



Junos[®] OS

Broadband Subscriber Management Getting Started Guide



Modified: 2017-02-15

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

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the 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.

Junos[®] OS Broadband Subscriber Management Getting Started Guide
Copyright © 2017, Juniper Networks, Inc.
All rights reserved.

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 <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

	About the Documentation	ix
	Documentation and Release Notes	ix
	Supported Platforms	ix
	Using the Examples in This Manual	ix
	Merging a Full Example	x
	Merging a Snippet	x
	Documentation Conventions	xi
	Documentation Feedback	xiii
	Requesting Technical Support	xiii
	Self-Help Online Tools and Resources	xiii
	Opening a Case with JTAC	xiv
Part 1	Subscriber Management Overview	
Chapter 1	Subscriber Management Overview	3
	Subscriber Management Overview	3
	Subscriber Access Terms and Acronyms	4
	AAA Service Framework and Subscriber Management Overview	4
	Class of Service and Subscriber Management Overview	5
	Configuring Subscriber Access	5
	Subscriber Activation and Service Management in an Access Network	7
	Components of a Dynamic Profile	8
	Router Predefined Variables Used by Dynamic Profiles	8
Chapter 2	Subscriber Management Licensing	9
	Subscriber Access Licensing Overview	9
	Configuring the Router to Strictly Enforce the Subscriber Scaling License	9
Part 2	Creating Dynamic Configurations for Subscriber Management	
Chapter 3	Configuring Dynamic Profiles for Subscriber Management	13
	Dynamic Profiles Overview	13
	Dynamic Profile Interface Support	14
	What Dynamic Profiles Do	14
	How Dynamic Profiles Work	14
	Dynamic Profile Overrides	14
	Dynamic Profile Version Creation	14
	Dynamic Profile Semantic Checking	15
	Configuring a Basic Dynamic Profile	16
	Access Profiles and Service Profiles Overview	17
	Functionality Supported in Access and Service Profiles	17

Chapter 4	Using Variables in Dynamic Profiles	19
	Dynamic Variables Overview	19
	How Dynamic Variables Work	19
	Default Values for Predefined Variables	20
	Unique Identifier (UID) for Parameterized Filters	20
Chapter 5	Using Predefined Variables in Dynamic Profiles	21
	Junos OS Predefined Variables	21
	Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs	42
	Configuring Predefined Dynamic Variables in Dynamic Profiles	49
	Configuring Default Values for Predefined Variables in a Dynamic Profile	49
Chapter 6	Creating Customized Variables in Dynamic Profiles	51
	User-Defined Variables	51
	Configuring User-Defined Dynamic Variables in Dynamic Profiles	52
	Using Variable Expressions in User-Defined Variables	53
	Configuring Variable Expressions in Dynamic Profiles	56
	Conditional Configuration for Dynamic Profile Overview	58
Chapter 7	Modifying Dynamic Profiles That Are in Use by Active Subscribers	63
	Enabling Dynamic Profiles to Use Multiple Versions	63
	Modifying Dynamic Profiles with Versioning Disabled	64
Part 3	Configuration Statements	
Chapter 8	Configuration Statements	69
	default-value	69
	dynamic-profile-options	70
	dynamic-profiles	71
	mandatory	79
	predefined-variable-defaults (Dynamic Profiles)	80
	subscriber-management (Subscriber Management)	81
	variables	82
	versioning	82
Chapter 9	Operational Commands	83
	show dynamic-profile session	84
	show system subscriber-management route	89

List of Figures

Part 1	Subscriber Management Overview	
Chapter 1	Subscriber Management Overview	3
	Figure 1: Subscriber Access Configuration Workflow	7

List of Tables

	About the Documentation	ix
	Table 1: Notice Icons	xi
	Table 2: Text and Syntax Conventions	xi
Part 1	Subscriber Management Overview	
Chapter 1	Subscriber Management Overview	3
	Table 3: Subscriber Access Terms and Acronyms	4
Part 2	Creating Dynamic Configurations for Subscriber Management	
Chapter 3	Configuring Dynamic Profiles for Subscriber Management	13
	Table 4: Types of Variables Supported in Dynamic Profiles	18
	Table 5: Default Values and Expressions Supported in Dynamic Profiles	18
Chapter 5	Using Predefined Variables in Dynamic Profiles	21
	Table 6: Junos OS Predefined Variables and Definitions	21
	Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables	42
Chapter 6	Creating Customized Variables in Dynamic Profiles	51
	Table 8: Operators and Functions	54
	Table 9: Expression Errors and Actions	56
	Table 10: Expression Examples	57
Part 3	Configuration Statements	
Chapter 9	Operational Commands	83
	Table 11: show system subscriber-management route Output Fields	90

About the Documentation

- Documentation and Release Notes on page ix
- Supported Platforms on page ix
- Using the Examples in This Manual on page ix
- Documentation Conventions on page xi
- Documentation Feedback on page xiii
- Requesting Technical Support on page xiii

Documentation and Release Notes

To obtain the most current version of all Juniper Networks[®] technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- MX Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see [CLI Explorer](#).

Documentation Conventions

[Table 1 on page xi](#) defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

[Table 2 on page xi](#) defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure

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

Convention	Description	Examples
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none">Introduces or emphasizes important new terms.Identifies guide names.Identifies RFC and Internet draft titles.	<ul style="list-style-type: none">A policy <i>term</i> is a named structure that defines match conditions and actions.<i>Junos OS CLI User Guide</i>RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none">To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level.The console port is labeled CONSOLE.
< > (angle brackets)	Encloses optional keywords or variables.	stub <default-metric <i>metric</i>>;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast (<i>string1</i> <i>string2</i> <i>string3</i>)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Encloses a variable for which you can substitute one or more values.	community name members [<i>community-ids</i>]
Indentation and braces ({ })	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none">In the Logical Interfaces box, select All Interfaces.To cancel the configuration, click Cancel.

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

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select Protocols>Ospf .

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

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 Partner Support Service 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 <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://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: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>

- Download the latest versions of software and review release notes:
<http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications:
<http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum:
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

Opening a Case with JTAC

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

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- 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, see <http://www.juniper.net/support/requesting-support.html>.

PART 1

Subscriber Management Overview

- [Subscriber Management Overview on page 3](#)
- [Subscriber Management Licensing on page 9](#)

CHAPTER 1

Subscriber Management Overview

- [Subscriber Management Overview on page 3](#)
- [AAA Service Framework and Subscriber Management Overview on page 4](#)
- [Class of Service and Subscriber Management Overview on page 5](#)
- [Configuring Subscriber Access on page 5](#)
- [Subscriber Activation and Service Management in an Access Network on page 7](#)

Subscriber Management Overview

The Juniper Networks Junos OS subscriber management feature provides subscriber access, authentication, and service creation, activation, and deactivation. You can also collect accounting information and statistics for subscriber service sessions.

The subscriber access feature supports both CLI and AAA-based configuration (such as RADIUS) for subscribers. Access and services start when the router receives a message from a client (such as a DHCP discover message). For RADIUS clients, RADIUS Access-Accept messages and Change-of-Authorization-Request (CoA-Request) messages can create, modify, and delete subscriber sessions as well as activate and deactivate service sessions. You can use CLI commands to create a dynamic profile, which acts as a template of user attributes.

A subscriber service is based on the combination of a defined dynamic profile and attributes configured through authentication. Dynamic profiles can include dynamic firewall filters, class-of-service (CoS) settings, and protocol (IGMP) settings that define access limits for subscribers and the scope of a service granted to the subscriber after access is obtained.

The subscriber access feature provides the following convenience and flexibility to service providers and subscribers:

- Service providers can separate services and access technology and eliminate unprofitable flat-rate billing. They gain the ability to efficiently design, manage, and deliver services that subscribers want, and then bill subscribers based on connect time, bandwidth, and the actual service used.
- Subscribers benefit by gaining access to multiple simultaneous services. Depending on the service provider configuration, subscribers can dynamically connect to and disconnect from various services when they want and for however long they want.

Subscribers can be billed based on the service level and usage, rather than being charged a set rate regardless of usage.

Subscriber Access Terms and Acronyms

[Table 3 on page 4](#) defines terms and acronyms that are used in this discussion of subscriber access.

Table 3: Subscriber Access Terms and Acronyms

Term	Definition
AAA method for subscriber authentication	The AAA method that uses authentication (for example, including RADIUS VSAs in the Access-Accept packet) to verify a subscriber and activate a service when the subscriber logs in.
Dynamic profile	A template that defines a set of characteristics that are combined with authorization attributes and are dynamically assigned to static interfaces to provide dynamic subscriber access and services for broadband applications.
RADIUS CoA method	The method that uses RADIUS CoA-Request messages and VSAs to activate a service for a subscriber that is already logged in.
Subscriber access technology	The technology used by a subscriber to access services (for example, DHCP).

Related Documentation

- [Subscriber Access Network Overview](#)
- [Subscriber Access Licensing Overview on page 9](#)
- [Subscriber Access Operation Flow Using DHCP Relay](#)
- [Configuring Subscriber Access on page 5](#)

AAA Service Framework and Subscriber Management Overview

You use AAA Service Framework for authentication, authorization, accounting, address assignment, and dynamic services request that the BNG uses for network access. The framework supports authentication and authorization through external servers, such as RADIUS. The framework also supports accounting and dynamic-request CoA and disconnect operations through external servers, and address assignment through a combination of local address-assignment pools and RADIUS.

The BNG interacts with external servers to determine how individual subscribers access the broadband network. The router also obtains information from external servers for the following:

- Methods used for authentication and accounting.
- How accounting statistics are collected and used.
- How dynamic requests are handled.

- Related Documentation**
- *AAA Service Framework Overview*
 - *RADIUS-Initiated Change of Authorization (CoA) Overview*
 - *RADIUS-Initiated Disconnect Overview*

Class of Service and Subscriber Management Overview

Class of service (CoS) enables you to divide traffic into classes and offer various levels of throughput and acceptable packet loss when congestion occurs. CoS also provides the option of using differentiated services when best-effort traffic delivery is insufficient. You can also configure the services router to provide hierarchical scheduling for subscribers by dynamically adding or deleting queues when subscribers require services.

By using a dynamic profile, you can provide all subscribers in your network with default CoS parameters when they log in. For example, you can configure an access dynamic profile to specify that all subscribers receive a basic data service. If you use RADIUS variables in the dynamic profile, you can enable the service to be activated for those subscribers at login. You can also use variables to configure a service profile that enables subscribers to activate a service or upgrade to different services through RADIUS change-of-authorization (CoA) messages following initial login.

- Related Documentation**
- *CoS for Subscriber Access Overview*

Configuring Subscriber Access

To configure subscriber access:

1. Configure the client access protocol.
 - Configure DHCP local server.
See Extended DHCP Local Server Overview.
 - Configure DHCP relay.
See Extended DHCP Relay Agent Overview.
 - Configure PPP.
See Configuring Logical Interface Properties and Configuring PPPoE
2. Configure subscriber authentication, accounting, and addressing.
 - a. Configure RADIUS:
 1. Specify the RADIUS servers.
See Specifying RADIUS Authentication and Accounting Servers for Subscriber Access.
 2. Specify any optional server attributes.
See Configuring RADIUS Server Options for Subscriber Access.

3. (Optional) Configure the CoA feature for the RADIUS dynamic-request server to change or deactivate the service after login.

See *Configuring RADIUS-Initiated Dynamic Request Support*.

4. Configure subscriber accounting (RADIUS accounting).

See *Configuring Per-Subscriber Session Accounting*.

- b. Configure addressing:

- See *Configuring Address-Assignment Pools*.

3. Create and manage dynamic profiles for access and service.

- a. Configure a basic dynamic profile.

See [“Configuring a Basic Dynamic Profile” on page 16](#).

See *Example: Minimum PPPoE Dynamic Profile*

- b. Configure a dynamic profile for access.

See *Configuring Dynamic DHCP Client Access to a Multicast Network*.

- c. Configure a dynamic profile for services.

See *Defining Various Levels of Services for DHCP Subscribers*.

- d. Configure a default subscriber service.

See *Configuring a Default Subscriber Service*.

- e. Configure the static subscriber interfaces to be referenced in the dynamic profile.

- f. Specify the interface-name and unit variables that the router uses to dynamically associate to a subscriber's incoming interface.

- g. Add, modify, or delete dynamic profile values to manage subscriber access and services.

The router dynamically activates or modifies the subscriber service using the RADIUS configuration.

- When the subscriber logs in, the router dynamically activates the service.

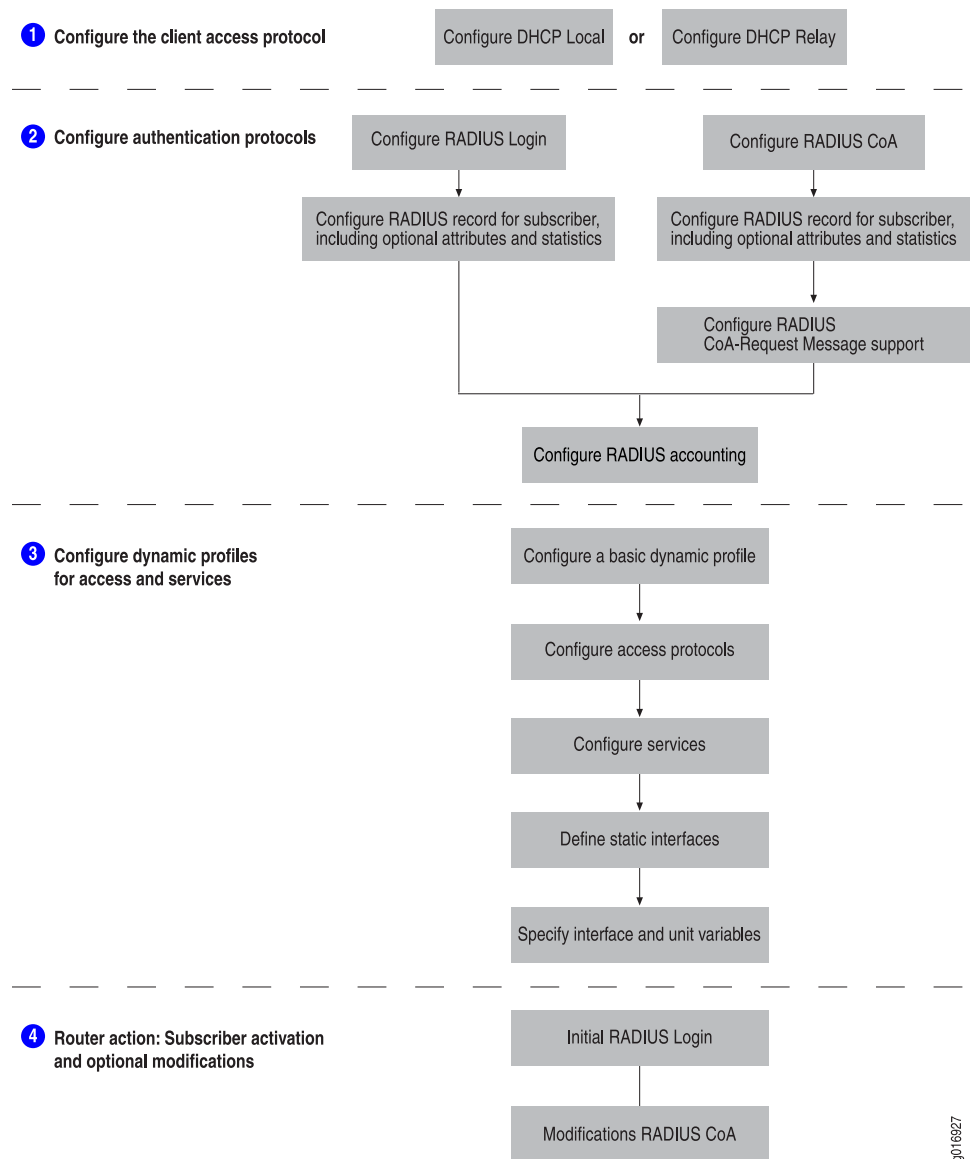
See *Dynamic Service Activation During Login Overview*.

