JUNIPER NETWORKS

Junos® OS

Authentication and Integrated User Firewalls Feature Guide for Security Devices

Modified: 2017-02-13

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

Junos® OS Authentication and Integrated User Firewalls Feature Guide for Security Devices

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

List of Figures

List of Tables

About the Documentation

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:

- SRX5800
- SRX5600
- SRX5400
- SRX1500
- SRX550M
- SRX345
- SRX340
- SRX320

- SRX300
- SRX 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.xsl;
       }
     }
   }
   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.xsl; }
   ```

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 xxiv 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 xxiv 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**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	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** *domain-name*

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

Convention	Description	Examples
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• 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 *metric*>;**
| (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** **(*string1* | *string2* | *string3*)**
# (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 [*community-ids*]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  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.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (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

Overview

CHAPTER 1

Introduction to User Authentication

Understanding User Authentication for Security Devices

Firewall user authentication lets you define firewall users and create policies that require the users to authenticate themselves through one of two authentication schemes: pass-through authentication or web authentication.

User role firewall policies can be integrated with firewall authentication both to authenticate users and to retrieve username and role information. The information is mapped to the IP address of the traffic, stored in the firewall authentication table, and used for user role firewall policy enforcement.

Infranet authentication occurs when an SRX Series device acts as an Infranet Enforcer for an IC Series device. You deploy the Infranet Enforcer in front of the servers and resources that you want to protect. Authentication occurs on the IC Series device and provides policies to the Enforcer to determine whether or not to allow an endpoint access to protected resources.

Understanding the Three-Tiered User Firewall Features

Juniper Networks offers three tiers of user firewall. The three features have different characteristics that are appropriate in different environments. Figure 1 on page 4 illustrates the relative security level of the three tiers. Table 3 on page 4 compares them to help you decide which best suits your implementation.

Figure 1: Three-Tiered User Firewall Features

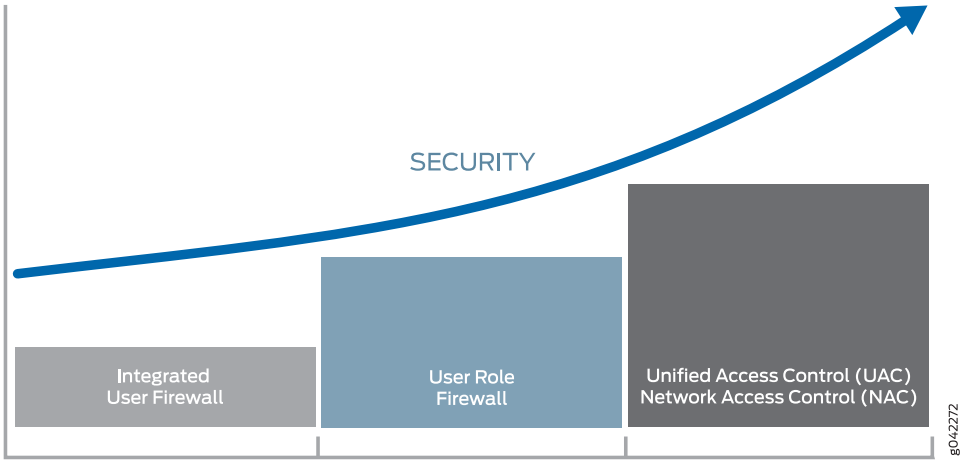


Table 3 on page 4 describes the basic differences among the three features.

Table 3: Comparison of User Firewall Features

	Integrated User Firewall	User Role Firewall	Unified Access Control (UAC) Network Access Control (NAC)
Authentication	Passive authentication–Does not interact with client directly; polls the Active Directory for login information.	Active authentication–Queries the client.	End-to-end–Authenticates the user down to the access level where user connects, whether wired or wireless.
Extent of Authentication	Best effort.	Deterministic–User is identified.	Deterministic–User is identified.
Where Enforced	Enforced at firewall.	Enforced at firewall.	Enforced at access (switch or WiFi) and firewall.
Devices Needed	SRX Series	SRX Series and MAG Series	SRX Series and MAG Series
Ideal Environments	• Needs visibility into who is accessing the SRX Series • Small-to-medium business • Low-scale deployment	• Security-conscious environments • Scales up to 50,000 users	• Large-scale deployment • Interface for Metadata Access Points (IF-MAP) federation

- You can upgrade to a higher tier if you choose. From integrated user firewall, simply add the MAG Series to get user role firewall. From there, add licenses to get full UAC NAC.
- The three offerings provide maximum flexibility; they are supported on all SRX Series hardware platforms.

Related Documentation

- Overview of Integrated User Firewall on page 141
- *Understanding User Role Firewalls*

PART 2

Configuring Firewall User Authentication

CHAPTER 2

Understanding Firewall Authentication

Firewall User Authentication Overview

A firewall user is a network user who must provide a username and password for authentication when initiating a connection across the firewall. Junos OS enables administrators to restrict and permit firewall users to access protected resources (different zones) behind a firewall based on their source IP address and other credentials.

Junos OS also supports the administrator and Point-to-Point Protocol (PPP) user types.

NOTE: Starting in Junos OS Release 12.1X44-D10, HTTPS-based authentication is introduced on high-end SRX Series Services Gateways. Starting in Junos OS Release 15.1X49-D40, HTTPS-based authentication is introduced on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways.

After you define firewall users, you can create a policy that requires the users to authenticate themselves through one of three authentication schemes:

- **Pass-through authentication**—A host or a user from one zone tries to access resources on another zone. You must use an FTP client, a Telnet client, an HTTP client, or an HTTPS client to access the IP address of the protected resource and to get authenticated by the firewall. The device uses FTP, Telnet, HTTP, or HTTPS to collect username and password information, and subsequent traffic from the user or host is allowed or denied based on the result of this authentication. When the device is using an HTTPS server, and after the authentication is done, the subsequent traffic from the user is always terminated whether the authentication is successful or not.
- **Pass-through with web-redirect authentication**—This authentication scheme is for HTTP and HTTPS. when firewall authentication is performed using the pass-through authentication scheme for HTTP and HTTPs clients, the web-redirect feature optionally, redirects HTTP or HTTPS requests to the device's internal webserver by sending a redirect HTTP or HTTPS response to the client system to reconnect to the webserver

for user authentication. The interface on which the client's request arrived is the interface to which the request is redirected.

Using this feature allows for a richer user login experience. For example. instead of a popup prompt asking for username and password, users can get the login page in a browser. Enabling **web-redirect** has the same effect as users typing the Web authentication IP address in a client browser. Using **web-redirect** provides a more seamless authentication experience because users do not need to know the Web authentication IP address but only the IP address of the resource they are trying to access. After the user has been authenticated this way, traffic from user's IP address is authenticated to go through the **web-redirect** method.

When you web-redirect, you http or https the destination url, the authentication page prompts you after successful authentication.

A message is displayed to inform you about the successful authentication. After successful authentication, the browser launches your original destination URL without your needing to retype the URL.

The following message is displayed:

