocol for connections to the server that stores persistent login cache data.
- Value—Enable or Disable
- Default—Disable
- Property name—UserCacheDataSource.repository.ldap.server.security.protocol

Configuring the Location of Router, Persistent Login, and Persistent Session Data

To use SDX Configuration Editor to configure the location of router data, persistent login information for DHCP scenarios, and persistent session data:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab.

The screenshot shows the LDAP configuration pane with three search base fields:

- Network Data Search Base:** o=Network, <base>
- SAE Cache Repository Search Base:** o=userProfileCache, <base>
- Persistent Session Cache repository search base:** o=PersistentSessions, <base>

3. Edit or accept the default values in the Network Data Search Base, SAE Cache Repository Search Base, and Persistent Session Cache Repository Search Base fields.

See *Router Data, DHCP Persistent Login Information, and Persistent Session Data Fields* on page 39.

4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Router Data, DHCP Persistent Login Information, and Persistent Session Data Fields

In SDX Configuration Editor, you can edit the following fields in the LDAP pane in an SAE configuration file.

Network Data Search Base

- Subtree in the directory that stores router data.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*o = Network, < base >*
- Property name—`NetworkDataSource.repository.ldap.baseDN`

SAE Cache Repository Search Base

- Base DN for storing and retrieving subscriber profiles. This is the directory subtree in which persistent login information is stored for DHCP scenarios.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*o = userProfileCache, < base >*
- Property name—`UserDataSource.repository.ldap.server.cache.dir`

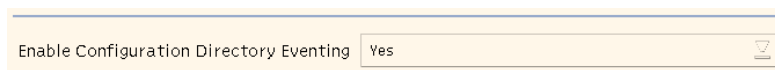
Persistent Session Cache Repository Search Base

- Base DN for storing and retrieving persistent session information.
- Value— < DN >
You can use the special value < base > to refer to the globally configured base DN.
- Default—*o = PersistentSessions, < base >*
- Property name—`UserDataSource.repository.ldap.server.persistent.session`

Enabling Automatic Discovery of Changes in SAE Configuration Data

To use SDX Configuration Editor to enable directory eventing of SAE configuration data:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **LDAP** tab.



3. Edit or accept the default value in the Enable Configuration Directory Eventing field.

See *Enable Configuration Directory Eventing Field* on page 41.

4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Enable Configuration Directory Eventing Field

In SDX Configuration Editor, you can edit the following fields in the LDAP pane in an SAE configuration file.

Enable Configuration Directory Eventing

- Enables automatic discovery of changes in the SAE configuration data. If this property is enabled, the SAE detects changes in its configuration and reconfigures itself.
- Value—Yes or No
- Default—Yes
- Property name—Main.auto-reconfigure

Storing Subscriber and Service Session Data

To aid in recovering from an SAE failover, the SAE stores subscriber and service session data. In releases earlier than SDX Release 6.2, the software stored subscriber and session data on the router. The SAE now stores session data in flat files on the SAE host. The SRC component that controls the storage of session data on the SAE is called the session store. The session store queues data and then writes the data to session store files on the SAE host's disk. Once the data is written to disk, it can survive a server reboot.

You can configure how the SAE stores session data for JUNOSe routers, JUNOS routing platforms, simulated routers, and *PacketCable Multimedia Specification* (PCMM) devices.

Session Store Files

Session store files are numbered flat files. When the SAE starts, the session store component creates a directory on the SAE host. Session store files are located in the directory. You can configure the size of session store files. Once the maximum size is reached, the session store creates a new file and begins writing data to the new file.

Store operations, such as adding a session to the store (put store operations) or removing a session from the store (remove store operations), are queued in a buffer before they are written to the session store file. You can configure parameters that determine when the session store writes a queue to a session store file.

Session store files are deleted if they have not been modified and if no session activity has taken place for one week. All of the data files that contain the sessions associated with a particular virtual router are deleted at the same time.

Active and Passive Session Stores

You can have a community of SAEs and duplicate session store data on each SAE in the community in case of an SAE failover. SAE communities are made up of SAEs that you configure as connected SAEs for a virtual router object. You can configure SAE communities with SDX Admin by selecting the SAE Connection tab in the VirtualRouter pane.