- If RADIUS CoA has been configured, the router can dynamically modify the service for a subscriber.

See *RADIUS-Initiated Change of Authorization (CoA) Overview*.

[Figure 1 on page 7](#) shows the configuration sequence you perform for DHCP-based subscriber access. It also shows the dynamic configuration performed by the router.

Figure 1: Subscriber Access Configuration Workflow



g016927

- Related Documentation**
- [Subscriber Management Overview on page 3](#)
 - *Default Subscriber Service Overview*
 - *CLI-Activated Subscriber Services*

Subscriber Activation and Service Management in an Access Network

The subscriber access feature uses dynamic profiles to activate subscribers and manage services.

A dynamic profile is a set of characteristics, defined in a template, that the router uses to provide dynamic subscriber access and services.

By using dynamic profiles you can:

- Define access for your network
- Define different service levels for subscribers
- Preprovision services that you can activate later

Using AAA-based login (RADIUS-based login or RADIUS CoA) you can:

- Provide subscribers with dynamic activation and deactivation based on service selection
- Provide greater flexibility and efficient management for a large number of subscribers and services

Components of a Dynamic Profile

You can use dynamic profiles to define various router components for subscriber access.

These components include the following:

- Dynamic firewall filters—Includes input and output filters to enforce rules that define whether to permit or deny packets that are transmitting an interface on the router. To apply dynamic firewall filters to the subscriber interface, you configure static input and output firewall filters and reference those filters in dynamic profiles.
- Dynamic Class of Service (CoS)—Includes CoS values that define a service for a subscriber. For example, you can configure the shaping rate for traffic in a video service by referencing CoS statements in a dynamic profile.
- Dynamic signaling protocol—Includes dynamic IGMP configuration for host to router signaling for IPv4 to support IP multicasting.

Router Predefined Variables Used by Dynamic Profiles

The router contains many predefined variables. These variables enable dynamic association of certain interface-specific values to incoming subscriber requests. You must specify these predefined variables in certain statements within a dynamic profile. When a client accesses the router, the dynamic profile configuration replaces the predefined variable with the actual data from an incoming client data packet and configuration (local and RADIUS).

Related Documentation

- [Dynamic Profiles Overview on page 13](#)
- [DHCP Subscriber Interface Overview](#)
- [Junos OS Predefined Variables on page 21](#)

CHAPTER 2

Subscriber Management Licensing

- [Subscriber Access Licensing Overview on page 9](#)
- [Configuring the Router to Strictly Enforce the Subscriber Scaling License on page 9](#)

Subscriber Access Licensing Overview

To enable some Juniper Networks Junos OS features or router scaling levels, you might have to purchase, install, and manage separate software license packs. The presence on the router of the appropriate software license keys (passwords) determines whether you can configure and use certain features or configure a feature to a predetermined scale.

Related Documentation

- [Configuring the Router to Strictly Enforce the Subscriber Scaling License on page 9](#)
- For information about installing and managing Junos OS licenses, see the *Installation and Upgrade Guide*

Configuring the Router to Strictly Enforce the Subscriber Scaling License

You can configure the router to strictly enforce the subscriber scaling feature, which is part of the Junos Subscriber Access Feature Pack license. The subscriber scaling feature specifies the maximum number of subscribers that can be logged in at any one time.

When you configure strict scaling license support, the router performs the following actions:

- Strictly enforces the subscriber scaling license and does not allow any grace period. When the number of logged-in subscriber reaches the number allowed by the scaling license, no additional subscribers are allowed to log in.
- Creates the informational log message, "90 percent of installed subscriber scale licenses in use" in `/var/log/messages`, to inform you when you have 10 percent of the total allowed licenses remaining. The router clears this condition when license usage falls below 90 percent. The log message is created again if the 90 percent usage is later reached.

To configure the router to strictly enforce the subscriber scaling license:

1. Specify that you want to configure subscriber management.

```
[edit system services]  
user@host# edit subscriber-management
```

2. Configure the router to enforce the scaling license.

```
[edit system services subscriber-management]  
user@host# set enforce-strict-scale-limit-license
```

**Related
Documentation**

- [Subscriber Access Licensing Overview on page 9](#)
- *Junos OS Feature Licenses*
- *Verifying Junos OS License Installation*

PART 2

Creating Dynamic Configurations for Subscriber Management

- [Configuring Dynamic Profiles for Subscriber Management on page 13](#)
- [Using Variables in Dynamic Profiles on page 19](#)
- [Using Predefined Variables in Dynamic Profiles on page 21](#)
- [Creating Customized Variables in Dynamic Profiles on page 51](#)
- [Modifying Dynamic Profiles That Are in Use by Active Subscribers on page 63](#)

CHAPTER 3

Configuring Dynamic Profiles for Subscriber Management

- [Dynamic Profiles Overview on page 13](#)
- [Configuring a Basic Dynamic Profile on page 16](#)
- [Access Profiles and Service Profiles Overview on page 17](#)

Dynamic Profiles Overview

A dynamic profile is a set of characteristics, defined in a type of template, that you can use to provide dynamic subscriber access and services for broadband applications. These services are assigned dynamically to interfaces. The **dynamic-profiles** hierarchy appears at the top level of the CLI hierarchy and contains many Juniper Networks configuration statements that you normally define statically.

Dynamic profile statements appear in the following subhierarchies within the **[edit dynamic-profiles]** hierarchy:

- **class-of-service**
- **firewall**
- **interfaces**
- **predefined-variable-defaults**
- **protocols**
- **routing-instances**
- **routing-options**
- **variables**

This topic covers:

- [Dynamic Profile Interface Support on page 14](#)
- [What Dynamic Profiles Do on page 14](#)
- [How Dynamic Profiles Work on page 14](#)
- [Dynamic Profile Overrides on page 14](#)

- [Dynamic Profile Version Creation on page 14](#)
- [Dynamic Profile Semantic Checking on page 15](#)

Dynamic Profile Interface Support

You can identify subscribers statically or dynamically. To identify subscribers statically, you can reference a static VLAN interface in a dynamic profile. To identify subscribers dynamically, you create variables for demux interfaces that are dynamically created when subscribers log in.

What Dynamic Profiles Do

A dynamic profile acts as a kind of template that enables you to create, update, or remove a configuration that includes client access (for example, interface or protocol) or service (for example, class of service (CoS)) attributes or objects that are created dynamically (for example pseudowire). Using these profiles enables you to consolidate all of the common attributes of a client (and eventually a group of clients) and apply the attributes or dynamically created objects simultaneously.

How Dynamic Profiles Work

After profiles are created, they reside on the router in a profile library. These profiles can contain various configurations. For example, you can create a client network access configuration, a services activation configuration, or both. When a router interface receives a join message from a client, the router applies the values configured in the specified dynamic profile to that router interface. The profile can contain interface, CoS, and protocol values that are applied directly to the interface. In addition, the dynamic profile can call input or output firewall filters that reside outside of the dynamic profiles hierarchy.

Dynamic Profile Overrides

Starting in Junos OS Release 14.1, you can specify a different dynamic profile in the RADIUS Client-Profile-Name VSA [26–174] to have RADIUS override a configured client dynamic profile. RADIUS returns this VSA to AAA with other client session attributes in the Access-Accept message. AAA subsequently overrides the corresponding profile name attribute in the session database entry for the client, and this new profile is instantiated instead of the originally configured profile.

Dynamic Profile Version Creation

You can create new versions of dynamic profiles that are currently in use by subscribers. Dynamic profile version creation is enabled at the **[edit system]** hierarchy level. When enabled, you can create multiple versions of any dynamic profiles on the router. Any subscriber that logs in following a dynamic profile modification uses the latest version of the dynamic profile. Subscribers that are already active continue to use the older version of the dynamic profile until they log out or their session terminates.

When creating versions of dynamic profiles, keep the following in mind:

- You must enable or disable dynamic profile version creation before creating or using any dynamic profiles on the router. Enabling or disabling dynamic profile version creation after dynamic profiles are configured is not supported.



NOTE: Before you can enable or disable dynamic profile version creation for a router on which any dynamic profiles are configured, you must first remove all dynamic profiles from the router configuration.

- Each version of a dynamic profile is stored in the profile database as a new profile.
- The name of the new profile version is derived by appending a string to the original base dynamic profile name. This string contains two dollar sign (\$) characters to identify the version field of the profile name. These two characters are followed by numerical characters that represent the “version number” of the dynamic profile (for example, 01).
- The version number of the dynamic profile is automatically generated by the system.
- The dynamic profile that you modify is always stored as the latest version. You cannot create a modified dynamic profile and save it as an earlier version. For example, if you modify version three of a dynamic profile while it is in use, the dynamic profile is saved as version four.
- You can only modify the latest version of a dynamic profile.
- The maximum value for the version number is 99999. However, for each profile, only 10 active versions are supported at a time.
- If the dynamic profile version that you modify is not in use by any subscriber, the profile is overwritten with committed changes without creating a new version.
- After reaching the 99999th modified version of a dynamic profile, any further modifications to the dynamic profile result in overwriting that final version. If the final version is in use, any modification attempts fail upon commit.
- You can delete a dynamic profile only when none of its versions are in use.
- The dynamic profile version feature supports graceful restart and unified ISSU.

Dynamic Profile Semantic Checking

Variables are applied to dynamic profiles dynamically and cannot be checked with existing CLI commands. Semantic checking validates some variables in dynamic profiles to help identify potential configuration errors.

Semantic checks are performed during commit and during profile instantiation. Commit time checks ensure that variables appear in the correct location within the dynamic profile. Checks performed before profile instantiation ensure that the values that replace the variables are correct. The checks performed on the values include the following:

- Range validation
- Variable type validation
- Existence of variables where they are mandatory
- Variable matching to regular expressions

A commit time check failure results in an error message being displayed and logged in the `/var/log/messages` file and the commit failing. An instantiation failure results in an error being logged in the `/var/log/messages` file and the profile instantiation failing.

Release History Table

Release	Description
14.1	Starting in Junos OS Release 14.1, you can specify a different dynamic profile in the RADIUS Client-Profile-Name VSA [26–174] to have RADIUS override a configured client dynamic profile.

Related Documentation

- [Configuring a Basic Dynamic Profile on page 16](#)
- [Enabling Dynamic Profiles to Use Multiple Versions on page 63](#)
- [Dynamic Variables Overview on page 19](#)

Configuring a Basic Dynamic Profile

This topic describes how to create a basic dynamic profile. A basic profile must contain a profile name and have both an interface variable name (such as `$junos-interface-ifd-name`) included at the `[edit dynamic-profiles profile-name interfaces` hierarchy level and logical interface variable name (such as `$junos-underlying-interface-unit` or `$junos-interface-unit`) at the `[edit dynamic-profiles profile-name interfaces variable-interface-name unit]` hierarchy level.

Before you configure dynamic profiles for initial client access:

1. Configure the necessary router interfaces that you want DHCP clients to use when accessing the network.
See [DHCP Subscriber Interface Overview](#) for information about the types of interfaces you can use with dynamic profiles and how to configure them.
2. Configure all RADIUS values that you want the profiles to use when validating DHCP clients for access to the multicast network.
See [Configuring RADIUS Server Parameters for Subscriber Access](#)

To configure a basic dynamic profile:

1. Name the profile.

```
[edit]
user@host# edit dynamic-profiles basic-profile
```

2. Define the **interface-name** statement with the internal **\$junos-interface-ifd-name** variable used by the router to match the interface name of the receiving interface.

```
[edit dynamic-profiles basic-profile]
user@host# edit interfaces $junos-interface-ifd-name
```

3. Define the **unit** statement with the internal variable:

- When referencing an existing interface, specify the **\$junos-underlying-interface-unit** variable used by the router to match the unit value of the receiving interface.
- When creating dynamic interfaces, specify the **\$junos-interface-unit** variable used by the router to generate a unit value for the interface.

```
[edit dynamic-profiles basic-profile interfaces "$junos-interface-ifd-name"]
user@host# set unit $junos-underlying-interface-unit
```

or

```
[edit dynamic-profiles basic-profile interfaces "$junos-interface-ifd-name"]
user@host# set unit $junos-interface-unit
```

Related Documentation

- [Dynamic Profiles Overview on page 13](#)
- [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)
- [Dynamic Variables Overview on page 19](#)
- [Junos OS Predefined Variables on page 21](#)

Access Profiles and Service Profiles Overview

Dynamic profiles enable you to configure parameters that enable access and services to subscribers.

Access profiles, also known as client profiles, contain the parameters to grant access and provide basic service to a subscriber during initial login. By configuring the access profile with Junos OS predefined variables, you also enable the service to be activated for those subscribers at login. The RADIUS variables in an access profile map to one or more VSAs.

Service profiles contain parameters that activate or deactivate services for a subscriber. You can apply a service profile with an access profile at login, or apply the service profile separately to modify a service. A service profile maps to a Service VSA.

Functionality Supported in Access and Service Profiles

The types of variables, expressions, and default values that you can use depends on the type of dynamic profile.

[Table 4 on page 18](#) lists the types of variables supported by access profiles and service profiles.

Table 4: Types of Variables Supported in Dynamic Profiles

Type of Dynamic Profile	Junos OS Predefined Variable (Local)	Junos OS Predefined Variable (RADIUS)	User-Defined Variable
Access Profile	Yes	Yes	Yes
Service Profile	Yes	No	Yes

[Table 5 on page 18](#) lists the default values, expressions, and unique identifiers supported by access profiles and service profiles.