```
Redirecting to the original url, please wait
```

- Web authentication—Users try to connect, using HTTP or HTTPS, to an IP address on the device that is enabled for Web authentication; in this scenario, you do not use HTTP or HTTPS to get to the IP address of the protected resource. You are prompted for the username and password that are verified by the device. Subsequent traffic from the user or host to the protected resource is allowed or denied based on the result of this authentication.

Release History Table

Release	Description
15.1X49-D40	Starting in Junos OS Release 15.1X49-D40, HTTPS-based authentication is introduced on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways.
12.1X44-D10	Starting in Junos OS Release 12.1X44-D10, HTTPS-based authentication is introduced on high-end SRX Series Services Gateways.

Related Documentation

- Understanding Pass-Through Authentication on page 35
- Understanding Web Authentication on page 51
- Understanding External Authentication Servers on page 65
- Configuring SSL Proxy on page 23

Obtaining Username and Role Information Through Firewall Authentication

User role firewall policies can be integrated with firewall authentication both to authenticate users and to retrieve username and role information. The information is mapped to the IP address of the traffic, stored in the firewall authentication table, and used for user role firewall policy enforcement.

The following CLI statements configure firewall authentication for user role firewall enforcement.

1. If not already established, define the access profile to be used for firewall authentication. You can skip this step if an existing access profile provides the client data needed for your implementation.

   The access profile is configured in the **[edit access profile]** hierarchy as with other firewall authentication types. It defines clients as firewall users and the passwords that provide them access. Use the following command to define a profile and add client names and passwords for firewall authentication.

   set access profile *profile-name* client *client-name* firewall-user password *pwd*

2. If HTTPS traffic is expected, define the access profile to be used for SSL termination services. You can skip this step if an existing SSL termination profile provides the services needed for your implementation.

   The SSL termination profile is configured in the **[edit services ssl]** hierarchy.

   set services ssl termination profile *ssl-profile-name* server-certificate *certificate-type*

3. Enable the firewall authentication table as an authentication source.

   set security user-identification authentication-source firewall-authentication priority *priority*

   The priority value determines the sequence in which authentication sources are checked. The default value is 150 for the firewall authentication table. (It is 100 for the local authentication table and 200 for the Unified Access Control (UAC) authentication table.) By default, the local authentication table is checked first, the firewall authentication table is next, and the UAC authentication table is third if it is enabled. You can change this sequence by changing the priority value of one or more of the tables.

4. Configure policies that permit traffic for user firewall authentication.

   edit security policies from-zone *zone* to-zone *zone* policy *policy-name*
   set match source-identity unauthenticated-user
   set then permit firewall-authentication user-firewall access-profile *profile-name*
   ssl-termination-profile *profile-name*

   When unauthenticated traffic is permitted for firewall authentication, the user is authenticated based on the access profile configured in this statement. The **ssl-termination-profile** option is needed only for HTTPS traffic.

   By specifying the authentication type **user-firewall**, the firewall authentication table is propagated with the IP address, the username, and any group names associated with the authenticated user. (Group names from firewall authentication are interpreted

as roles by the user role firewall.) Any further traffic from this IP address will match the IP address in the firewall authentication table, and not require authentication. The associated username and roles are retrieved from the table for use as potential match criteria in subsequent security policies.

Related Documentation

- *Understanding the User Identification Table*

CHAPTER 3

Configuring Encrypted Files Using SSL Proxy

SSL Proxy Overview

Secure Sockets Layer (SSL) is an application-level protocol that provides encryption technology for the Internet. SSL, also called Transport Layer Security (TLS), ensures the secure transmission of data between a client and a server through a combination of privacy, authentication, confidentiality, and data integrity. SSL relies on certificates and private-public key exchange pairs for this level of security. SSL is supported on the SRX550M, SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and the SRX5800 devices.

Server authentication guards against fraudulent transmissions by enabling a Web browser to validate the identity of a webserver. Confidentiality mechanisms ensure that communications are private. SSL enforces confidentiality by encrypting data to prevent unauthorized users from eavesdropping on electronic communications. Finally, message integrity ensures that the contents of a communication have not been tampered with.

SSL proxy is transparent; that is, it performs SSL encryption and decryption between the client and the server, but neither the server nor the client can detect its presence. Existing features like SSL offload and SSL inspection require the servers to share their secret keys to be able to decrypt the SSL traffic. However, sharing server keys is sometimes not feasible or might not be available in certain circumstances, in which case the SSL traffic cannot be decrypted.

SSL proxy addresses this problem by ensuring that it has the keys to encrypt and decrypt the payload:

- For the server, SSL proxy acts as a client—Because SSL proxy generates the shared pre-master key, it determines the keys to encrypt and decrypt.
- For the client, SSL proxy acts as a server—SSL proxy first authenticates the original server and replaces the public key in the original server certificate with a key that is known to it. It then generates a new certificate by replacing the original issuer of the