SAEs in a community are given the role of either active SAE or passive SAE. The active SAE keeps session data up to date within the community. Each active session store opens a Transmission Control Protocol (TCP) connection to its passive SAE. The TCP connection triggers the creation of a passive session store in that SAE. When the active session store writes operations to the session store file, it passes them to passive sessions stores on all SAEs in the community.

When you modify a community, wait for passive session stores on the new community members to be updated before you shut down the currently active SAE. Otherwise, if you add a new member to a community, and then a failover from the current active SAE to the new member is triggered immediately, the new member's session store may not have received all data from the active SAE's session store.

Session Store File Rotation

The session store periodically rotates the session store files. During rotation, the session store copies put store operations for live sessions from the oldest file to the end of the newest file. (Live sessions are sessions that have been created but not yet deleted.) It then deletes the oldest file. Sessions are rotated in batches, and you can configure the number of sessions that are rotated at the same time, and how much disk space is used by live sessions before files are rotated. No session store activity can take place while a batch of sessions is rotated.

Configuring the Session Store Feature

There are three things that you can configure for the session store feature:

- Configure session store parameters for a router or device driver. See *Configuring Session Store Parameters for a Device Driver* on page 43.
- Configure global session store parameters that are shared by all session store instances (active or passive) on the SAE. See *Configuring Global Session Store Parameters* on page 47.
- Reduce the size of session objects that the SAE sends across the network for the session store feature. See *Reducing the Size of Objects for the Session Store Feature* on page 48.

Configuring Session Store Parameters for a Device Driver

This section describes how to configure session store parameters within a router or other device driver configuration.

To use SDX Configuration Editor to configure session store parameters for a router driver or other device driver:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **Router** tab, and expand the driver section.

Session Store	
Maximum Queue Age [ms]	5000
Maximum Queued Operations	50
Maximum Queue Size [bytes]	51050
Maximum File Size [bytes]	25000000
Minimum Disk Space Usage	25
Rotation Batch Size	50
Maximum Session Data Bytes	10000
Disk Load Buffer Size [bytes]	1000000
Network Buffer Size [bytes]	51050
Retry Interval [ms]	300000
Communications Timeout [ms]	60000
Load Timeout [ms]	420000
Session Store Idle Timeout [ms]	3600000
Maximum Backlog Ratio	1.5
Minimum Backlog	5000000

3. Edit or accept the default values in the fields.

See *Session Store Fields* on page 44.

4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Session Store Fields

In SDX Configuration Editor, you can edit the following fields for a driver in the Router pane in an SAE configuration file.

Maximum Queue Age [ms]

- Maximum age that a queue of buffered store operations (such as adding a session to the store or removing a session from the store) can reach before the queue is written to a session store file.
- Value—Number of milliseconds in the range 0–2147483647. A value of –1 indicates that there is no limit. A value of zero causes the session store to write each store operation to a session store file immediately.
- Default—5000
- Property name—Router. < deviceType > .sessionStore.maxQueueAge

Maximum Queued Operations

- Number of buffered store operations that are queued before the queue is written to a session store file.
- Value—Integer in the range 0–2147483647. A value of –1 indicates that there is no limit. A value of zero causes the session store to write each store operation to a session store file immediately.
- Default—50
- Property name—Router. < deviceType > .sessionStore.maxQueueOps

Maximum Queue Size [bytes]

- Maximum size that a queue of buffered store operations can reach before the queue is written to a session store file.
- Value—Number of bytes in the range 0–2147483647
- Default—51050
- Property name—Router. < deviceType > .sessionStore.maxQueueBytes

Maximum File Size [bytes]

- Maximum size of session store files. When a file reaches this size, a new file is created.
- Value—Number of bytes in the range 0–2147483647
- Default—25000000
- Property name—Router. < deviceType > .sessionStore.maxFileBytes

Minimum Disk Space Usage

- Percentage of space in all session store files that is used by live sessions. When the percentage of space in the session store files that is used by live sessions decreases to this percentage, the oldest session store file is compacted and appended to the newest session store file, and then the oldest session store file is deleted.
- Value—Percentage of disk space in the range 1–100