Table 5: Default Values and Expressions Supported in Dynamic Profiles

Type of Dynamic Profile	Default Values	Expressions	Unique Identifiers
Access Profile	Yes (RADIUS predefined variables only)	No	Yes (Schedulers and Scheduler maps only)
Service Profile	Yes (User-defined variables only)	Yes (Service activation only)	Yes (Firewall filters only)

Related Documentation

- [Dynamic Profiles Overview on page 13](#)
- [Using Variable Expressions in User-Defined Variables on page 53](#)
- [Unique Identifiers for Firewall Variables](#)

CHAPTER 4

Using Variables in Dynamic Profiles

- [Dynamic Variables Overview on page 19](#)

Dynamic Variables Overview

Variables constitute the dynamic component of a dynamic profile. You use variables in dynamic profiles as placeholders for dynamically obtained or dynamically generated values that the dynamic profiles use to configure subscriber interfaces and provision subscribers.

- [How Dynamic Variables Work on page 19](#)
- [Default Values for Predefined Variables on page 20](#)
- [Unique Identifier \(UID\) for Parameterized Filters on page 20](#)

How Dynamic Variables Work

Dynamic variables are data placeholders that you define and place in dynamic profiles. When a particular event occurs on an interface (for example, a DHCP client accesses the interface), the dynamic profiles obtain data to fill these placeholders from one of three sources—the interface receiving an incoming client data packet, an externally configured server (for example, RADIUS), or a value associated with each user-configurable variable.

For your convenience, Junos OS provides several predefined variables that you can use within a dynamic profile. Most of these variables relate to interface-specific data obtained directly from the interface that receives an incoming client data packets (for example, interface name, interface unit value, and so on). When a client accesses the interface, the router software extracts the necessary interface data, propagates this data to the dynamic profile, and then uses the dynamic profile to configure the interface for the accessing client.

You define user-defined variables for individual dynamic profiles at the **[dynamic-profiles profile-name variables]** hierarchy level. At this hierarchy level, you create an association between a variable value (for example, `$junos-igmp-version`) that appears in the body of the dynamic profile and data associated with that call value that is managed in an externally configured server (for example, a RADIUS VSA managed on a RADIUS server) or defined as a value in the **variables** stanza. When an event occurs on an interface to trigger the instantiation of a dynamic profile for the interface, Junos OS obtains values

for each variable from an external server (for example, from RADIUS authentication and authorization VSAs) during the subscriber authentication process. At run time, the variables are replaced by these actual values and are used to configure the subscriber interface.

Default Values for Predefined Variables

You can optionally configure default values for many of the predefined variables. If the external RADIUS server is not available or the VSA does not contain a value for the predefined variable, Junos OS uses the default values.

When a default value is configured for a variable and RADIUS also returns a value, the system uses the value from RADIUS instead.

Unique Identifier (UID) for Parameterized Filters

Parameterized filters use unique identifiers (UIDs) in dynamic profiles created for services. The generated UIDs enable you to identify and configure separate parameter values for filters with the same variable name. In addition, assigning a UID improves performance of the router.

For service profiles, you can request the generation of an UID for a user-defined variable by including the **uid** statement at the **[dynamic-profiles profile-name variables]** hierarchy level. You then reference the variable name in the filter.

To enable selection of a particular filter in a dynamic profile that contains multiple variables of the same parameter and criteria type, you must indicate that the variable refers to a UID. To configure, include the **uid-reference** statement at the **[dynamic-profiles profile-name variables]** hierarchy level. For example, if the variable **\$in-filter** receives the value of "filter1" from RADIUS, the filter definition named **\$filter** is used.

Related Documentation

- [Junos OS Predefined Variables on page 21](#)
- [User-Defined Variables on page 51](#)
- [Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs on page 42](#)
- [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [*RADIUS Attributes and Juniper Networks VSAs Supported by the AAA Service Framework*](#)

CHAPTER 5

Using Predefined Variables in Dynamic Profiles

- [Junos OS Predefined Variables on page 21](#)
- [Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs on page 42](#)
- [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)
- [Configuring Default Values for Predefined Variables in a Dynamic Profile on page 49](#)

Junos OS Predefined Variables

Junos OS contains many predefined variables. The dynamic profile obtains and replaces values for these variables from an incoming client data packet and configuration (local and RADIUS). These variables are predefined—you use them in the body of a dynamic profile without first having to define the variables at the **[dynamic-profiles profile-name variables]** hierarchy level. [Table 6 on page 21](#) provides a list of predefined variables, their descriptions, and where in the Junos OS hierarchy you can configure them.

Table 6: Junos OS Predefined Variables and Definitions

Variable	Definition
Access and Access-Internal Routes	
<code>\$junos-framed-route-cost</code>	Cost metric of an IPv4 access route. You specify this variable at the [edit dynamic-profiles profile-name routing-options access route address] hierarchy level for the metric statement.
<code>\$junos-framed-route-distance</code>	Distance of an IPv4 access route. You specify this variable at the [edit dynamic-profiles profile-name routing-options access route address] hierarchy level for the preference statement.
<code>\$junos-framed-route-ip-address-prefix</code>	Route prefix of an IPv4 access route. You specify this variable at the [edit dynamic-profiles profile-name routing-options access] hierarchy level for the route statement.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-framed-route-ipv6-address-prefix	Route prefix of an IPv6 access route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access] hierarchy level for the route statement.
\$junos-framed-route-ipv6-cost	Cost metric of an IPv6 access route. You specify this variable with the metric statement at the [edit dynamic-profiles <i>profile-name</i> routing-instances \$junos-routing-instance routing-options rib \$junos-ipv6-rib access route \$junos-framed-route-ipv6-address-prefix] hierarchy level.
\$junos-framed-route-ipv6-distance	Distance of an IPv6 access route. You specify this variable with the preference statement at the [edit dynamic-profiles <i>profile-name</i> routing-instances \$junos-routing-instance routing-options rib \$junos-ipv6-rib access route \$junos-framed-route-ipv6-address-prefix] hierarchy level.
\$junos-framed-route-ipv6-nexthop	IPv6 next-hop address of an access route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access route address] hierarchy level for the next-hop statement.
\$junos-framed-route-ipv6-tag	Tag value of an IPv6 access route. You specify this variable with the tag statement at the [edit dynamic-profiles <i>profile-name</i> routing-instances \$junos-routing-instance routing-options rib \$junos-ipv6-rib access route \$junos-framed-route-ipv6-address-prefix] hierarchy level.
\$junos-framed-route-nexthop	IPv4 next-hop address of an access route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access route address] hierarchy level for the next-hop statement.
\$junos-framed-route-tag	Tag value of an IPv4 access route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access route address] hierarchy level for the tag statement.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-interface-name	<p>Logical interface of an access-internal route. DHCP or PPP supplies this information when the subscriber logs in. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access-internal route address] hierarchy level for the qualified-next-hop statement.</p> <p>This variable is also used for creating dynamic IP demux interfaces.</p>
\$junos-ipv6-rib	<p>Routing table for an IPv6 access route. You specify this variable with the rib statement at the [edit dynamic-profiles <i>profile-name</i> routing-instances \$junos-routing-instance routing-options] hierarchy level.</p> <p>You can use this variable to specify a nondefault routing instance for the route.</p>
\$junos-subscriber-ip-address	<p>IP address of a subscriber identified in an access-internal route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access-internal] hierarchy level for the route statement.</p> <p>This variable is also used for creating dynamic IP demux interfaces.</p>
\$junos-subscriber-mac-address	<p>MAC address for a subscriber identified in an access-internal route. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> routing-options access-internal route address qualified-next hop underlying-interface] hierarchy level for the mac-address statement.</p>
Dynamic Protocols	
\$junos-igmp-access-group-name	Specifies the access list to use for the source (S) filter.
\$junos-igmp-access-source-group-name	Specifies the access list to use for the source-group (S,G) filter.
\$junos-igmp-enable	<p>Ensures that IGMP is not disabled on the interface by an AAA-based authentication and management method (for example, RADIUS). You specify this variable at the [dynamic-profiles <i>profile-name</i> protocols igmp] hierarchy level for the interface statement.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
<code>\$junos-igmp-immediate-leave</code>	Enables IGMP immediate leave on the interface. You specify this variable at the [dynamic-profiles profile-name protocols igmp] hierarchy level for the interface statement.
<code>\$junos-igmp-version</code>	IGMP version configured in a client access profile. Junos OS obtains this information from the RADIUS server when a subscriber accesses the router. The version is applied to the accessing subscriber when the profile is instantiated. You specify this variable at the [dynamic-profiles profile-name protocols igmp] hierarchy level for the interface statement.
<code>\$junos-interface-name</code>	<p>Name of the dynamic interface to which the subscriber access client connects. Its use is in dynamically enabling IGMP on the subscriber interface. You specify this variable at the [dynamic-profiles profile-name protocols igmp] hierarchy level for the interface statement.</p> <p>The interface name is derived from concatenating the <code>\$junos-interface-afd-name</code> and the <code>\$junos-underlying-interface-unit</code> variables obtained when a subscriber is created dynamically at the [dynamic-profiles profile-name interfaces] hierarchy level.</p>
<code>\$junos-ipv6-ndra-prefix</code>	Prefix value for the router advertisement interface. Junos OS obtains this information from the RADIUS server when a subscriber accesses the router. The prefix value is applied to the accessing subscriber when the profile is instantiated. You specify this variable at the [dynamic-profiles profile-name protocols router-advertisement interface \$junos-interface-name] hierarchy level.
<code>\$junos-mld-access-group-name</code>	Specifies the access list to use for the group (G) filter.
<code>\$junos-mld-access-source-group-name</code>	Specifies the access list to use for the source-group (S,G) filter.
<code>\$junos-mld-enable</code>	Ensures that MLD is not disabled on the interface by an AAA-based authentication and management method (for example, RADIUS). You specify this variable at the [dynamic-profiles profile-name protocols mld] hierarchy level for the interface statement.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-mld-immediate-leave	Enables MLD immediate leave on the interface. You specify this variable at the [dynamic-profiles profile-name protocols mld] hierarchy level for the interface statement.
\$junos-mld-version	MLD version configured in a client access profile. Junos OS obtains this information from the RADIUS server when a subscriber accesses the router. The version is applied to the accessing subscriber when the profile is instantiated. You specify this variable at the [dynamic-profiles profile-name protocols mld] hierarchy level for the interface statement.
Dynamic CoS — Traffic-Control Profile Parameters	
\$junos-cos-adjust-minimum	<p>Minimum adjusted shaping rate configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the adjust-minimum statement at the [edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name] hierarchy level.</p>
\$junos-cos-byte-adjust	<p>Byte adjustment value configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the bytes option with the overhead-accounting statement at the [edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name] hierarchy level.</p>
\$junos-cos-byte-adjust-cell	<p>Overhead bytes when downstream ATM traffic is in cell-mode.</p> <p>NOTE: Do not configure the \$junos-cos-byte-adjust-cell variable when the \$junos-cos-byte-adjust variable is configured.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-byte-adjust-frame	<p>Overhead bytes when downstream ATM traffic is in frame-mode.</p> <p>NOTE: Do not configure the \$junos-cos-byte-adjust-frame variable when the \$junos-cos-byte-adjust variable is configured.</p>
\$junos-cos-delay-buffer-rate	<p>Delay-buffer rate configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the delay-buffer-rate statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>
\$junos-cos-excess-rate	<p>Excess rate configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the excess-rate statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>
\$junos-cos-excess-rate-high	<p>Rate configured for excess high-priority traffic in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the excess-rate-high statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-excess-rate-low	<p>Rate configured for excesslow-priority traffic in a traffic-control profile in a dynamic profile for subscriber access. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the excess-rate-low statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>
\$junos-cos-guaranteed-rate	<p>Guaranteed rate configured in a traffic-control profile in a dynamic profile Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the guaranteed-rate statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>
\$junos-cos-guaranteed-rate-burst	<p>Burst size for the guaranteed rate that is configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable with the burst-size option in the guaranteed-rate statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-scheduler-map	<p>Scheduler-map name configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the scheduler-map statement at the [edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name] hierarchy level.</p> <p>NOTE: The scheduler map can be defined dynamically (at the [edit dynamic-profiles profile-name class-of-service scheduler-maps] hierarchy level) or statically (at the [edit class-of-service scheduler-maps] hierarchy level).</p>
\$junos-cos-shaping-mode	<p>Shaping mode configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the overhead-accounting statement at the [edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name] hierarchy level.</p>
\$junos-cos-shaping-rate	<p>Shaping rate configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the shaping-rate statement at the [edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name] hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-shaping-rate-burst	<p>Burst size for the shaping rate configured in a traffic-control profile in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable with the burst-size option in the shaping-rate statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service traffic-control-profiles <i>profile-name</i>] hierarchy level.</p>
\$junos-cos-shaping-rate-excess-high	Shaping rate configured for excess high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-excess-high-burst	Shaping rate burst size configured for excess high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-excess-low	Shaping rate configured for excess low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-excess-low-burst	Shaping rate burst size configured for excess low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-priority-high	Shaping rate configured for high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-priority-high-burst	Shaping rate burst size configured for high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-shaping-rate-priority-low	Shaping rate configured for low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-priority-low-burst	Shaping rate burst size configured for low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-priority-medium	Shaping rate configured for medium-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-shaping-rate-priority-medium-burst	Shaping rate burst size configured for medium-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
\$junos-cos-traffic-control-profile	<p>Traffic-control profile configured in a dynamic profile for subscriber access. The Junos OS obtains the profile information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the traffic-control-profiles statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service] hierarchy level.</p>
Dynamic CoS — Scheduler Parameters	
\$junos-cos-scheduler	<p>Name of a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable at the [edit dynamic-profiles <i>profile-name</i> class-of-service schedulers] hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
<code>\$junos-cos-scheduler-bs</code>	<p>Buffer size as a percentage of total buffer, specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the buffer-size statement with the percent option at the [edit dynamic-profiles <i>profile-name</i> class-of-service schedulers <i>scheduler-name</i>] hierarchy level.</p>
<code>\$junos-cos-scheduler-pri</code>	<p>Packet-scheduling priority value specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the priority statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service schedulers <i>scheduler-name</i>] hierarchy level.</p>
<code>\$junos-cos-scheduler-dropfile-any</code>	<p>Name of the drop profile for random early detection (RED) for loss-priority level any specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the drop-profile statement at the [edit dynamic-profiles <i>profile-name</i> class-of-service schedulers <i>scheduler-name</i> drop-profile-map loss-priority <i>any</i> protocol <i>any</i>] hierarchy level.</p> <p>NOTE: The drop profile must be configured statically (at the [edit class-of-service drop-profiles] hierarchy level).</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-scheduler-dropfile-high	<p>Name of the drop profile for random early detection (RED) for loss-priority level high specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the drop-profile statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name drop-profile-map loss-priority high protocol any] hierarchy level.</p> <p>NOTE: The drop profile must be configured statically (at the [edit class-of-service drop-profiles] hierarchy level).</p>
\$junos-cos-scheduler-dropfile-low	<p>Name of the drop profile for random early detection (RED) for loss-priority level low specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the drop-profile statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name drop-profile-map loss-priority low protocol any] hierarchy level.</p> <p>NOTE: The drop profile must be configured statically (at the [edit class-of-service drop-profiles] hierarchy level) for loss-priority low.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-scheduler-dropfile-medium-high	<p>Name of the drop profile for random early detection (RED) for loss-priority level medium-high specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the drop-profile statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name drop-profile-map loss-priority medium-high protocol any] hierarchy level.</p> <p>NOTE: The drop profile must be configured statically (at the [edit class-of-service drop-profiles] hierarchy level).</p>
\$junos-cos-scheduler-dropfile-medium-low	<p>Name of the drop profile for random early detection (RED) for loss-priority level medium-low specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the drop-profile statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name drop-profile-map loss-priority medium-low protocol any] hierarchy level.</p> <p>NOTE: The drop profile must be configured statically (at the [edit class-of-service drop-profiles] hierarchy level).</p>
\$junos-cos-scheduler-excess-priority	<p>Priority value of the excess rate specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the excess-priority statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name] hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-cos-scheduler-excess-rate	<p>Value of the excess rate specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the excess-rate statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name] hierarchy level.</p>
\$junos-cos-scheduler-shaping-rate	<p>Value of the shaping rate specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the shaping-rate statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name] hierarchy level.</p>
\$junos-cos-scheduler-tx	<p>Transmit rate specified for a scheduler configured in a dynamic profile. Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the dynamic profile is attached.</p> <p>You reference this variable in the transmit-rate statement at the [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name] hierarchy level.</p>
Dynamic Connectivity Fault Management Parameters	
\$junos-action-profile	Name of the action profile configured in a dynamic profile.
\$junos-ccm-interval	Continuity check interval time configured in a dynamic profile.
\$junos-loss-threshold	The number of continuity check messages lost before marking the remote MEP as down, configured in a dynamic profile.
\$junos-ma-name-format	Name of the maintenance association name format configured in a dynamic profile.
\$junos-md-name-format	Name of the maintenance domain format configured in a dynamic profile.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-ma-name	Name of the maintenance association configured in a dynamic profile.
\$junos-md-level	Value of 'Level', configured in a dynamic profile.
\$junos-md-name	Name of the maintenance domain configured in a dynamic profile.
\$junos-mep-id	The 'MEP' value configured in the dynamic profile.
\$junos-remote-mep-id	The 'Remote MEP' value configured in the dynamic profile.
Filters — RADIUS-obtained Policies	
\$junos-input-filter	Name of an input filter to be attached; filter name is derived from RADIUS VSA 26-10 (Ingress-Policy-Name) or RADIUS attribute 11 (Filter-ID) to the interface.
\$junos-input-interface-filter	<p>Name of an input filter to be attached to a family any interface; filter name is derived from RADIUS VSA 26-191 (Input-Interface-Filter) to the interface.</p> <p>You can also specify the filter name with the <code>\$junos-input-interface-filter</code> statement at the <code>[edit dynamic-profiles profile-name interfaces interface-name unit logical-interface-number filter input]</code> hierarchy level.</p>
\$junos-input-ipv6-filter	Name of an IPv6 input filter to be attached; filter name is derived from RADIUS VSA 26-106 (IPv6-Ingress-Policy-Name) to the interface.
\$junos-output-filter	Name of an output filter to be attached; filter name is derived from RADIUS VSA 26-11 (Egress-Policy-Name) to the interface.
\$junos-output-interface-filter	<p>Name of an output filter to be attached to a family any interface; filter name is derived from RADIUS VSA 26-191 (Output-Interface-Filter) to the interface.</p> <p>You can also specify the filter name with the <code>\$junos-output-interface-filter</code> statement at the <code>[edit dynamic-profiles profile-name interfaces interface-name unit logical-interface-number filter output]</code> hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-output-ipv6-filter	Name of an IPv6 output filter to be attached; filter name is derived from RADIUS VSA 26-107 (IPv6-Egress-Policy-Name) to the interface.
Subscriber Interfaces — Dynamic Demux Interfaces	
\$junos-interface-ifd-name	<p>Name of the device to which the subscriber access client connects. All interfaces are created on this device. Its primary use is in creating single or multiple subscribers on a statically created interface. You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces] hierarchy level.</p> <p>When creating a logical underlying interface for a dynamic VLAN demux interface, you must also specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces demux0 unit \$junos-interface-unit demux-options underlying-interface] hierarchy level.</p>
\$junos-interface-unit	Creates a unit number assigned to the logical interface. The router supplies this information when the subscriber accesses the network. You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces <i>interface-name</i>] hierarchy level for the unit statement.
\$junos-ipv6-address	Selects the IPv6 address of the interface the subscriber uses. You specify this variable at the [edit dynamic-profiles <i>profile-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i>] , [edit dynamic-profiles <i>profile-name</i> interfaces demux0 unit <i>logical-unit-number</i> family <i>family</i>] , [edit dynamic-profiles <i>profile-name</i> interfaces pp0 unit "\$junos-interface-unit" family <i>family</i>] , and [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family <i>family</i>] hierarchy level for the address statement.
\$junos-loopback-interface	Selects the loopback interface the subscriber uses. You specify this variable at the [dynamic profiles <i>profile-name</i> interfaces demux0 unit "\$junos-interface-unit" family inet] hierarchy level for the unnumbered-address statement.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
<code>\$junos-preferred-source-address</code>	<p>Selects the preferred IPv4 source address (family inet) associated with the loopback address used for the subscriber. You specify this variable at the <code>[dynamic profiles profile-name interfaces demux0 unit "\$junos-interface-unit" family inet unnumbered-address "\$junos-loopback-interface"]</code> hierarchy level for the <code>preferred-source-address</code> statement.</p> <p>NOTE: Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the <code>\$junos-routing-instance</code> predefined variable, you must not configure an IPv4 preferred source address. This constraint applies whether you use the <code>\$junos-preferred-source-address</code> predefined variable or the <code>preferred-source-address</code> statement. Configuring the preferred source address in this circumstance causes a commit failure.</p>
<code>\$junos-preferred-source-ipv6-address</code>	<p>Selects the preferred IPv6 source address (family inet6) associated with the loopback address used for the subscriber. You specify this variable at the <code>[dynamic profiles profile-name interfaces demux0 unit "\$junos-interface-unit" family inet6 unnumbered-address "\$junos-loopback-interface"]</code> hierarchy level for the <code>preferred-source-address</code> statement.</p> <p>NOTE: Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the <code>\$junos-routing-instance</code> predefined variable, you must not configure an IPv6 preferred source address. This constraint applies whether you use the <code>\$junos-preferred-source-ipv6-address</code> predefined variable or the <code>preferred-source-address</code> statement. Configuring the preferred source address in this circumstance causes a commit failure.</p>
<code>\$junos-subscriber-ip-address</code>	<p>IP address of the subscriber. You specify this variable at the <code>[dynamic-profiles profile-name interfaces demux0 unit family inet demux-source]</code> hierarchy level.</p> <p>This variable is also used for creating access-internal routes.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-subscriber-ipv6-address	IPv6 address for subscriber. You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family inet6 demux-source] hierarchy level.
\$junos-subscriber-ipv6-multi-address	<p>Expands the demux-source into multiple addresses; for example, the IPv6 prefix and /128 address for the subscriber.</p> <p>You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family inet6 demux-source] hierarchy level.</p>
\$junos-underlying-interface	<p>Creates a logical underlying interface for a dynamic IP demux interface. The client logs in on this interface. You specify this variable at the [dynamic profiles <i>profile-name</i> interfaces demux0 unit "\$junos-interface-unit" demux-options] hierarchy level for the underlying-interface statement.</p> <p>When configured, the underlying interface is used to determine the \$junos-underlying-interface, \$junos-underlying-interface-unit, and \$junos-ifd-name variables. For example, if the receiving logical interface is ge-0/0/0.1, the \$junos-underlying-interface variable is set to ge-0/0/0 and the \$junos-underlying-interface-unit variable is set to 1.</p> <p>This variable is also used for creating access-internal routes.</p>
Subscriber Interfaces — Static VLAN Interfaces	
\$junos-interface-ifd-name	Name of the device to which the subscriber access client connects. All interfaces are created on this device. Its primary use is in creating single or multiple subscribers on a statically created interface. You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces] hierarchy level.
\$junos-underlying-interface-unit	Obtains the unit number for the underlying interface. It specifies the use of the underlying interface for the subscriber. You specify this variable at the [dynamic-profiles <i>profile-name</i> interfaces <i>\$junos-interface-ifd-name</i>] hierarchy level for the unit statement.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
Subscriber Interfaces — Dynamic PPPoE Interfaces	
<code>\$junos-interface-unit</code>	Specifies the logical unit number when the router dynamically creates a PPPoE logical interface. The <code>\$junos-interface-unit</code> predefined variable is dynamically replaced with the unit number supplied by the network when the PPPoE subscriber logs in. You specify this variable at the <code>[edit dynamic-profiles profile-name interfaces pp0]</code> hierarchy level for the <code>unit</code> statement.
<code>\$junos-underlying-interface</code>	Specifies the name of the underlying Ethernet interface on which the router dynamically creates the PPPoE logical interface. The <code>\$junos-underlying-interface</code> predefined variable is dynamically replaced with the name of the underlying interface supplied by the network when the PPPoE subscriber logs in. You specify this variable at the <code>[edit dynamic-profiles profile-name interfaces pp0 unit "\$junos-interface-unit" pppoe-options]</code> hierarchy level for the <code>underlying-interface</code> statement.
Subscriber Interfaces — Dynamic Interface Sets	
<code>\$junos-interface-set-name</code>	Name of an interface set configured in a dynamic profile. To represent the name of a dynamically created agent circuit identifier (ACI) interface set, use the <code>\$junos-interface-set-name</code> predefined variable in the <code>interface-set</code> statement at the <code>[edit dynamic-profiles profile-name interfaces]</code> hierarchy level.