certificate with its own identity and signs this new certificate with its own public key (provided as a part of the proxy profile configuration). When the client accepts such a certificate, it sends a shared pre-master key encrypted with the public key on the certificate. Because SSL proxy replaced the original key with its own key, it is able to receive the shared pre-master key. Decryption and encryption take place in each direction (client and server), and the keys are different for both encryption and decryption.

Figure 2 on page 14 depicts how SSL inspection (on an existing SRX Series IDP module) is typically used to protect servers. SSL inspection requires access to the private keys used by the servers so that the SRX Series device can decrypt the encrypted traffic.

Figure 2: SSL Inspection on an Existing SRX Series IDP Module

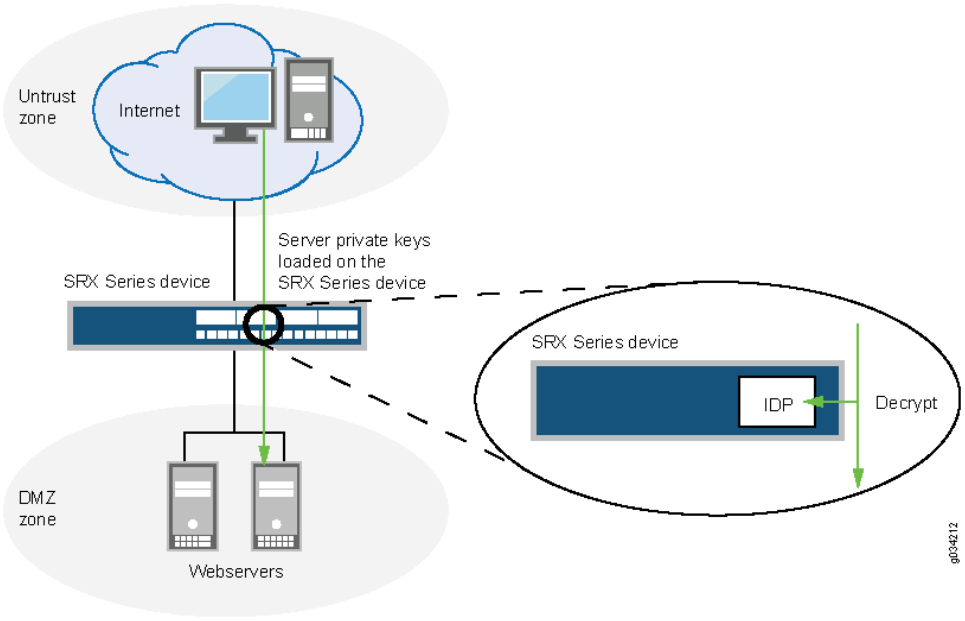


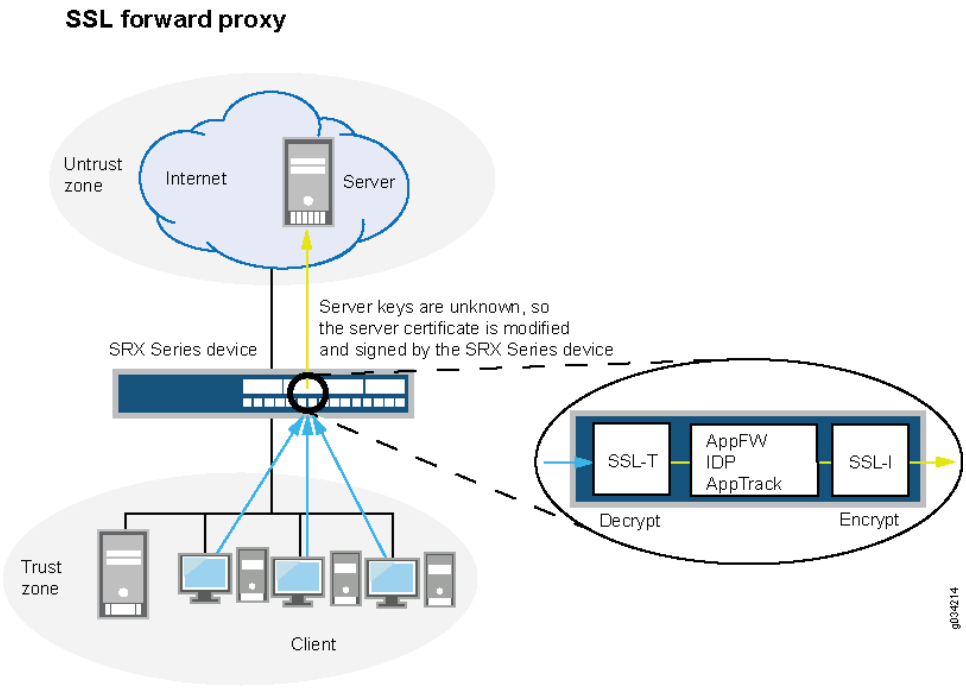
Figure 3 on page 15 shows how SSL proxy works on an encrypted payload. When application firewall (AppFW), Intrusion Detection and Prevention (IDP), or application tracking (AppTrack) is configured, the SSL proxy acts as an SSL server by terminating the SSL session from the client and establishing a new SSL session to the server, the SRX Series device decrypts and then reencrypts all SSL proxy traffic. SSL proxy uses the following:

- SSL-T-SSL terminator on the client side.
- SSL-I-SSL initiator on the server side.
- Configured AppFW, IDP, or AppTrack services use the decrypted SSL sessions.

NOTE: If none of the services (AppFW, IDP, or AppTrack) are configured, then SSL proxy services are bypassed even if an SSL proxy profile is attached to a firewall policy.

NOTE: The IDP module will not perform its SSL inspection on a session if SSL proxy is enabled for that session. That is, if both SSL inspection and SSL proxy are enabled on a session, SSL proxy will always take precedence.

Figure 3: SSL Proxy on an Encrypted Payload



Supported Ciphers in Proxy Mode

An SSL cipher comprises encryption ciphers, authentication method, and compression. Table 4 on page 16 displays a list of supported ciphers. NULL ciphers are excluded.

The following SSL protocols are supported on SRX Series devices:

- TLS version 1.0—Provides secure communication over networks by providing privacy and data integrity between communicating applications
- TLS version 1.1—This enhanced version of TLS provides protection against cipher-block chaining (CBC) attacks.
- TLS version 1.2 — This enhanced version of TLS provides improved flexibility for negotiation of cryptographic algorithms.

Starting with Junos OS Release 15.1X49-D30 , TLS version 1.1 and TLS version 1.2 protocols are supported on SRX Series devices along with TLS version 1.0.

Starting with Junos OS Release 15.1X49-D20, the SSL protocol 3.0 (SSLv3) support is deprecated.

Table 4: Supported SSL Cipher List

SSL Cipher	Key Exchange Algorithm	Data Encryption	Message Integrity
RSA_WITH_RC4_128_MD5	RSA key exchange	128-bit RC4	Message Digest 5 (MD5) hash
RSA_WITH_RC4_128_SHA	RSA key exchange	128-bit RC4	Secure Hash Algorithm (SHA) hash
RSA_WITH_DES_CBC_SHA	RSA key exchange	DES CBC	SHA hash
RSA_WITH_3DES_EDE_CBC_SHA	RSA key exchange	3DES EDE/CBC	SHA hash
RSA_WITH_AES_128_CBC_SHA	RSA key exchange	128-bit AES/CBC	SHA hash
RSA_WITH_AES_256_CBC_SHA	RSA key exchange	256-bit AES/CBC	SHA hash
RSA_EXPORT_WITH_RC4_40_MD5	RSA-export	40-bit RC4	MD5 hash
RSA_EXPORT_WITH_DES40_CBC_SHA	RSA-export	40-bit DES/CBC	SHA hash
RSA_EXPORT1024_WITH_DES_CBC_SHA	RSA 1024 bit export	DES/CBC	SHA hash
RSA_EXPORT1024_WITH_RC4_56_MD5	RSA 1024 bit export	56-bit RC4	MD5 hash
RSA_EXPORT1024_WITH_RC4_56_SHA	RSA 1024 bit export	56-bit RC4	SHA hash

NOTE: Cipher suites that have “export” in the title are intended for use outside of the United States and might have encryption algorithms with limited key sizes.

Export ciphers are not enabled by default. You need to either configure the export ciphers to enable or install a domestic package.

NOTE: Supported SSL ciphers for HTTPS firewall authentication are RSA_WITH_3DES_EDE_CBC_SHA , RSA_WITH_AES_128_CBC_SHA, and RSA_WITH_AES_256_CBC_SHA.

Server Authentication

Implicit trust between the client and the device (because the client accepts the certificate generated by the device) is an important aspect of SSL proxy. It is extremely important

that server authentication is not compromised; however, in reality, self-signed certificates and certificates with anomalies are in abundance. Anomalies can include expired certificates, instances of common name not matching a domain name, and so forth. Server authentication is governed by setting the **ignore-server-auth-failure** option in the SSL proxy.

- By default, the **ignore-server-auth-failure** option is not defined as an action in the SSL proxy profile, and the following occurs:
  - If authentication succeeds, a new certificate is generated by replacing the keys and changing the issuer name to the issuer name that is configured in the root CA certificate in the proxy profile.
  - If authentication fails, the connection is dropped.
- If the **ignore-server-auth-failure** option is defined as an action in the SSL proxy profile, the following occurs:
  - If the certificate is self-signed, a new certificate is generated by replacing the keys only. The issuer name is not changed. This ensures that the client browser displays a warning that the certificate is not valid.
  - If the certificate has expired or if the common name does not match the domain name, a new certificate is generated by replacing the keys and changing the issuer name to SSL-PROXY: DUMMY_CERT:GENERATED DUE TO SRVR AUTH FAILURE. This ensures that the client browser displays a warning that the certificate is not valid.

Trusted CA List

SSL proxy ensures secure transmission of data between a client and a server. Before establishing a secure connection, SSL proxy checks CA certificates to verify signatures on server certificates. For this reason, a reasonable list of trusted CA certificates is required to effectively authenticate servers.

Junos OS provides the following options for trusted CA certificates:

- Loading the default trusted CA list—Junos OS provides a default list of certificates that contains well-known trusted CA certificates similar to the default certificates used by most common browsers. Without these default certificates, browsers would not be able to validate the identity of most websites and would mark them as untrusted sites.