- Guidelines—We recommend a range of 30–50.
- Default—40
- Property name—Router. < deviceType > .sessionStore.minDiskSpaceUsage

Rotation Batch Size

- When the oldest session store file is rotated, specifies the number of sessions that are rotated from the oldest file to the newest file at the same time. While a set of sessions is rotated, no other session store activity can take place.
- Value—Integer in the range 0–2147483647
- Default—50
- Property name—Router. < deviceType > .sessionStore.rotationBatchSize

Maximum Session Data [bytes]

- Maximum size of a single subscriber or service session. Use this parameter to reserve memory for an internal buffer.
- Value—Number of bytes in the range 0–2147483647
- Default—10000
- Property name—Router. < deviceType > .sessionStore.maxSessionDataBytes

Disk Load Buffer Size [bytes]

- Size of the buffer that is used to load all of a session store's files from disk at startup.
- Value—Number of bytes in the range 0–2147483647
- Default—1000000
- Property name—Router. < deviceType > .sessionStore.diskLoadBufferBytes

Network Buffer Size [bytes]

- Size of the buffer that holds messages or message segments that are waiting to be sent to passive session stores.
- Value—Number of bytes in the range
21 + < size of maximum session size field > – 2147483647
- Default—51050
- Property name—Router. < deviceType > .sessionStore.networkBufferBytes

Retry Interval [ms]

- Time interval between attempts by the active session store to connect to missing passive session stores.
- Value—Number of milliseconds in the range 0–2147483647
- Default—300000
- Property name—Router. < deviceType > .sessionStore.retryInterval

Communications Timeout [ms]

- Amount of time that a session store waits before closing when it is blocked from reading or writing a message. This timeout does not apply when a session store is waiting for a remote session store to load its state from disk.
- Value—Number of milliseconds in the range 0–2147483647
- Default—60000
- Property name—Router. < deviceType > .sessionStore.communicationsTimeout

Load Timeout [ms]

- Amount of time that an active session store waits for a passive session store or a passive session store waits for an active session store to load its data from disk before it closes the connection to the session store.
- Value—Number of milliseconds in the range 0–2147483647
- Default—420000
- Property name—Router. < deviceType > .sessionStore.remoteStoreLoadTimeout

Session Store Idle Timeout [ms]

- Amount of time that a passive session store waits for activity from the active session store before it closes the connection to the active session store. This timeout applies after the session store startup and initial update processes are complete.
- Value—Number of milliseconds in the range 0–2147483647
- Default—3600000
- Property name—Router. < deviceType > .sessionStore.idleTimeout

Maximum Backlog Ratio

- Along with the minimum backlog size, specifies when the active session store closes the connection to a passive session store because of a backlog of messages waiting to be sent. After the startup and initial update processes are complete, if the backlog becomes too large, the connection to the passive session store is closed. After the retry interval ends, a new connection is opened.

If the backlog of unsent operations (in bytes) divided by the total size (in bytes) of all live store operations is greater than this number, the connection is closed.
- Value—Floating point number
- Default—1.5
- Property name—Router. < deviceType > .sessionStore.backlogDeathMaxRatio

Minimum Backlog Size [bytes]

- Along with the maximum backlog ratio, specifies when the active session store closes the connection to a passive session store because of a backlog of messages waiting to be sent to the passive session store. After the startup and initial update processes are complete, if the backlog becomes too large, the connection to the passive session store is closed. After the retry interval ends, a new connection is opened.

If the maximum backlog ratio is met, the active session store does not close the connection unless the backlog of messages (in bytes) is greater than this number.

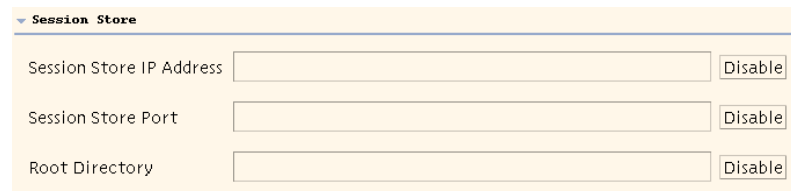
- Value—Number of bytes in the range 0–2147483647
- Default—5000000
- Property name—Router. < deviceType > .sessionStore.backlogDeathMinBehind

Configuring Global Session Store Parameters

This section describes how to configure global session store parameters that are shared by all session store instances (active or passive) on the SAE. You can also configure session store parameters within a router or other device driver configuration. See *Configuring Session Store Parameters for a Device Driver* on page 43.