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
<code>\$junos-phy-ifd-interface-set-name</code>	<p>Name of an interface set associated with the underlying physical interface in a dynamic profile.</p> <p>In a heterogeneous topology where residential and business subscribers share the same physical interface, although only two levels of CoS are required for residential access, business access requires three levels. Because they share the same physical interface, three levels are configured for both, causing an unnecessary level 2 node to be consumed for each residential connection.</p> <p>Starting in Junos OS Release 16.1, you can reduce the CoS resources wasted on residential access by collecting the residential subscribers into an interface set associated with the physical interface. In this way, a level 2 node is used for the interface set rather than for each residential interface. To do so, specify the <code>\$junos-phy-ifd-interface-set-name</code> predefined variable with the <code>interface-set</code> statement at the <code>[edit dynamic-profiles <i>profile-name</i> interfaces]</code> hierarchy level to create the interface set based on the underlying physical interface.</p>
<code>\$junos-svlan-interface-set-name</code>	<p>Locally generated interface set name for use by dual-tagged VLAN interfaces based on the outer tag of the dual-tagged VLAN. The format of the generated variable is <i>physical_interface_name - outer_VLAN_tag</i>.</p>
Wholesale Networking	
<code>\$junos-interface-name</code>	<p>Name of the dynamic interface to which the subscriber access client connects. Its use is in identifying the subscriber interface. You specify this variable at the <code>[dynamic-profiles <i>profile-name</i> routing-instance <i>\$junos-routing-instance</i>]</code> hierarchy level for the <code>interface</code> statement.</p> <p>The interface name is derived from concatenating the <code>\$junos-interface-ifd-name</code> and the <code>\$junos-underlying-interface-unit</code> variables obtained when a subscriber is created dynamically at the <code>[dynamic-profiles <i>profile-name</i> routing-instance <i>\$junos-routing-instance</i> interface]</code> hierarchy level.</p>

Table 6: Junos OS Predefined Variables and Definitions (*continued*)

Variable	Definition
\$junos-routing-instance	<p>Name of the routing instance to which the subscriber is assigned. This variable triggers a return value from the RADIUS server for Virtual-Router (VSA 26-1).</p> <p>You reference this variable in the statement at the [dynamic-profiles <i>profile-name</i>] hierarchy level for the routing-instance statement.</p> <p>NOTE: Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the \$junos-routing-instance predefined variable, you must not configure a preferred source address. This constraint applies whether you use the \$junos-preferred-source-address predefined variable, the \$junos-preferred-source-ipv6-address predefined variable, or the preferred-source-address statement. Configuring the preferred source address in this circumstance causes a commit failure.</p>
\$junos-inner-vlan-map-id	<p>Identifier for the inner VLAN tag for Layer 2 wholesale, ANCP-triggered, autosensed dynamic VLANs. The VLAN tag is allocated from the inner VLAN ID swap ranges that are provisioned on the core-facing physical interface. The inner VLAN tag is swapped with (replaces) the outer VLAN tag when the subscriber traffic is tunneled to the NSP.</p> <p>You specify this variable with the inner-vlan-id statement at the [edit dynamic-profiles <i>profile-name</i> interfaces <i>profile-name</i> interfaces \$junos-interface-ifd-name unit \$junos-interface-unit input-vlan-map] hierarchy level.</p>
\$junos-vlan-map-id	<p>Identifier for a VLAN that is rewritten at the input or output interface as specified by a VLAN map.</p> <p>You specify this variable with the vlan-id statement at the [edit dynamic-profiles <i>profile-name</i> interfaces <i>profile-name</i> interfaces \$junos-interface-ifd-name unit \$junos-interface-unit input-vlan-map] or [edit dynamic-profiles <i>profile-name</i> interfaces \$junos-interface-ifd-name unit \$junos-interface-unit input-vlan-map] hierarchy levels.</p>

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the \$junos-routing-instance predefined variable, you must not configure an IPv4 preferred source address.
16.1	Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the \$junos-routing-instance predefined variable, you must not configure an IPv6 preferred source address.
16.1	Starting in Junos OS Release 16.1, you can reduce the CoS resources wasted on residential access by collecting the residential subscribers into an interface set associated with the physical interface.
16.1	Starting in Junos OS Release 16.1, when you specify a static logical interface for the unnumbered interface in a dynamic profile that includes the \$junos-routing-instance predefined variable, you must not configure a preferred source address.

Related Documentation

- [Dynamic Variables Overview on page 19](#)
- [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)
- [Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs on page 42](#)
- [User-Defined Variables on page 51](#)

Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs

Table 7 on page 42 lists the RADIUS attributes and Juniper Networks VSAs and their corresponding Junos OS predefined variables that are used in dynamic profiles. When the router instantiates a dynamic profile following subscriber access, the Junos OS uses the predefined variable to specify the RADIUS attribute or VSA for the information obtained from the RADIUS server.