  The Junos OS package contains the default CA certificates as a PEM file (for example, trusted_CA.pem). After you download the package and reboot your device, you can easily load the default certificates on your system using a CLI command.

  We recommend you load the default trusted CA list if you want to trust the same CA certificates as common browsers and avoid importing CA certificates manually.
- Importing the trusted CA list manually—You can import your own trusted CA certificates using the Public Key Infrastructure (PKI). The PKI helps verify and authenticate the validity of the trusted CA certificates. You create CA profile groups that include trusted CA certificates, then import the group on your device for server authentication.

- Ignoring server authentication—You can use the **ignore-server-auth-failure** option to ignore server authentication completely. In this case, SSL proxy ignores errors encountered during the server certificate verification process (such as CA signature verification failure, self-signed certificates, and certificate expiry).

  We do not recommend this option for authentication, because configuring it results in websites not being authenticated at all. However, you can use this option to effectively identify the root cause for dropped SSL sessions. See "Enabling Debugging and Tracing for SSL Proxy" on page 33.

Root CA

In a public key infrastructure (PKI) hierarchy, the root CA is at the top of the trust path. The root CA identifies the server certificate as a trusted certificate.

Client Authentication

Currently, client authentication is not supported in SSL proxy. If a server requests client authentication, a warning is issued that a certificate is not available. The warning lets the server determine whether to continue or to exit.

Whitelists

Because SSL encryption and decryption are complicated and expensive procedures, network administrators can selectively bypass SSL proxy processing for some sessions. Such sessions mostly include connections and transactions with trusted servers or domains with which network administrators are very familiar. There are also legal requirements to exempt financial and banking sites. Such exemptions are achieved by configuring the IP addresses or domain names of the servers under whitelists.

Dynamic Resolution of Domain Names

The IP addresses associated with domain names are dynamic and can change at any time. Whenever a domain IP address changes, it is propagated to the SSL proxy configuration (similar to what is done in the firewall policy configuration).

Session Resumption

An SSL session refers to the set of parameters and encryption keys created by performing a full handshake. A connection is the conversation or active data transfer that occurs within the session. The computational overhead of a complete SSL handshake and generation of master keys is considerable. In short-lived sessions, the time taken for the SSL handshake can be more than the time for data transfer.

To improve throughput and still maintain an appropriate level of security, SSL session resumption provides a session caching mechanism so that session information, such as the pre-master secret key and agreed-upon ciphers, can be cached for both the client and server. The cached information is identified by a session ID. In subsequent connections both parties agree to use the session ID to retrieve the information rather than create a new pre-master secret key. Session resumption shortens the handshake process and accelerates SSL transactions.

Session Renegotiation

After a session is created and SSL tunnel transport has been established, a change in SSL parameters requires renegotiation. SSL proxy supports both secure (RFC 5746) and nonsecure (TLS v1.0, TLS v1.1, and TLS v1.2) renegotiation. When session resumption is enabled, session renegotiation is useful in the following situations:

- Cipher keys need to be refreshed after a prolonged SSL session.
- Stronger ciphers need to be applied for a more secure connection.

A change in an SSL proxy profile that modifies a certificate, cipher strength, or trusted CA list flushes cache entries when the modified policy is committed. When a session is resumed, the SSL parameters associated with its session ID are retrieved from the cache. If the SSL proxy profile is not altered, cache entries corresponding to that profile are not flushed and the session continues. If the cache has been flushed, however, a full handshake must be performed to establish the new SSL parameters. (There is no impact to non-SSL sessions.)

NOTE: The National Cybersecurity FFRDC, operated by the MITRE Corporation, maintains a system that provides a reference module for publicly known information-security vulnerabilities and exposures in publicly released software packages. The MITRE Corporation's documentation defines unique CVE IDs to identify specific vulnerabilities.

Among these known vulnerabilities is CVE-2015-3644, which addresses possible exploitation of authentication bypass through use of the redirect option. Among the affected areas, this vulnerability can be exploited in stunnels, versions 5.00 to 5.13.

The vulnerability addressed by CVE-2015-3644 does not apply to the Junos OS because the Junos OS default configuration does not use the redirect option with stunnel. Junos OS does not expose a CLI command for enabling the redirect. option. Moreover, you should not manually modify the stunnel configuration to use the redirect option.

SSL Proxy Logs

When logging is enabled in an SSL proxy profile, SSL proxy can generate the messages shown in Table 5 on page 19.

Table 5: SSL Proxy Logs

Syslog Type	Description
SSL_PROXY_SSL_SESSION_DROP	Logs generated when a session is dropped by SSL proxy.
SSL_PROXY_SSL_SESSION_ALLOW	Logs generated when a session is processed by SSL proxy even after encountering some minor errors.
SSL_PROXY_SESSION_IGNORE	Logs generated if non-SSL sessions are initially mistaken as SSL sessions.

Table 5: SSL Proxy Logs *(continued)*

Syslog Type	Description
SSL_PROXY_SESSION_WHITELIST	Logs generated when a session is whitelisted.
SSL_PROXY_ERROR	Logs used for reporting errors.
SSL_PROXY_WARNING	Logs used for reporting warnings.
SSL_PROXY_INFO	Logs used for reporting general information.

All logs contain similar information as shown in the following example (actual order of appearance):