To use SDX Configuration Editor to configure global session store parameters:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **Router** tab, and expand the **Session Store** section.



The screenshot shows a configuration window titled "Session Store" with a dropdown arrow. Below the title, there are three rows of configuration fields:

Parameter	Value Field	Action
Session Store IP Address	<input type="text"/>	Disable
Session Store Port	<input type="text"/>	Disable
Root Directory	<input type="text"/>	Disable

3. Edit or accept the default values in the fields.
See *Global Session Store Fields* on page 48.
4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Global Session Store Fields

In SDX Configuration Editor, you can edit the following fields in the Session Store section of the Router pane in an SAE configuration file.

Session Store IP Address

- IP address or hostname that the session store infrastructure on this SAE uses to listen for incoming TCP connections from active session stores.
- Value—IP address. The address must be an IP address configured for the SAE host. If you do not enter an address or if you disable this field, active session stores cannot create passive session stores on this SAE.
- Guidelines—We recommend that you enter an address that is configured in a list of connected SAEs.
- Default—No value
- Property name—Router.sessionStore.ServerIp

Session Store Port

- TCP port number on which the session store infrastructure on this SAE listens for incoming connections from active session stores. Note that this field has no effect if you have not configured a session store IP address.
- Value—Port number
- Default—8820
- Property name—Router.sessionStore.ServerPort

Root Directory

- Root directory in which the session store creates files. Note that this field has no effect if you have not configured a session store IP address.
- Value—Directory name
- Default—var/sessionStore
- Property name—Router.sessionStore.rootDir

Reducing the Size of Objects for the Session Store Feature

You can use serialized data compression to reduce the size of sessions objects that the SAE sends across the network for the session store feature. Enabling this property reduces the size of objects, but increases the CPU load on the SAE.

To use SDX Configuration Editor to specify whether or not session objects are compressed:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **Miscellaneous** tab.



The screenshot shows a configuration field labeled 'Compress Serialized Data' with an empty text input box and a dropdown arrow icon on the right.

3. Fill in the Compress Serialized Data field.
See *Compressed Serialized Data Field* on page 49.
4. Select **File > Save**.
5. Right-click the configuration file, select **SDX System Configuration > Export to LDAP Directory**.

Compressed Serialized Data Field

In SDX Configuration Editor, you can edit the following field in the Miscellaneous pane in an SAE configuration file.

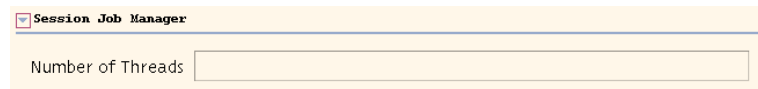
Compress Serialized Data

- Enables or disables reducing the size of session objects (subscriber and service sessions) that the SAE sends across the network for the session store feature.
- Value—Yes or No
- Guidelines—We recommend that you do not enable this option because of the increase to the CPU load.
- Default—No
- Property name—Main.compressSerializedData

Configuring the Number of Threads for Sessions

To use SDX Configuration Editor to configure the number of threads used to handle session-related activity:

1. In the navigation pane, select a configuration file for the SAE that you want to configure.
2. Select the **Miscellaneous** tab, and expand the **Session Job Manager** section.



3. Fill in the field.

See *Number of Threads for Sessions Field* on page 50.

4. Select **File > Save**.
5. Right-click the configuration file, **select SDX System Configuration > Export to LDAP Directory**.

Number of Threads for Sessions Field

In SDX Configuration Editor, you can edit the following field in the Session Job Manager section of the Miscellaneous pane in an SAE configuration file.

Number of Threads

- Sets the number of threads used for session-related activity; for example, interim accounting, subscriber and service session timeout, idle timeouts, aggregate service keepalives, and remote session monitoring.
- Value—Integer in the range 1–50
- Default—10
- Property name—SessionJobManager.numThreads