Some predefined variables support the configuration of default values. The configured default value is used in the event that RADIUS fails to return a value for the variable. You configure default values with the **predefined-variable-defaults predefined-variable default-value** statement at the **[edit dynamic-profiles]** hierarchy level. When you specify the **predefined-variable**, you use the name of the Junos OS predefined variable, but you omit the leading **\$junos-** prefix.

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
RADIUS Attribute			
Framed-IP-Address (8)	\$junos-framed-route-ip-address	Address for the client	No

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
Filter-ID (11)	\$junos-input-filter NOTE: Variable is also used for VSA 26–10.	Input filter to apply to client IPv4 interface	Yes
Framed-Route (22)	\$junos-framed-route-ip-address-prefix	(Subattribute 1): Route prefix for access route	No
	\$junos-framed-route-nexthop	(Subattribute 2): Next hop address for access route	No
	\$junos-framed-route-cost	(Subattribute 3): Metric for access route	No
	\$junos-framed-route-distance	(Subattribute 5): Preference for access route	No
	\$junos-framed-route-tag	(Subattribute 6): Tag for access route	No
Framed-IPv6-Prefix (97)	\$junos-ipv6-ndra-prefix	Prefix value in IPv6 Neighbor Discovery route advertisements	No
Framed-IPv6-Route (99)	\$junos-framed-route-ipv6-address-prefix	(Subattribute 1): Framed IPv6 route prefix configured for the client	No
	\$junos-framed-route-ipv6-cost	(Subattribute 3): Metric for access route	No
	\$junos-framed-route-ipv6-distance	(Subattribute 5): Preference for access route	No
	\$junos-framed-route-ipv6-nexthop	(Subattribute 2): IPv6 routing information configured for the client	No
	\$junos-framed-route-ipv6-tag	(Subattribute 6): Tag for access route	No
Juniper Networks VSA			
Virtual-Router (26–1)	\$junos-routing-instance	Routing instance to which subscriber is assigned	Yes

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
Ingress-Policy-Name (26–10)	\$junos-input-filter NOTE: Variable is also used for RADIUS attribute 11.	Input filter to apply to client IPv4 interface	Yes
Egress-Policy-Name (26–11)	\$junos-output-filter	Output filter to apply to client IPv4 interface	Yes
IGMP-Enable (26–23)	\$junos-igmp-enable	Enable or disable IGMP on client interface	Yes
IGMP-Access-Name (26–71)	\$junos-igmp-access-group-name	Access list to use for the group (G) filter	Yes
IGMP-Access-Src-Name (26–72)	\$junos-igmp-access-source-group-name	Access List to use for the source group (S,G) filter	Yes
MLD-Access-Name (26–74)	\$junos-mld-access-group-name	Access list to use for the group (G) filter	Yes
MLD-Access-Src-Name (26–75)	\$junos-mld-access-source-group-name	Access List to use for the source group (S,G) filter	Yes
MLD-Version (26–77)	\$junos-mld-version	MLD protocol version	Yes
IGMP-Version (26–78)	\$junos-igmp-version	IGMP protocol version	Yes
IGMP-Immediate-Leave (26–97)	\$junos-igmp-immediate-leave	IGMP immediate leave	Yes
MLD-Immediate-Leave (26–100)	\$junos-mld-immediate-leave	MLD immediate leave	Yes
IPv6-Ingress-Policy-Name (26–106)	\$junos-input-ipv6-filter	Input filter to apply to client IPv6 interface	Yes
IPv6-Egress-Policy-Name (26–107)	\$junos-output-ipv6-filter	Output filter to apply to client IPv6 interface	Yes
CoS-Traffic-Control-Profile-Parameter-Type (26–108)	\$junos-cos-scheduler-map	(T01: Scheduler-map name) Name of scheduler map configured in traffic-control profile	Yes

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
	\$junos-cos-shaping-rate	(T02: Shaping rate) Shaping rate configured in traffic-control profile	Yes
	\$junos-cos-guaranteed-rate	(T03: Guaranteed rate) Guaranteed rate configured in traffic-control profile	Yes
	\$junos-cos-delay-buffer-rate	(T04: Delay-buffer rate) Delay-buffer rate configured in traffic-control profile	Yes
	\$junos-cos-excess-rate	(T05: Excess rate) Excess rate configured in traffic-control profile	Yes
	\$junos-cos-traffic-control-profile	(T06: Traffic-control profile) Name of the traffic-control profile configured in a dynamic profile	Yes
	\$junos-cos-shaping-mode	(T07: Shaping mode) CoS shaping mode configured in a dynamic profile	Yes

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
	\$junos-cos-byte-adjust	(T08; Byte adjust) Byte adjustments configured for the shaping mode in a dynamic profile	Yes
	\$junos-cos-adjust-minimum	(T09; Adjust minimum) Minimum adjusted value allowed for the shaping rate in a dynamic profile	Yes
	\$junos-cos-excess-rate-high	(T10; Excess rate high) Excess rate configured for high-priority traffic in a dynamic profile	Yes
	\$junos-cos-excess-rate-low	(T11; Excess rate low) Excess rate configured for low-priority traffic in a dynamic profile	Yes
	\$junos-cos-shaping-rate-burst	(T12; Shaping rate burst) Burst size configured for the shaping rate in a dynamic profile	Yes
	\$junos-cos-guaranteed-rate-burst	(T13; Guaranteed rate burst) Burst size configured for the guaranteed rate in a dynamic profile	Yes
Qos-Set-Name (26–130)	\$junos-interface-set-name	Name of an interface set configured in a dynamic profile	Yes
CoS-Scheduler-Pmt-Type (26–146)	\$junos-cos-scheduler	(Null: Scheduler name) Name of scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-tx	(T01: CoS scheduler transmit rate) Transmit rate for scheduler configured in a dynamic profile	Yes Available for multiple parameters: <ul style="list-style-type: none"> • Percent • Rate

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
	\$junos-cos-scheduler-bs	(T02: CoS scheduler buffer size) Buffer size for scheduler configured in a dynamic profile	Yes Available for multiple parameters: <ul style="list-style-type: none"> • Percent • Temporal
	\$junos-cos-scheduler-pri	(T03: CoS scheduler priority) Packet-scheduling priority for scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-dropfile-low	(T04: CoS scheduler drop-profile low) Name of drop profile for RED loss-priority level low for scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-dropfile-medium-low	(T05: CoS scheduler drop-profile medium-low) Name of drop profile for RED loss-priority level medium-low for scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-dropfile-medium-high	(T06: CoS scheduler drop-profile medium-high) Name of drop profile for RED loss-priority level medium-high for scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-dropfile-high	(T07: CoS scheduler drop-profile high) Name of drop profile for RED loss-priority level high for scheduler configured in a dynamic profile	Yes

Table 7: RADIUS Attributes and Corresponding Junos OS Predefined Variables (*continued*)

RADIUS Attribute or VSA	Junos OS Predefined Variable	Description	Default Value Support for Junos OS Predefined Variable
	\$junos-cos-scheduler-dropfile-any	(T08: CoS scheduler drop-profile any) Name of drop profile for RED loss-priority level any for scheduler configured in a dynamic profile	Yes
	\$junos-cos-scheduler-excess-rate	(T09: CoS scheduler excess rate) Excess rate configured for a scheduler in a dynamic profile	Yes Available for multiple parameters: <ul style="list-style-type: none"> • Percent • Proportion
	\$junos-cos-scheduler-shaping-rate	(T10: CoS scheduler shaping rate) Shaping rate configured for a scheduler in a dynamic profile	Yes Available for multiple parameters: <ul style="list-style-type: none"> • Percent • Rate
	\$junos-cos-scheduler-excess-priority	(T11: CoS scheduler excess priority) Excess priority configured for a scheduler in a dynamic profile	Yes
Input-Interface-Filter (26-191)	\$junos-input-interface-filter	Name of an input filter to be attached to a family any interface.	No
Output-Interface-Filter (26-192)	\$junos-output-interface-filter	Name of an output filter to be attached to a family any interface.	No

- Related Documentation**
- [Dynamic Variables Overview on page 19](#)
 - [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)
 - [Junos OS Predefined Variables on page 21](#)

Configuring Predefined Dynamic Variables in Dynamic Profiles

This topic discusses how to configure predefined variables in a dynamic profile. The dynamic profile obtains and replaces data for these variables from an incoming client data packet. You can specify these variables in the body of a dynamic profile without having to first define the variables at the **[edit dynamic-profiles profile-name variables]** hierarchy level.

Before you configure dynamic variables:

1. Create a basic dynamic profile.
See [“Configuring a Basic Dynamic Profile” on page 16](#).
2. Ensure that the router hardware is configured in the network to accept subscriber access.

To configure predefined variables in a dynamic profile:

1. Access the desired dynamic profile.

```
[edit]
user@host# edit dynamic-profiles igmpProfile1
[edit dynamic-profiles igmpProfile1]
```
2. Configure the necessary variables.

```
[edit dynamic-profiles igmpProfile1]
user@host# set protocols igmp interface $junos-interface-name
```

For a complete list of supported predefined variables, see [“Junos OS Predefined Variables” on page 21](#).

Related Documentation

- [Configuring a Basic Dynamic Profile on page 16](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [Dynamic Profiles Overview on page 13](#)
- [Dynamic Variables Overview on page 19](#)
- [Junos OS Predefined Variables on page 21](#)
- [Example: Firewall Dynamic Profile](#)
- [Example: IGMP Dynamic Profile](#)

Configuring Default Values for Predefined Variables in a Dynamic Profile

You can configure default values for some Junos OS predefined variables that are configured in a dynamic profile. These default values are used when RADIUS does not supply a value.

To configure default values for Junos predefined variables:

1. Specify that you want to configure the dynamic profile.

```
[edit]
user@host# edit dynamic-profile profile-name
```

2. Configure the default value for a predefined variable or for a specific option within a predefined variable.

```
[edit dynamic-profiles profile-name]
user@host# set predefined-variable-defaults predefined-variable variable-option
default-value
```



NOTE: When you specify the *predefined-variable*, you use the name of the Junos OS predefined variable, but you omit the leading \$junos- prefix.

For example, consider the behavior when you have the following configuration to specify a default value for the \$junos-routing-instance predefined variable:

```
[edit dynamic-profiles prof1]
user@host# set predefined-variable-defaults routing-instances RI-def
```

- When RADIUS does not return a routing instance, the subscribers come up in the RI-def routing instance.
- When RADIUS returns routing-instance RI-res, the subscribers come up in the RI-res routing instance.

When you do not configure a default value for the \$junos-routing-instance predefined variable and RADIUS does not return a value, the subscribers come up in the master routing instance, which is the Junos OS default.

Related Documentation

- For a list of predefined variables and options for which you can configure default values, see [Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs on page 42](#)
- [Junos OS Predefined Variables on page 21](#)
- [Dynamic Variables Overview on page 19](#)

CHAPTER 6

Creating Customized Variables in Dynamic Profiles

- [User-Defined Variables on page 51](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [Using Variable Expressions in User-Defined Variables on page 53](#)
- [Configuring Variable Expressions in Dynamic Profiles on page 56](#)
- [Conditional Configuration for Dynamic Profile Overview on page 58](#)

User-Defined Variables

In service profiles, the Junos OS enables you to configure custom variables at the **[edit dynamic-profiles profile-name variables]** hierarchy level and use those variables in the **[edit dynamic-profiles]** hierarchy. The dynamic profile obtains and replaces data for these variables from an external server (for example, RADIUS) during the subscriber authentication process. At run time, the variables are replaced by actual values and used to configure subscriber interfaces.

You can configure user-defined variable with any of the following statements:

- **default-value** – Configure a default value for a user-defined variable in a dynamic profile. The values that the system uses for these variables are applied when the subscriber authenticates. Specifying a default value provides a standalone configuration for the associated statement or a backup for the statement configuration if the external server is inaccessible or does not contain a value for the variable.
- **equals** – Configure an expression for a user-defined variable that is evaluated at run time and returned as the variable value.
- **mandatory** – Specify that an external server (for example, RADIUS) must return a value for the user-defined variable. If the external server does not return a value for the variable, the dynamic profile fails.



NOTE: The order in which you define how variables are obtained is important. To ensure that you obtain any mandatory variables from an external server, and not derive values from defaults or through variable expressions, you must define any mandatory variables first.

- **uid** – Configure a unique ID for parameterized filters and CoS in a dynamic profile created for services.
- **uid-reference** – Configure a variable that references a unique ID for parameterized filters or CoS in a dynamic profile created for services.

**Related
Documentation**

- [Dynamic Profiles Overview on page 13](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [RADIUS Attributes and Juniper Networks VSAs Supported by the AAA Service Framework](#)
- [Using Variable Expressions in User-Defined Variables on page 53](#)
- [Configuring Variable Expressions in Dynamic Profiles on page 56](#)
- [Junos OS Predefined Variables on page 21](#)

Configuring User-Defined Dynamic Variables in Dynamic Profiles

This topic discusses how to configure a user-defined dynamic variable in a dynamic profile. You can specify a user-defined default for dynamic profile variables at the **[edit dynamic-profiles profile-name variables]** hierarchy level. At this hierarchy level, you create a default value for any dynamic variable that appears in the body of the dynamic profile. The default variable values are used in the event the router is unable to access an external server (for example, RADIUS) or otherwise obtain a value for use as the dynamic variable.

Before you configure any dynamic variable default values:

1. Create a basic dynamic profile.
[See “Configuring a Basic Dynamic Profile” on page 16.](#)
2. Ensure that the router is configured to enable communication between the client and the RADIUS server.

See [Specifying the Authentication and Accounting Methods for Subscriber Access](#).

3. Configure all RADIUS values that you want the profiles to use when validating subscribers.

See [Configuring RADIUS Server Parameters for Subscriber Access](#)

To configure variables in a dynamic profile:

1. Access the **variables** stanza in the desired dynamic profile.

```
[edit]
user@host# edit dynamic-profiles Profile1 variables
```

2. Define the variable.

```
[edit dynamic-profiles Profile1 variables]
user@host# set video-filter equals "Filter1"
```

3. Create a dynamic variable and specify a default value for use by the variable in the event the router cannot contact the external server or if the external server does not contain a value for the assigned attribute.

```
[edit dynamic-profiles Profile1 variables]
user@host# set video-filter default-value Filter_default
```

4. (Optional) Specify that the external server must return a value for a user-defined variable.

```
[edit dynamic-profiles Profile1 variables]
user@host# set video-filter mandatory
```



NOTE: When you include the **mandatory** statement, if the external server does not return a value for the variable, the dynamic profile fails.