```
logical-system-name, session-id, source-ip-address, source-port,
destination-ip-address,destination-port,
nat-source-ip-address, nat-source-port, nat-destination-ip-address,
nat-destination-port, proxy profile name, source-zone-name,
source-interface-name, destination-zone-name,destination-interface-name, message
```

The **message** field contains the reason for the log generation. One of three prefixes shown in Table 6 on page 20 identifies the source of the message. Other fields are descriptively labeled.

Table 6: SSL Proxy Log Prefixes

Prefix	Description
system	Logs generated due to errors related to the device or an action taken as part of the SSL proxy profile. Most logs fall into this category.
openssl error	Logs generated during the handshaking process if an error is detected by the openssl library.
certificate error	Logs generated during the handshaking process if an error is detected in the certificate (x509 related errors).

Sample logs:

```
Jun  1 05:11:13 4.0.0.254 junos-ssl-proxy: SSL_PROXY_SSL_SESSION_DROP: lsys:root
  23 < 203.0.113.1/35090->192.0.2.1/443>  NAT:< 203.0.113.1/35090->192.0.2.1/443>
 ssl-inspect-profile  <untrust:ge-0/0/0.0->trust:ge-0/0/1.0> message:certificate
 error: self signed certificate
```

NOTE: These logs capture sessions that are dropped by SSL proxy, not sessions that are marked by other modules that also use SSL proxy services.

For SSL_PROXY_SESSION_WHITELIST messages, an additional **host** field is included after the **session-id** and contains the IP address of the server or domain that has been whitelisted.

```
Jun  1 05:25:36 4.0.0.254 junos-ssl-proxy: SSL_PROXY_SESSION_WHITELIST: lsys:root
 24 host:192.0.2.1/443<203.0.113.1/35090->192.0.2.1/443>  NAT:<
```

```
203.0.113.1/35090->192.0.2.1/443 > ssl-inspect-profile
<untrust:ge-0/0/0.0->trust:ge-0/0/1.0> message:system: session whitelisted
```

Leveraging Dynamic Application Identification

SSL proxy uses application identification services to dynamically detect if a particular session is SSL encrypted. SSL proxies are allowed only if a session is SSL encrypted. The following rules apply for a session:

- Session is marked **Encrypted=Yes** in the application system cache. If the session is marked **Encrypted=Yes**, it indicates that the final match from application identification for that session is SSL encrypted, and SSL proxy transitions to a state where proxy functionality can be initiated.
- Session is marked **Encrypted=No** in the application system cache. If a non-SSL entry is found in the application system cache, it indicates that the final match from application identification for that session is non-SSL and SSL proxy ignores the session.
- An entry is not found in the application system cache. This can happen on the first session, or when the application system cache has been cleaned or has expired. In such a scenario, SSL proxy cannot wait for the final match (requires traffic in both directions). In SSL proxy, traffic in reverse direction happens only if SSL proxy has initiated an SSL handshake. Initially, for such a scenario SSL proxy tries to leverage prematch or aggressive match results from application identification , and if the results indicate SSL, SSL proxy will go ahead with the handshake.
- Application identification fails due to resource constraints and other errors. Whenever the result from application identification is not available, SSL proxy will assume static port binding and will try to initiate SSL handshake on the session. This will succeed for actual SSL sessions, but it will result in dropped sessions for non SSL sessions.

Logical Systems Support

It is possible to enable SSL proxy on firewall policies that are configured using logical systems; however, note the following limitations:

- The “services” category is currently not supported in logical systems configuration. Because SSL proxy is under “services,” you cannot configure SSL proxy profiles on a per-logical-system basis.
- Because proxy profiles configured at a global level (within “services ssl proxy”) are visible across logical system configurations, it is possible to configure proxy profiles at a global level and then attach them to the firewall policies of one or more logical systems.

Limitations

NOTE:

- Starting from Junos OS Release 15.1X49-D30, certificate revocation list (CRL) checks are supported.
- Starting from Junos OS Release 15.1X49-D30, server certificates that have key size greater than 4096 are supported. Prior to Junos OS Release 115.1X49-D30, server certificates with key size greater than 2048 bits were not supported because of cryptography hardware limitations.

NOTE: On SRX Series devices, for a particular session, the SSL proxy is only enabled if a relevant feature related to SSL traffic is also enabled. Features that are related to SSL traffic are IDP, application identification, application firewall, and application tracking. If none of the above listed features are active on a session, the SSL proxy bypasses the session and logs are not generated in this scenario.

NOTE: On all SRX Series devices, the current SSL proxy implementation has the following connectivity limitations:

- The SSLv3.0 protocol support is depricated.
- The SSLv2 protocol is not supported. SSL sessions using SSLv2 are dropped.
- Only X.509v3 certificate is supported.
- Client authentication of SSL handshake is not supported.
- SSL sessions where client certificate authentication is mandatory are dropped.
- SSL sessions where renegotiation is requested are dropped.

Release History Table

Release	Description
15.1X49-D30	Starting with Junos OS Release 15.1X49-D30 , TLS version 1.1 and TLS version 1.2 protocols are supported on SRX Series devices along with TLS version 1.0.
15.1X49-D30	Starting from Junos OS Release 15.1X49-D30, certificate revocation list (CRL) checks are supported.
15.1X49-D30	Starting from Junos OS Release 15.1X49-D30, server certificates that have key size greater than 4096 are supported.
15.1X49-D20	Starting with Junos OS Release 15.1X49-D20, the SSL protocol 3.0 (SSLv3) support is deprecated.

Related Documentation

- *Understanding Address Books*
- *Understanding Global Address Books*
- *Understanding Self-Signed Certificates*
- Configuring SSL Proxy on page 23

Configuring SSL Proxy

SSL proxy works transparently between the client and the server. All requests from a client first go to the proxy server; the proxy server evaluates the request, and if the request is valid, forwards the request to the outbound side. Similarly, inbound requests are also evaluated by the proxy server. Both client and server interpret that they are communicating with each other; however, it is the SSL proxy that functions between the two. SSL proxy is supported on the SRX550M, SRX4200, SRX5400, SRX5600, and SRX5800 devices.

SSL proxies provide encryption and decryption by residing between the server and the client. Because SSL proxies are hidden from both the server and the client, secret keys are shared between the two to decrypt the SSL traffic. Proxies are known as *forward proxies* because proxy servers are used to hide any detailed information from the servers.

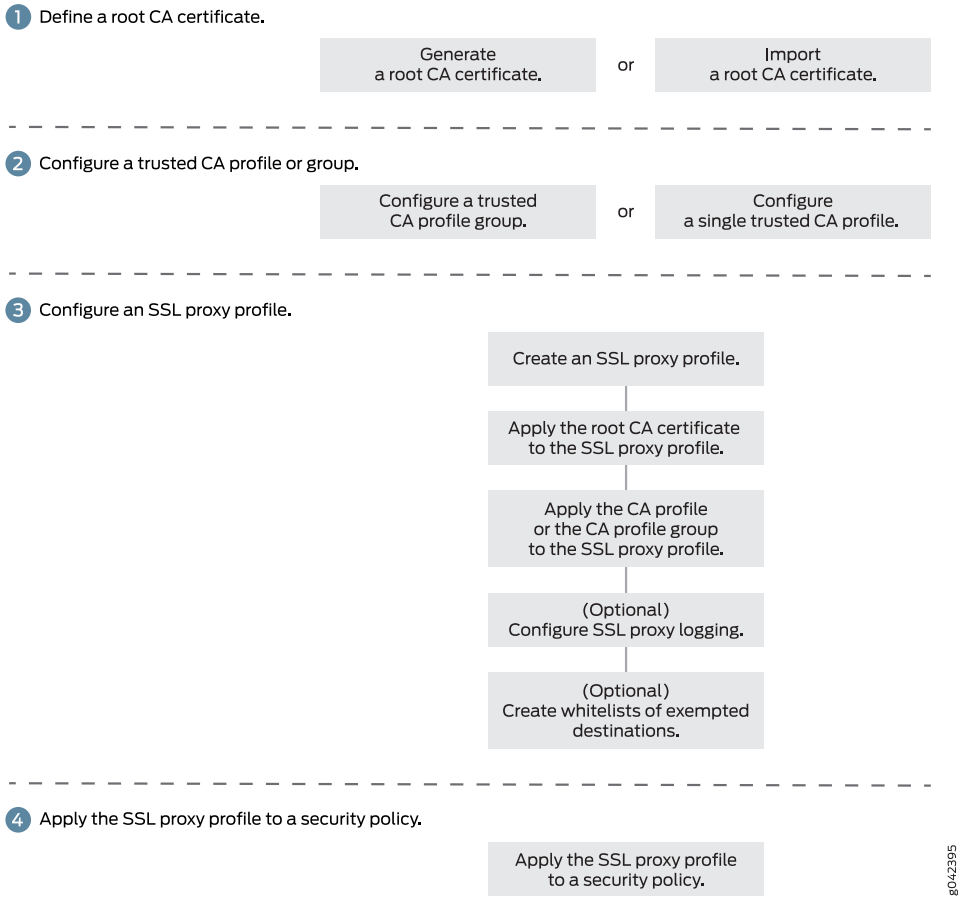
Integrity, confidentiality, and authenticity of traffic are validated through PKI, which includes digital certificates issued by the CA, certificate validity and expiration dates, details about the certificate owner and issuer, and security policies.

- SSL Proxy Configuration Overview on page 24
- Configuring a Root CA Certificate on page 24
- Configuring a CA Profile Group on page 26
- Configuring a Trusted CA Profile on page 27
- Importing a Root CA Certificate into a Browser on page 28
- Applying an SSL Proxy Profile to a Security Policy on page 29
- Creating a Whitelist of Exempted Destinations on page 30

SSL Proxy Configuration Overview

Figure 4 on page 24 displays an overview of how SSL proxy is configured. It includes some required steps, such as configuring the root CA certificate, loading a CA profile group, and applying an SSL proxy profile to a security policy, and some optional steps, such as creating whitelists and SSL proxy logging.

Figure 4: SSL Proxy Configuration Overview



Configuring a Root CA Certificate

A CA can issue multiple certificates in the form of a tree structure. A root certificate is the topmost certificate of the tree, the private key of which is used to *sign* other certificates. All certificates immediately below the root certificate inherit the signature or trustworthiness of the root certificate. This is somewhat like the *notarizing* of an identity.

You can configure a root CA certificate by first obtaining a root CA certificate (by either generating a self-signed one or importing one) and then applying it to an SSL proxy profile. There are two ways you can obtain a root CA certificate—by using the Junos OS CLI on an SRX Series device or by using OpenSSL on a UNIX device.

To generate a root CA certificate using the Junos OS CLI, follow these steps on an SRX Series device:

1. From operational mode, generate a PKI public/private key pair for a local digital certificate.

   user@host>**request security pki generate-key-pair certificate-id** ***certificate-id*** **size** ***size*** **type** ***type***

2. From operational mode, define a self-signed certificate. Specify certificate details such as the certificate identifier (generated in the previous step), a fully qualified domain name (FQDN) for the certificate, and an e-mail address of the entity owning the certificate. You can also specify other information such as the common name and the organization involved. By configuring the **add-ca-constraint** option, you make sure that the certificate can be used for signing other certificates.

   user@host>**request security pki local-certificate generate-self-signed certificate-id** ***certificate-id*** **domain-name** ***domain-name*** **subject** ***subject*** **email** ***email-id*** **add-ca-constraint**

3. From configuration mode, apply the loaded certificate as root-ca in the SSL proxy profile.

   [edit]
   user@host# **set services ssl proxy profile** ***profile-name*** **root-ca** ***certificate-id***

4. Import the root CA as a trusted CA into client browsers. This is required for the client browsers to trust the certificates signed by the SRX Series device. See "Importing a Root CA Certificate into a Browser" on page 28.

To generate a root CA certificate using OpenSSL, follow these steps on a UNIX device:

1. Create folders **keys** and **certs**.

   mkdir /etc/pki/tls/keys
   mkdir /etc/pki/tls/certs

2. Change to the **openssl** directory.

   cd /etc/pki/tls

3. Create a CA certificate key. The following command creates an RSA key using the 3DES encryption named **ca.key** that is 2048 in length. You also need to enter a password that is used to encrypt the private key. This is critical to security if the key is lost because it will still be encrypted.

   % openssl genrsa -des3 -out keys/ssl-proxy-ca.key 2048

4. Create a CA certificate based on the CA private key (created in the previous step). The expiration date for this certificate is 3 years or 1095 days. However, you can set it to a different value. When creating the certificate, you need to enter the password

and the certificate information that includes distinguished name (DN), country name, and so forth.