Related Documentation

- [Dynamic Profiles Overview on page 13](#)
- [Dynamic Variables Overview on page 19](#)
- [User-Defined Variables on page 51](#)
- [Using Variable Expressions in User-Defined Variables on page 53](#)
- [Configuring Variable Expressions in Dynamic Profiles on page 56](#)
- [Configuring Predefined Dynamic Variables in Dynamic Profiles on page 49](#)

Using Variable Expressions in User-Defined Variables

Junos OS enables you to create expressions—groups of arithmetic operators, string operators, and operands—for use as variables within dynamic profiles. You configure variable expressions at the **[dynamic-profiles profile-name variables]** hierarchy level. At run time, the variable expressions are calculated and used as variable values to configure dynamic subscriber interfaces.

When configuring expressions in dynamic profiles, you must adhere to the following rules:

- You can configure expressions only within a variable stanza of a dynamic profile.
- Dynamic profiles that contain expressions must be used only for service activation.
- You can assign expressions only to user-defined variables. You cannot assign expressions to internal variables or predefined variables.
- Expression values are given precedence over default values.
- Entire expressions must be contained within quotation marks (" ").
- Strings within the expressions must be quoted within single quotation marks (') and the single quotation marks can contain only strings.
- White space is treated as a delimiter for all operands and operators. Strings containing spaces that you create within expressions are treated as single strings and include any leading or trailing white space. For example:

```
dynamic-profiles {
  service profile {
    variables {
      scheduler-name;
      video-filter equals " ' Filter 1 ' " # Everything within the single quotation marks is
      considered a string, including the leading and trailing white space
    }
  }
}
```

- The expression must be either all arithmetic operators or all string operators; mixing arithmetic operators and string operators is not allowed unless properly converted to the correct type.
- Expressions can refer to other system predefined variables or other user-defined variables. However, no circular referencing between variables is allowed. For example, the following reference is incorrect:

```
dynamic-profiles {
  Service_Profile_1 {
    variables {
      scheduler-name;
      transmit-rate2 equals " ( $transmit-rate1 * 2)/3" # refers to transmit-rate1
      transmit-rate1 equals " ( $transmit-rate2 * 2)/3" # refers to transmit-rate2
    }
  }
}
```

- Any mandatory variable that does not contain a "default" value or an "equals" expression must contain a value as a part of service activation. For example, a RADIUS service VSA like "service-video(value1, value2)" that contains two or fewer mandatory variables in the dynamic service profile definition "service-video" succeeds. The service activation fails if at least one mandatory variable does not have any value associated with it, either through default or equals attribute evaluation.

Table 8 on page 54 lists supported operators and functions you can use to create expressions.



NOTE: Precedence 5 is the highest level.

Table 8: Operators and Functions

Operation	Operator	Associativity	Precedence	Action
Arithmetic Addition	+	Left	1	Adds the elements to the right and left of the operator together.
Arithmetic Subtraction	-	Left	1	Subtracts the element to the right of the operator from the element to the left of the operator.
Arithmetic Multiplication	*	Left	2	Multiplies the element to the left of the operator by the element to the right of the operator.

Table 8: Operators and Functions (*continued*)

Operation	Operator	Associativity	Precedence	Action
Arithmetic Division	/	Left	2	Divides the element to the left of the operator by the element to the right of the operator.
Arithmetic Modulo	%	Left	2	Divides the element to the left of the operator by the element to the right of the operator and returns the integer remainder. If the element to the left of the operator is less than the element to the right of the operator, the result is the element to the left of the operator.
Concatenation	##	Left	3	Creates a new string by joining the string values to the left of the operator and the values to the right of the operator together.
Maximum	max(param1,param2)	Left	4	Takes the maximum of the two values passed as parameters.
Minimum	min(param1,param2)	Left	4	Takes the minimum of the two values passed as parameters.
Round	round(param1)	-	4	Rounds the value to the nearest integer.
Truncate	trunc(param1)	-	4	Truncates a non-integer value to the value left of the decimal point.
Convert to String	toStr(param1)	-	4	Converts the variable inside the parentheses to a null terminated string.
Convert to Integer	toInt(param1)	-	4	Converts the parameter to an integer. A single string or variable is allowed as a parameter.
Random	rand()	-	4	Generates a random numerical value.
If Not Zero	ifNotZero(param1, param2)	Left	4	Returns the second parameter if the first parameter is not zero. Returns NULL if first parameter is zero.
Parentheses	()	-	5	Groups operands and operators to achieve results different from simple precedence; effectively has the highest precedence.

Expressions are evaluated after variables are populated with values. The evaluation is conducted immediately before profile instantiation and includes value checking. If the computed values are not acceptable, or rules governing expression syntax are broken, the expression evaluation fails, profile instantiation does not occur, and messages are logged to describe the errors.

[Table 9 on page 56](#) lists the possible expression error scenarios and the action taken by the router software.

Table 9: Expression Errors and Actions

Error	Occurance	Action	Variable Value
Parsing error	Commit check phase	Commit fails	not applicable
Circular variable dependency error	Commit check phase	Commit fails	not applicable
Variables inside the expressions are not defined	Commit check phase	Commit fails	not applicable
Divide by zero	Profile Instantiation	Profile instantiation fails	Zero (0)
Adding string to a number	Profile Instantiation	Profile instantiation fails	Zero (0)
Overflow error	Profile Instantiation	Profile instantiation fails	Undefined
Underflow error	Profile Instantiation	Profile instantiation fails	Undefined

You can also configure the user-defined variables with a default value. The default value provides a standalone configuration for the associated statement or a backup for the statement configuration if the RADIUS server is inaccessible or the VSA attribute does not contain a value.

Related Documentation

- [Configuring Variable Expressions in Dynamic Profiles on page 56](#)
- [Dynamic Profiles Overview on page 13](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [RADIUS Attributes and Juniper Networks VSAs Supported by the AAA Service Framework](#)
- [Junos OS Predefined Variables on page 21](#)

Configuring Variable Expressions in Dynamic Profiles

You can create expressions—groups of arithmetic operators, string operators, and operands—for use as variables within dynamic profiles. These expressions are used as variable values to configure dynamic subscriber interfaces.

To configure dynamic profile variable expressions:

1. Access the dynamic profile for which you want to create variable expressions.

```
[edit]
user@host# edit dynamic-profiles profile-name
```

2. Access the **variables** hierarchy for the dynamic profile.

```
[edit dynamic-profiles profile-name]
user@host# edit variables
```

3. Define the variable using the expression operators and operands described in [“Using Variable Expressions in User-Defined Variables” on page 53](#).

[edit dynamic-profiles *profile-name* variables]
 user@host# **set expression**

Table 10 on page 57 provides several examples of expressions that you can create using the supported operators and functions.

Table 10: Expression Examples

Example	Description
video-filter equals "" Filter1" "	Assigns the string " Filter1" to the dynamic \$video-filter variable.
video-filter2 equals "\$video-filter ## ' Filter2' "	Converts dynamic variable "\$video-filter" to a string and concatenates the new string with the string " Filter2". The result is the string "\$video-filter Filter2" assigned to the \$video-filter2 variable.
tempvar equals "120"	Converts "120" to an integer and assigns the integer to the \$tempvar variable.
transmit-rate2 equals "(\$transmit-rate1 * 2)/3 + \$tempvar)"	Multiplies the "transmit-rate1" variable by 2 and divides that value by the sum of 3 and the value of "\$tempvar". The result is assigned to the \$transmit-rate2 variable.
host-ip equals " 203.0.113.2" "	Assigns the string "203.0.113.2" to the \$host-ip variable.
max-val "max(\$max1,\$max2)"	Assigns the greater of value "max1" or "max2" to the \$max-val variable.
min-val "\$min(\$var1,30)"	Assign the smaller of value "var1" and "30" to the \$min-val variable.
rounded-var equals "round(\$var1)" "	Rounds off the value of the variable "\$var1" to the nearest integer and assigns the value to the \$rounded-var variable.
trunc-var equals "trunc(1234.5)"	Truncates the value in parentheses to the left side of the decimal and assigns the resulting value to the \$trunc-var variable.
bwg-shaping-rate equals "\$anqp-downstream - (\$anqp-downstream % 2 * (1 - \$sp-qos-cell-mode))"	Evaluates the expression as per the precedence set in the parentheses.
temp-filter1 equals " 'Filter1' ## toStr(\$filter)"	Converts the "\$filter" variable to a string value and concatenates the converted string to the string "Filter1". The resulting combined string is assigned to the \$temp-filter1 variable.

Related Documentation

- [Using Variable Expressions in User-Defined Variables on page 53](#)
- [Dynamic Profiles Overview on page 13](#)
- [Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52](#)
- [Junos OS Predefined Variables on page 21](#)

Conditional Configuration for Dynamic Profile Overview

You can configure conditional configuration statements for dynamic profiles to dynamically obtain subscriber information for a client or service.

Conditional configuration involves two main steps:

1. Defining the conditional variable
2. Referencing the conditional variable in a configuration statement

A conditional variable is defined as an expression **ifNotZero (param1, param2)**. In this expression, *param1* is a user-defined variable whose value is derived from an external server such as RADIUS and *param2* can be a user-defined variable, a function, operation, number, or string. A conditional variable can be user-defined or Unique ID (UID) *reference* variable. It cannot be a predefined or UID variable. In Junos OS, conditional variables are supported only for the service dynamic profiles.

The configuration statements in which the conditional variables are referenced are called *conditional* configuration statements. After the conditional variable are defined, they are referenced in **dynamic-profiles** configuration statements and are processed when the service profile is instantiated. The following service profile configuration statements support conditional variables:

- **dynamic-profiles** *profile-name* interfaces *interface-name* unit *unit-no* family *type* filter input *filter-name*
- **dynamic-profiles** *profile-name* interfaces *interface-name* unit *unit-no* family *type* filter output *filter-name*
- **dynamic-profiles** *profile-name* firewall family *type* filter *filter-name* term *term-name*
- **dynamic-profiles** *profile-name* firewall family *type* filter *filter-name* term *term-name* then policer *policer-name*
- **dynamic-profiles** *profile-name* firewall family *type* filter *filter-name* term *term-name* then hierarchical-policer *policer-name*
- **dynamic-profiles** *profile-name* class-of-service scheduler-maps *map-name* forwarding-class *class-name* scheduler *scheduler-name*

The system follows the following set of rules while evaluating the conditional variables and conditional configuration statements during service profile instantiation:

- In the function **ifNotZero(param1, param2)**, if the value of a *param1* is not received from an external server and if the default value is not configured, the value of the variable is treated as non-zero and *param2* is evaluated.
- If the value of *param1* in the function **ifNotZero(param1, param2)** is 0, then NULL is returned as the value of the expression and *param2* is not evaluated. In this case, the value of the conditional variable becomes NULL and the configuration statement in which the conditional variable is referenced is ignored.

- If the value of *param1* is non-zero, then *param2* is evaluated and its value is returned as the value of the expression.

The following **filter-service** and **cos-service** configuration examples show how the rules are applied:

Filter Service Configuration Example

```
filter-service {
  variables {
    input-filter-var mandatory;
    output-filter-var mandatory;
    bw-limit-var mandatory;
    term1-var default-value term1;
    input-filter-ref {
      equals "ifNotZero($input-filter-var,$input-filter-var)";
      uid-reference;
    }
    output-filter-ref {
      equals "ifNotZero($output-filter-var,$output-filter-var)";
      uid-reference;
    }
    policer1-ref {
      equals "ifNotZero($bw-limit-var,'policer1')";
      uid-reference;
    }
    term1 equals "ifNotZero($term1-var,$term1-var)";
    input-filter uid;
    output-filter uid;
    policer1 uid;
  }
  interfaces {
    pp0 {
      unit "$junos-interface-unit" {
        family inet {
          filter {
            input "$input-filter-ref" precedence 50;
            output "$output-filter-ref" precedence 50;
          }
        }
      }
    }
  }
  firewall {
    family inet {
      filter "$input-filter" {
        interface-specific;
        term $term1 {
          then {
            policer "$policer1-ref";
            service-accounting;
          }
        }
        term rest {
          then accept;
        }
      }
      filter "$output-filter" {
        interface-specific;
        term rest {
          then accept;
        }
      }
    }
  }
}
```

```
    }
  }
  policer "$policer1" {
    if-exceeding {
      bandwidth-limit "$bw-limit-var";
      burst-size-limit 15k;
    }
    then discard;
  }
}
```

In the **filter-service** configuration example, **input-filter-ref**, **output-filter-ref**, **policer1-ref**, and **term1** are conditional variables while **input "\$input-filter-ref" precedence 50**, **output "\$output-filter-ref" precedence 50**, **term \$term1**, and **policer "\$policer1-ref"** are conditional configuration statements. In this example, if the value of **input-filter-var** is 0, the value of the conditional variable **input-filter-ref** becomes NULL. Thus the entire configuration statement, **input "\$input-filter-ref" precedence 50**, in which the conditional variable is referenced, is ignored. If, however, the value of the variable is non-zero, the configuration statement is processed during the service profile instantiation.

CoS Service Configuration Example

```
cos-service {
  variables {
    sch1_var mandatory;
    sch2_var mandatory;
    sch1_ref {
      equals "ifNotZero($sch1_var,$sch1_var)";
      uid-reference;
    }
    Sch2_ref {
      equals "ifNotZero($sch2_var,$sch2_var)";
      uid-reference;
    }
    smap1 uid;
    sch1 uid;
    sch2 uid;
  }
  class-of-service {
    scheduler-maps {
      "$smap1" {
        forwarding-class best-effort scheduler "$sch1_ref";
        forwarding-class assured-forwarding scheduler "$sch2_ref";
      }
    }
    schedulers {
      "$sch1" {
        transmit-rate percent 30;
        buffer-size percent 30;
        priority low;
      }
      "$sch2" {
        transmit-rate percent 10;
        buffer-size percent 10;
        priority high;
      }
    }
  }
}
```

```
    }  
}
```

In the **cos-service** configuration example, **sch1_ref** and **sch2_ref** are conditional variables while **forwarding-class best-effort scheduler "\$sch1_ref"** and **forwarding-class assured-forwarding scheduler "\$sch2_ref"** are conditional configuration statements. Similar to the evaluation in the **filter-service** configuration example, if the value of any variable, referenced in a conditional variable is 0, the configuration statement in which the conditional variable is referenced is ignored and not processed during CoS service profile instantiation.

**Related
Documentation**

- *Configuring RADIUS Server Parameters for Subscriber Access*
- [Dynamic Profiles Overview on page 13](#)
- [Using Variable Expressions in User-Defined Variables on page 53](#)

CHAPTER 7

Modifying Dynamic Profiles That Are in Use by Active Subscribers

- [Enabling Dynamic Profiles to Use Multiple Versions on page 63](#)
- [Modifying Dynamic Profiles with Versioning Disabled on page 64](#)

Enabling Dynamic Profiles to Use Multiple Versions

You can create new versions of dynamic profiles that are currently in use by subscribers. Any subscriber that logs in following a dynamic profile modification uses the latest version of the dynamic profile. Subscribers that are already active continue to use the older version of the dynamic profile until they log out or their session terminates.



NOTE: You must enable or disable dynamic profile version creation before creating or using any dynamic profiles on the router. Enabling or disabling dynamic profile version creation after dynamic profiles are configured is not supported.

To configure versioning for dynamic profiles:

1. Access the router system hierarchy level.

```
[edit]  
user@host# edit system
```

2. Access the global dynamic profile options.

```
[edit system]  
user@host# edit dynamic-profile-options
```

3. Enable version creation for dynamic profiles on the router.

```
[edit system dynamic-profile-options]  
user@host# set versioning
```

Related Documentation

- For special considerations when configuring dynamic profile version creation, see [Dynamic Profiles Overview on page 13](#).
- [Configuring a Basic Dynamic Profile on page 16](#)

Modifying Dynamic Profiles with Versioning Disabled

You use dynamic profiles to configure large groups of subscribers. However, after you have configured and applied dynamic profiles, be cautious when modifying any dynamic profiles that are in use by active subscribers on the router if you have not enabled the router to use dynamic profile versioning. This section provides guidelines and procedures for modifying existing profiles and applying them to subscriber interfaces if dynamic profile versioning is not enabled on the router.

When modifying dynamic profiles, keep the following considerations in mind:

- Do not modify a dynamic profile when dynamic profile versioning is disabled and the dynamic profile is in use by active subscribers.
- Modifying a dynamic profile when dynamic profile versioning is disabled and when the dynamic profile is in use by active subscribers can lead to unpredictable behavior.

When a dynamic profile is modified and committed when dynamic profile versioning is not enabled, the router:

1. Logs a warning that the profiles are being modified and committed.
2. Determines whether the profile is currently being use by any subscriber.
3. If the profile is in use by a subscriber, the commit fails and the router logs errors to report the conflict.

Juniper Networks recommends that you only modify dynamic profiles when you have enabled dynamic profile versioning on the router. However, to properly modify a dynamic profile when dynamic profile versioning is disabled on the router:

1. Ensure that no subscribers are using the dynamic profile.
2. Create a new dynamic profile with a different name that contains the desired changes:

Original Profile

```
profile1 {
  interfaces {
    "$junos-interface-ifd-name" {
      unit "$junos-underlying-interface-unit" {
        family inet {
          filter {
            input "$junos-input-filter";
          }
        }
      }
    }
  }
}
```

Original DHCP Configuration

```
forwarding-options {
  dhcp-relay {
    traceoptions {
```

```
        flag all;
    }
    .....
    dynamic-profile profile1;
    .....
}
}
```

New Profile

```
profile2 {
  interfaces {
    "$junos-interface-ifd-name" {
      unit "$junos-underlying-interface-unit" {
        family inet {
          filter {
            input "$junos-input-filter";
            output "$junos-output-filter; /* added output filter variable */";
          }
        }
      }
    }
  }
}
```

Modified DHCP Configuration

```
forwarding-options {
  dhcp-relay {
    traceoptions {
      flag all;
    }
    .....
    dynamic-profile profile2; /* Name changed from profile1 */
    .....
  }
}
```

3. Commit the configuration containing the modified profile.

The modified profile is used for any new subscribers that access the router.

Related Documentation

- [Configuring a Basic Dynamic Profile on page 16](#)
- [Dynamic Profiles Overview on page 13](#)
- [Enabling Dynamic Profiles to Use Multiple Versions on page 63](#)

PART 3

Configuration Statements

- [Configuration Statements on page 69](#)
- [Operational Commands on page 83](#)

CHAPTER 8

Configuration Statements

- [default-value on page 69](#)
- [dynamic-profile-options on page 70](#)
- [dynamic-profiles on page 71](#)
- [mandatory on page 79](#)
- [predefined-variable-defaults \(Dynamic Profiles\) on page 80](#)
- [subscriber-management \(Subscriber Management\) on page 81](#)
- [variables on page 82](#)
- [versioning on page 82](#)

default-value

Syntax	<code>default-value <i>default-value</i>;</code>
Hierarchy Level	[edit dynamic-profiles <i>profile-name</i> variables <i>variable-name</i>]
Release Information	Statement introduced in Junos OS Release 9.3.
Description	Configure a default value for a user-defined variable in a dynamic profile. The values that the system uses for these variables are applied when the subscriber authenticates.
Options	<i>default-value</i> —Default value for the variable.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• User-Defined Variables on page 51• Using Variable Expressions in User-Defined Variables on page 53• Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52

dynamic-profile-options

Syntax	<code>dynamic-profile-options { versioning; }</code>
Hierarchy Level	[edit system]
Release Information	Statement introduced in Junos OS Release 11.4.
Description	<p>Configure global dynamic profile options.</p> <p>The remaining statement is explained separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Enabling Dynamic Profiles to Use Multiple Versions on page 63

dynamic-profiles

```

Syntax  dynamic-profiles {
        profile-name {
            class-of-service {
                interfaces {
                    interface-name ;
                }
                unit logical-unit-number {
                    classifiers {
                        type (classifier-name | default);
                    }
                    output-traffic-control-profile (profile-name | $junos-cos-traffic-control-profile);
                    rewrite-rules {
                        dscp (rewrite-name | default);
                        dscp-ipv6 (rewrite-name | default);
                        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
                        inet-precedence (rewrite-name | default);
                    }
                }
            }
        }
        scheduler-maps {
            map-name {
                forwarding-class class-name scheduler scheduler-name;
            }
        }
        schedulers {
            (scheduler-name) {
                buffer-size (seconds | percent percentage | remainder | temporal microseconds);
                drop-profile-map loss-priority (any | low | medium-low | medium-high | high)
                    protocol (any | non-tcp | tcp) drop-profile profile-name;
                excess-priority (low | high | $junos-cos-scheduler-excess-priority);
                excess-rate (percent percentage | percent $junos-cos-scheduler-excess-rate);
                overhead-accounting (shaping-mode) <bytes (byte-value)>;
                priority priority-level;
                shaping-rate (rate | predefined-variable);
                transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
            }
        }
        traffic-control-profiles profile-name {
            delay-buffer-rate (percent percentage | rate | $junos-cos-delay-buffer-rate);
            excess-rate (percent percentage | proportion value | percent $junos-cos-excess-rate);
            guaranteed-rate (percent percentage | rate | $junos-cos-guaranteed-rate);
            overhead-accounting (shaping-mode) <bytes (byte-value)>;
            scheduler-map map-name;
            shaping-rate (rate | predefined-variable);
        }
    }
    firewall {
        family family {
            fast-update-filter filter-name {
                interface-specific;
                match-order [match-order];
            }
        }
    }

```

```
term term-name {
  from {
    match-conditions;
  }
  then {
    action;
    action-modifiers;
  }
  only-at-create;
}
}
filter filter-name {
  enhanced-mode-override;
  fast-lookup-filter;
  instance-shared;
  interface-shared;
  interface-specific;
  term term-name {
    from {
      match-conditions;
    }
    then {
      action;
      action-modifiers;
    }
    only-at-create;
  }
  filter filter-name {
    interface-specific;
    term term-name {
      from {
        match-conditions;
      }
      then {
        action;
        action-modifiers;
      }
    }
  }
}
policer policer-name {
  filter-specific;
  if-exceeding {
    (bandwidth-limit bps | bandwidth-percent percentage);
    burst-size-limit bytes;
  }
  logical-bandwidth-policer;
  logical-interface-policer;
  physical-interface-policer;
  then {
    policer-action;
  }
}
}
hierarchical-policer uid {
  aggregate {
    if-exceeding {
      bandwidth-limit-limit bps;
      burst-size-limit bytes;
    }
  }
}
```

```

        then {
            policer-action;
        }
    }
    premium {
        if-exceeding {
            bandwidth-limit bps;
            burst-size-limit bytes;
        }
        then {
            policer-action;
        }
    }
}
policer uid {
    filter-specific;
    if-exceeding {
        (bandwidth-limit bps | bandwidth-percent percentage);
        burst-size-limit bytes;
    }
    logical-bandwidth-policer;
    logical-interface-policer;
    physical-interface-policer;
    then {
        policer-action;
    }
}
three-color-policer uid {
    action {
        loss-priority high then discard;
    }
    logical-interface-policer;
    single-rate {
        (color-aware | color-blind);
        committed-burst-size bytes;
        committed-information-rate bps;
        excess-burst-size bytes;
    }
    two-rate {
        (color-aware | color-blind);
        committed-burst-size bytes;
        committed-information-rate bps;
        peak-burst-size bytes;
        peak-information-rate bps;
    }
}
}
policy-options {
    prefix-list uid {
        ip-addresses;
        dynamic-db;
    }
}
interfaces interface-name {
    interface-set interface-set-name {

```

```
interface interface-name {
  unit logical unit number {
    advisory-options {
      downstream-rate rate;
      upstream-rate rate;
    }
  }
}
unit logical-unit-number {
  auto-configure {
    agent-circuit-identifier {
      dynamic-profile profile-name;
    }
    line-identity {
      include {
        accept-no-ids;
        circuit-id;
        remote-id;
      }
      dynamic-profile profile-name;
    }
  }
  encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-cisco-nlpid |
    atm-tcc-vc-mux | atm-mlppp-llc | atm-nlpid | atm-ppp-llc | atm-ppp-vc-mux |
    atm-snap | atm-tcc-snap | atm-vc-mux | ether-over-atm-llc |
    ether-vpls-over-atm-llc | ether-vpls-over-fr | ether-vpls-over-ppp | ethernet |
    frame-relay-ccc | frame-relay-ppp | frame-relay-tcc | frame-relay-ether-type |
    frame-relay-ether-type-tcc | multilink-frame-relay-end-to-end | multilink-ppp |
    ppp-over-ether | ppp-over-ether-over-atm-llc | vlan-bridge | vlan-ccc | vlan-vci-ccc
    | vlan-tcc | vlan-vpls);
  family family {
    address address;
    filter {
      adf {
        counter;
        input-precedence precedence;
        not-mandatory;
        output-precedence precedence;
        rule rule-value;
      }
      input filter-name (
        precedence precedence;
        shared-name filter-shared-name;
      )
      output filter-name {
        precedence precedence;
        shared-name filter-shared-name;
      }
    }
    rpf-check {
      fail-filter filter-name;
      mode loose;
    }
    service {
      input {
```



```

        service-set service-set-name {
            service-filter filter-name;
        }
        post-service-filter filter-name;
    }
    input-vlan-map {
        inner-tag-protocol-id tpid;
        inner-vlan-id number;
        (push | swap);
        tag-protocol-id tpid;
        vlan-id number;
    }
    output {
        service-set service-set-name {
            service-filter filter-name;
        }
    }
    output-vlan-map {
        inner-tag-protocol-id tpid;
        inner-vlan-id number;
        (pop | swap);
        tag-protocol-id tpid;
        vlan-id number;
    }
}
unnumbered-address interface-name <preferred-source-address address>;
}
filter {
    input filter-name (
        shared-name filter-shared-name;
    )
    output filter-name {
        shared-name filter-shared-name;
    }
}
ppp-options {
    chap;
    pap;
}
vlan-id number;
vlan-tags outer [tpid].vlan-id [inner [tpid].vlan-id];
}
}
interfaces {
    demux0 {...}
}
interfaces {
    pp0 {...}
}
policy-options {
    prefix-list uid {
        ip-addresses;
        dynamic-db;
    }
}
}
predefined-variable-defaults predefined-variable <variable-option> default-value;

```

```
protocols {
  igmp {
    interface interface-name {
      accounting;
      disable;
      group-policy;
      immediate-leave;
      no-accounting;
      promiscuous-mode;
      ssm-map ssm-map-name;
      static {
        group group {
          source source;
        }
      }
      version version;
    }
  }
  mld {
    interface interface-name {
      disable;
      (accounting | no-accounting);
      group-policy;
      immediate-leave;
      oif-map;
      passive;
      ssm-map ssm-map-name;
      static {
        group multicast-group-address {
          exclude;
          group-count number;
          group-increment increment;
          source ip-address {
            source-count number;
            source-increment increment;
          }
        }
      }
      version version;
    }
  }
  router-advertisement {
    interface interface-name {
      current-hop-limit number;
      default-lifetime seconds;
      (managed-configuration | no-managed-configuration);
      max-advertisement-interval seconds;
      min-advertisement-interval seconds;
      (other-stateful-configuration | no-other-stateful-configuration);
      prefix prefix;
      reachable-time milliseconds;
      retransmit-timer milliseconds;
    }
  }
}
routing-instances routing-instance-name {
```

```

interface interface-name;
routing-options {
  access {
    route prefix {
      next-hop next-hop;
      metric route-cost;
      preference route-distance;
      tag route-tag;
    }
  }
  access-internal {
    route subscriber-ip-address {
      qualified-next-hop underlying-interface {
        mac-address address;
      }
    }
  }
  multicast {
    interface interface-name {
      no-qos-adjust;
    }
  }
}
rib routing-table-name {
  access {
    route prefix {
      next-hop next-hop;
      metric route-cost;
      preference route-distance;
      tag route-tag;
    }
  }
  access-internal {
    route subscriber-ip-address {
      qualified-next-hop underlying-interface {
        mac-address address;
      }
    }
  }
}
}
routing-options {
  access {
    route prefix {
      next-hop next-hop;
      metric route-cost;
      preference route-distance;
      tag route-tag;
    }
  }
  access-internal {
    route subscriber-ip-address {
      qualified-next-hop underlying-interface {
        mac-address address;
      }
    }
  }
}

```

```
    }
    multicast {
      interface interface-name {
        no-qos-adjust;
      }
    }
  }
  variables {
    variable-name {
      default-value default-value;
      equals expression;
      mandatory;
      uid;
      uid-reference;
    }
  }
}
```

Hierarchy Level	[edit]
Release Information	Statement introduced in Junos OS Release 9.2. Support at the filter , policer , hierarchical-policer , three-color-policer , and policy options hierarchy levels introduced in Junos OS Release 11.4.
Description	Create dynamic profiles for use with DHCP or PPP client access.
Options	<i>profile-name</i> —Name of the dynamic profile; string of up to 80 alphanumeric characters. The remaining statements are explained separately.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring a Basic Dynamic Profile on page 16• <i>Configuring Dynamic VLANs Based on Agent Circuit Identifier Information</i>• Dynamic Profiles Overview on page 13

mandatory

Syntax	mandatory;
Hierarchy Level	[edit dynamic-profiles <i>profile-name</i> variables <i>variable-name</i>]
Release Information	Statement introduced in Junos OS Release 9.3.
Description	Specify that the external server (for example, RADIUS) must return a value for a user-defined variable. If the external server does not return a value for the variable, the dynamic profile fails.



NOTE: When a dynamic profile has mandatory and non-mandatory variables, configure mandatory variables first in the profile.

Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • User-Defined Variables on page 51 • Using Variable Expressions in User-Defined Variables on page 53 • Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52

predefined-variable-defaults (Dynamic Profiles)

Syntax	<code>predefined-variable-defaults predefined-variable <variable-option> default-value</code>
Hierarchy Level	[edit dynamic-profiles <i>profile-name</i>]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	Configure default values for the predefined variables that are configured in a dynamic profile. These default values are used when RADIUS does not supply a value.
Options	<p><i>predefined-variable</i>—Name of the Junos OS predefined variable to which you want to assign a default value. Do not include the \$junos- prefix.</p> <p><i>variable-option</i>—(Optional) Name of the specific variable option to which you want to assign a default value.</p> <p><i>default-value</i>—Default value that you want to assign to the predefined variable. Only certain predefined variables support multiple default values.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• For a list of predefined variables and options for which you can configure default values, see Junos OS Predefined Variables That Correspond to RADIUS Attributes and VSAs on page 42• Configuring Default Values for Predefined Variables in a Dynamic Profile on page 49

subscriber-management (Subscriber Management)

```
Syntax  subscriber-management {
        enable (Enhanced Subscriber Management);
        enforce-strict-scale-limit-license;
        gres-route-flush-delay;
    }
    overrides (Enhanced Subscriber Management) {
        shmlog {
            disable;
            file filename <size maximum-file-size>;
            filtering enable;
            log-name {
                all;
                logname {
                    <brief | detail | extensive | none | terse>;
                    <file-logging | no-file-logging>;
                }
            }
            log-type (debug | info | notice);
        }
        no-unsolicited-ra;
        shmlog {
        }
        traceoptions {
            file filename <files number> <match regular-expression > <size maximum-file-size>
              <world-readable | no-world-readable>;
            flag flag;
        }
    }
```

Hierarchy Level [edit system services]

Release Information Statement introduced in Junos OS Release 11.1.

Description Configure global services for subscriber management, such as maintaining subscribers, tracing operations, and enabling enhanced subscriber management.

The remaining statements are explained separately.

Required Privilege Level system—To view this statement in the configuration.
system-control—To add this statement to the configuration.

Related Documentation

- [Configuring the Router to Strictly Enforce the Subscriber Scaling License on page 9](#)
- *Delaying Removal of Access Routes and Access-Internal Routes After Graceful Routing Engine Switchover*
- *Configuring the Router to Maintain DHCP Subscribers During Interface Delete Events*
- *Tracing Subscriber Management Database Operations for Subscriber Access*
- *Configuring Junos OS Enhanced Subscriber Management*

variables

Syntax	<pre>variables { variable-name { default-value default-value; equals expression; mandatory; uid; uid-reference; } }</pre>
Hierarchy Level	[edit dynamic-profiles <i>profile-name</i>]
Release Information	Statement introduced in Junos OS Release 9.3.
Description	Configure user-defined variables in a dynamic profile. The values that the system uses for these variables are applied when the subscriber authenticates.
Options	<p><i>variable-name</i>—Name of the variable.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Dynamic Variables Overview on page 19• User-Defined Variables on page 51• Using Variable Expressions in User-Defined Variables on page 53• Configuring User-Defined Dynamic Variables in Dynamic Profiles on page 52

versioning

Syntax	<pre>versioning;</pre>
Hierarchy Level	[edit system dynamic-profile-options]
Release Information	Statement introduced in Junos OS Release 11.4.
Description	Enable version support for dynamic profiles on the system.
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Enabling Dynamic Profiles to Use Multiple Versions on page 63

CHAPTER 9

Operational Commands

- `show dynamic-profile session`
- `show system subscriber-management route`

show dynamic-profile session

Syntax show dynamic-profile session
 <client-id *client-id*>
 <profile-name *profile-name*>
 <service-id *service-id*>

Release Information Command introduced in Junos OS Release 13.3.

Description Display dynamic profile (client or service) information for all subscribers or for subscribers specified by client ID or service session ID. You can filter the output by also specifying a dynamic profile.



NOTE:

- The output does not display the variable stanzas defined in the dynamic profile configuration.
 - The variables in the profile configuration are replaced with subscriber specific values.
 - If the conditional variable in the dynamic profile is evaluated as NULL, the subscriber value for the variable is displayed as **NONE** in the command output.
 - The variable is also displayed as **NONE** when the variable (any variable and not necessarily conditional) in the dynamic profile has no value associated with it.
 - The format in which the configuration is displayed looks similar, but not exactly the same as the format of the **show configuration dynamic-profiles** command.
-

Options **client-id** *client-id*—Display dynamic profile information for subscribers associated with the specified client.

profile-name *profile-name*—(Optional) Display dynamic profile information for the specified subscriber or service profile.

service-id *service-id*—Display dynamic profile information for subscribers associated with the specified service session.

Required Privilege Level view

List of Sample Output [show dynamic-profile session client-id \(Client ID\) on page 85](#)
 [show dynamic-profile session client-id profile-name \(Client ID and Dynamic Profile\) on page 87](#)
 [show dynamic-profile session service-id \(Service Session\) on page 87](#)

Output Fields This command displays the dynamic client or service profile configuration for each subscriber.

Sample Output

show dynamic-profile session client-id (Client ID)

```
user@host>show dynamic-profile session client-id 20
ppoe {
  interfaces {
    pp0 {
      unit 1073741831 {
        ppp-options {
          chap;
          pap;
        }
        pppoe-options {
          underlying-interface ge-2/0/0.0;
          server;
        }
        family {
          inet {
            unnumbered-address lo0.0;
          }
        }
      }
    }
  }
  class-of-service {
    traffic-control-profiles {
      tcp1 {
        scheduler-map smap1_UID1024;
        shaping-rate 100m;
      }
    }
    interfaces {
      pp0 {
        unit 1073741831 {
          output-traffic-control-profile tcp1;
        }
      }
    }
    scheduler-maps {
      smap1_UID1024 {
        forwarding-class best-effort scheduler sch1_UID1023;
      }
    }
    schedulers {
      sch1_UID1023 {
        transmit-rate percent 40;
        buffer-size percent 40;
        priority low;
      }
    }
  }
}
filter-service {
  interfaces {
    pp0 {
      unit 1073741831 {
```

```
        family {
            inet {
                filter {
                    input input-filter_UID1026 precedence 50;
                    output output-filter_UID1027 precedence 50;
                }
            }
        }
    }
}
firewall {
    family {
        inet {
            filter input-filter_UID1026 {
                interface-specific;
                term t1 {
                    then {
                        policer policer1_UID1025;
                        service-accounting;
                    }
                }
                term rest {
                    then accept;
                }
            }
            filter output-filter_UID1027 {
                interface-specific;
                term rest {
                    then accept;
                }
            }
        }
    }
}
policer policer1_UID1025 {
    if-exceeding {
        bandwidth-limit 1m;
        burst-size-limit 15k;
    }
    then discard;
}
}
cos-service {
    class-of-service {
        scheduler-maps {
            smap2_UID1029 {
                forwarding-class assured-forwarding scheduler sch2_UID1028;
            }
        }
        schedulers {
            sch2_UID1028 {
                transmit-rate percent 60;
                buffer-size percent 60;
                priority high;
            }
        }
    }
}
}
bsimmons
}
```

show dynamic-profile session client-id profile-name (Client ID and Dynamic Profile)

```

user@host>show dynamic-profile session client-id 20 profile-name cos-service
cos-service {
  class-of-service {
    scheduler-maps {
      smap2_UID1029 {
        forwarding-class assured-forwarding scheduler sch2_UID1028;
      }
    }
    schedulers {
      sch2_UID1028 {
        transmit-rate percent 60;
        buffer-size percent 60;
        priority high;
      }
    }
  }
}

```

show dynamic-profile session service-id (Service Session)

```

user@host>show dynamic-profile session service-id 21
filter-service {
  interfaces {
    pp0 {
      unit 1073741831 {
        family {
          inet {
            filter {
              input input-filter_UID1026 precedence 50;
              output output-filter_UID1027 precedence 50;
            }
          }
        }
      }
    }
  }
}
firewall {
  family {
    inet {
      filter input-filter_UID1026 {
        interface-specific;
        term t1 {
          then {
            policer policer1_UID1025;
            service-accounting;
          }
        }
        term rest {
          then accept;
        }
      }
      filter output-filter_UID1027 {
        interface-specific;
        term rest {
          then accept;
        }
      }
    }
  }
}

```

```
    }  
    policer policer1_UID1025 {  
        if-exceeding {  
            bandwidth-limit 1m;  
            burst-size-limit 15k;  
        }  
        then discard;  
    }  
}
```

show system subscriber-management route

Syntax	<pre>show system subscriber-management route <family <i>family</i>> <incomplete> <level (brief detail)> <next-hop> <route-type <i>type</i>> <rrt-index <i>index</i>> <summary></pre>
Release Information	Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.
Description	Display information about how routes are mapped to specific enhanced subscriber management interfaces. You can customize the output by including one or more optional filters in the command. With the exception of the summary option, all filter options can be combined in a single command.
Options	<p>family <i>family</i>—(Optional) Display route mapping information for the specified protocol family: inet (IPv4) or inet6 (IPv6).</p> <p>incomplete—(Optional) Display route mapping information for incomplete routes that are missing elements required to add the routes to the routing table.</p> <p>level (brief detail)—(Optional) Display the specified level of output: brief or detail.</p> <p>next-hop—(Optional) Display the next hop associated with the route entry.</p> <p>route-type <i>type</i>—(Optional) Display route mapping information for the specified route type: access, access-internal, kernel, or local.</p> <p>rrt-index <i>index</i>—(Optional) Display mapping information for the specified routing table index, in the range 0 through 65535. An rrt-index value of 0 (zero) denotes routes in the default routing table managed by enhanced subscriber management.</p> <p>summary—(Optional) Display summary information about the routes managed by enhanced subscriber management.</p>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • <i>Verifying and Managing Junos OS Enhanced Subscriber Management</i>
List of Sample Output	<p>show system subscriber-management route family route-type rrt-index level brief on page 90</p> <p>show system subscriber-management route family route-type rrt-index level detail on page 91</p> <p>show system subscriber-management route family route-type rrt-index level brief on page 92</p>

Output Fields Table 11 on page 90 lists the output fields for the **show system subscriber-management route** command. Output fields are listed in the approximate order in which they appear.

Table 11: show system subscriber-management route Output Fields

Field Name	Field Description	Level of Output
<i>address</i>	IPv4 or IPv6 address associated with the route entry.	All levels
Route Type	One of the following route types: <ul style="list-style-type: none"> • Access • Access-internal • Kernel • Local 	All levels
Interface	Name of the enhanced subscriber management interface associated with the route entry.	All levels
Next-hop	Next-hop associated with the route entry.	All levels
Rtt-index	Value of the routing table index. A value of 0 (zero) denotes a route in the default routing table managed by enhanced subscriber management.	detail
Bbe index	Value of the interface index for the control plane.	detail
Flow id	Value of the route object index.	detail
Reference Count	Used for internal accounting.	detail
Dirty Flags	Used for internal accounting.	detail
Flags	Used for internal accounting.	detail
Family	One of the following protocol families: <ul style="list-style-type: none"> • AF_INET—IPv4 • AF_INET6—IPv6 	detail

Sample Output

show system subscriber-management route family route-type rtt-index level brief

The following example displays abbreviated information about IPv6 access routes in the default routing table (**rtt-index 0**) managed by enhanced subscriber management.

```

user@host> show system subscriber-management route family inet6 route-type access rtt-index 0
level brief
2001:db8::/64
    Route Type: Access
    Interface: pp0.3221225479, Next-hop:721
2001:db8:0:0:1::/64
    Route Type: Access

```



```

        Interface: pp0.3221225477, Next-hop:721
2001:db8:0:0:2::/64
    Route Type: Access
        Interface: pp0.3221225478, Next-hop:721
2001:db8:0:0:3::/64
    Route Type: Access
        Interface: pp0.3221225480, Next-hop:721
2001:db8:0:0:4::/64
    Route Type: Access
        Interface: pp0.3221225481, Next-hop:721
2001:db8:2002::/84
    Route Type: Access
        Interface: demux0.3221225492, Next-hop:721
2001:db8:0:0:5::/64
    Route Type: Access
        Interface: pp0.3221225487, Next-hop:721
2001:db8:0:0:6::/64
    Route Type: Access

```

show system subscriber-management route family route-type rtt-index level detail

The following example displays detailed information about IPv6 access routes in the default routing table (**rtt-index 0**) managed by enhanced subscriber management.

```

user@host> show system subscriber-management route family inet6 route-type access rtt-index 0
level detail

```

```

2001:db8::/64
    Route Type:      Access
    Interface:       pp0.3221225479
    Next-hop:        721
    Rtt-index:       0
    Bbe index:       9
    Flow id:         1
    Reference Count: 1
    Dirty Flags:     0
    Flags:           0x10082
    Family:          AF_INET6
2001:db8:0:0:1::/64
    Route Type:      Access
    Interface:       pp0.3221225477
    Next-hop:        721
    Rtt-index:       0
    Bbe index:       9
    Flow id:         1
    Reference Count: 1
    Dirty Flags:     0
    Flags:           0x10082
    Family:          AF_INET6
2001:db8:0:0:2::/64
    Route Type:      Access
    Interface:       pp0.3221225478
    Next-hop:        721
    Rtt-index:       0
    Bbe index:       9
    Flow id:         1
    Reference Count: 1
    Dirty Flags:     0
    Flags:           0x10082
    Family:          AF_INET6
2001:db8:0:0:3::/64
    Route Type:      Access

```

```
Interface:      pp0.3221225480
Next-hop:      721
Rtt-index:     0
Bbe index:     9
Flow id:       1
Reference Count: 1
Dirty Flags:   0
Flags:         0x10082
Family:        AF_INET6
```

show system subscriber-management route family route-type rtt-index level brief

The following example displays abbreviated information about IPv6 access routes in the default routing table (**rtt-index 0**) managed by enhanced subscriber management.

```
user@host> show system subscriber-management route family inet6 route-type access rtt-index 0
level brief
```

```
2000::/64
  Route Type: Access
  Interface: pp0.3221225479, Next-hop:721
2001:db8:0:0:1::/64
  Route Type: Access
  Interface: pp0.3221225477, Next-hop:721
2001:db8:0:0:2::/64
  Route Type: Access
  Interface: pp0.3221225478, Next-hop:721
2001:db8:0:0:3::/64
  Route Type: Access
  Interface: pp0.3221225480, Next-hop:721
2001:db8:0:0:4::/64
  Route Type: Access
  Interface: pp0.3221225481, Next-hop:721
2001:db8:2002::/84
  Route Type: Access
  Interface: demux0.3221225492, Next-hop:721
2001:db8:0:0:5::/64
  Route Type: Access
  Interface: pp0.3221225487, Next-hop:721
2001:db8:0:0:6::/64
  Route Type: Access
```