```
% openssl req -new -x509 -days 1095 –key keys/ssl-proxy-ca.key -out
  certs/ssl-inspect-ca.cer
```

5. Import the CA private and public keys into the SRX Series device. Copy the **ca.key** and **ca.cer** keys to the **/var/tmp** directory on the SRX Series device. You can copy using SCP, or open the files and copy them into "vi" on the SRX Series device to create new files.

   ```
   user@host> request security pki local-certificate load certificate-id ssl-inspect-ca key
     /var/tmp/ssl-inspect-ca.key filename /var/tmp/ssl-inspect-ca.cer passphrase
     password
   ```

6. From configuration mode, apply the loaded certificate as root-ca in the SSL proxy profile.

   ```
   [edit]
   user@host# set services ssl proxy profile ssl-inspect-profile root-ca ssl-inspect-ca
   ```

7. Import the root CA as a trusted CA into client browsers. This is required for the client browsers to trust the certificates signed by the SRX Series device. See "Importing a Root CA Certificate into a Browser" on page 28.

Configuring a CA Profile Group

The CA profile defines the certificate information to be used for authentication. It includes the public key that SSL proxy uses when generating a new certificate. Junos OS allows you to create a group of CA profiles and load multiple certificates in one action, view information about all certificates in a group, and delete unwanted CA groups.

You can load a group of CA profiles by obtaining a list of trusted CA certificates, defining a CA group, and attaching the CA group to the SSL proxy profile.

1. Obtain a list of trusted CA certificates by following one of these methods:

   - Junos OS provides a default list of trusted CA certificates that you can load on your system using the **default** command option. The Junos OS package contains the default CA certificates as a PEM file (for example, **trusted_CA.pem**). After you download the Junos OS package and reboot your device, the default certificates are available on your system.

     From operational mode, load the default trusted CA certificates (the group name identifies the CA profile group):

     ```
     user@host> request security pki ca-certificate ca-profile-group load ca-group-name
       group-name filename default
     ```

   - Alternatively, you can define your own list of trusted CA certificates and import them on your system. You get the list of trusted CAs in a single PEM file (for example **IE-all.pem**) and save the PEM file in a specific location (for example, **/var/tmp**). See Knowledge Base Article KB23144.

     From operational mode, load the trusted list to the device (the group name identifies the CA profile group):


```
user@host> request security pki ca-certificate ca-profile-group load ca-group-name
  group-name filename /var/tmp/IE-all.pem
```

2. From configuration mode, attach the CA profile group to the SSL proxy profile. You can attach one or all CA profile groups at a time:

   - To attach one CA profile group (the group name identifies the CA profile group):

     ```
     [edit]
     user@host# set services ssl proxy profile profile-name trusted-ca group-name
     ```

   - To attach all CA profile groups:

     ```
     [edit]
     user@host# set services ssl proxy profile profile-name trusted-ca all
     ```

You can easily display information about all certificates in a CA profile group:

```
user@host> show security pki ca-certificates ca-profile-group group-name
```

You can delete a CA profile group. Remember that deleting a CA profile group deletes all certificates that belong to that group:

```
user@host> clear security pki ca-certificates ca-profile-group group-name
```

Configuring a Trusted CA Profile

Typically, you import a list of trusted CA certificates by creating a group of CA profiles. However, you can also configure a single CA profile (containing one or multiple certificates) and import it using PKI commands. This section shows you how to import a trusted CA certificate from your browser's certificate store into your SRX Series device. The certificate that is configured under the trusted CA is loaded using the PKI commands and is used for validating the server certificate chain.

1. From configuration mode, configure the CA profile used for loading the certificate.

   ```
   [edit]
   user@host# set security pki ca-profile profile-name ca-identity ca-identity
   ```

2. From operational mode, load the certificate using PKI commands.

   ```
   user@host> request security pki ca-certificate load ca-profile profile-name filename
     filename
   ```

3. From configuration mode, disable the revocation check.

   NOTE: CRL checks are not supported; we recommend that you disable revocation checks.

   ```
   [edit]
   user@host# set security pki ca-profile profile-name ca-identity ca-identity
     revocation-check disable
   ```

4. From configuration mode, configure the loaded certificate as a trusted CA in the SSL proxy profile.

   ```
   [edit]
   ```

```
user@host# set services ssl proxy profile ssl-proxy-profile-name trusted-ca
  ca-profile-name
```

NOTE: More than one trusted CA can be configured for a profile.

5. (Optional) If you have multiple trusted CA certificates, you do not have to specify each trusted CA separately. You can load *all* the trusted CA certificates using the following command from configuration mode.

```
[edit]
user@host# set services ssl proxy profile ssl-proxy-profile-name trusted-ca all
```

NOTE: Alternatively, you can import a set of trusted CAs from your browser into the SRX Series device. See Knowledge Base article KB23144.

Importing a Root CA Certificate into a Browser

In order to have your browser or system automatically trust all certificates signed by the root CA configured in the SSL proxy profile, you must instruct your platform or browser to trust the CA root certificate.

To import a root CA certificate:

1. Generate a PEM format file for the configured root CA.

   ```
   request security pki local-certificate export certificate-id root-ca type pem filename
     path/file-name.pem
   ```

2. Import a root CA certificate into a browser.

   From Internet Explorer (version 8.0):

   a. From the Tools menu, choose **Internet Options**.

   b. On the Content tab, click **Certificates**.

   c. Select the **Trusted Root Certification Authorities** tab and click **Import**.

   d. In the Certificate Import Wizard, navigate to the required root CA certificate and select it.

   From Firefox (version 39.0):

   a. From the Tools menu, choose **Options**.

   b. From the Advanced menu, select the **Certificates** tab and click **View Certificate**.

c. In the Certificate Manager window, select the **Authorities** tab and click **Import**.

d. Navigate to the required root CA certificate and select it.

From Google Chrome (45.0):

a. From the Settings menu, choose **Show Advanced Settings**.

b. From the Advanced menu, select the **Certificates** tab and click **View Certificate**.

c. Under HTTPS/SSL, click **Manage Certificates**.

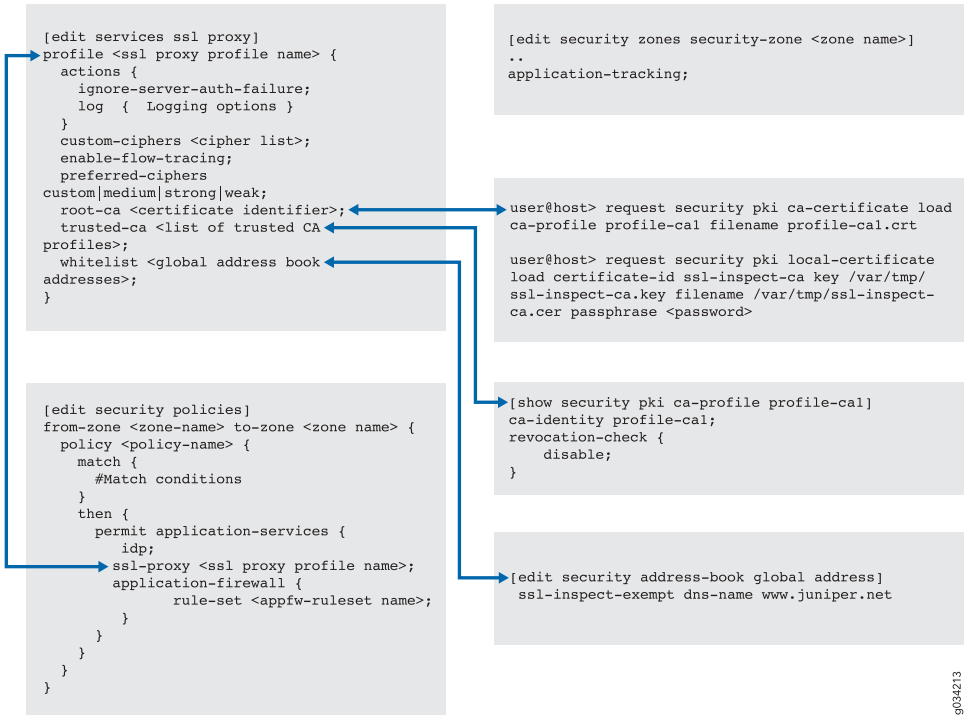
d. In the Certificate window, select **Trusted Root Certification Authorities** and click **Import**.

e. In the Certificate Import Wizard, navigate to the required root CA certificate and select it.

Applying an SSL Proxy Profile to a Security Policy

SSL proxy is enabled as an application service within a security policy. In a security policy, you specify the traffic that you want the SSL proxy enabled on as match criteria and then specify the SSL proxy CA profile to be applied to the traffic. Figure 5 on page 29 displays a graphical view of SSL proxy profile and security policy configuration.

Figure 5: Applying an SSL Proxy Profile to a Security Policy



To enable SSL proxy in a security policy:

1. Create a security policy and specify the match criteria for the policy. As match criteria, specify the traffic for which you want to enable SSL proxy.

   ```
   [edit]
   user@host# set security policies from-zone trust to-zone untrust policy policy-name
     match source-address source-address
   user@host# set security policies from-zone trust to-zone untrust policy policy-name
     match destination-address destination-address
   user@host# set security policies from-zone trust to-zone untrust policy policy-name
     match application application
   ```

2. Apply the SSL proxy profile to the security policy.

   ```
   [edit]
   user@host# set security policies from-zone trust to-zone untrust policy policy-name
     then permit application-services ssl-proxy profile-name profile-name
   ```

Creating a Whitelist of Exempted Destinations

SSL encryption and decryption are complicated and expensive procedures. You can selectively bypass SSL proxy processing for some sessions by configuring a whitelist. Typically, you would configure the whitelist to include trusted servers or domains with which you are very familiar. You might also include financial and banking sites that you are legally required to include.

Whitelists include addresses that you want to exempt from undergoing SSL proxy processing. For example, if you want to exempt all sessions to **www.mycompany.com**, then you would include it in the whitelist. To configure the whiltelist, you specify the domain that you want to exempt in an address book and then configure the address in the SSL proxy profile.

1. Configure the domain in the address book.

   ```
   [edit]
   user@host# set security address-book global address address dns-name
     www.mycompany.com
   ```

2. Specify the global address book address in the SSL proxy profile.

   ```
   [edit]
   user@host# set services ssl proxy profile profile-name whitelist address
   ```

Whitelist addresses and address sets are created under the global address book. The following type of addresses (from the global address book) are supported:

- IPv4 addresses (plain text). For example:

  ```
  [edit]
  user@host# set security address-book global address address-name ipv4-prefix
  ```

- IPv4 address range. For example:

  ```
  [edit]
  user@host# set security address-book global address address-name range-address
    range-low to range-high
  ```

- IPv4 wildcard. For example:

  ```
  [edit]
  user@host# set security address-book global address address-name wildcard-address
    addr/netmask
  ```

  Noncontiguous netmasks are not supported. For example:

  - 203.0.113.9/255.255.0.0 is supported.
  - 203.0.113.9/255.255.0.255 is NOT supported.

- IPv6 address (plain text). For example:

  ```
  [edit]
  user@host# set security address-book global address address-name ipv6-prefix
  ```

- DNS name. For example:

  ```
  [edit]
  user@host# set security address-book global address address-name dns-name
    domain-name
  ```

- Translated IP addresses. Sessions are whitelisted based on the actual IP address and not on the translated IP address. Because of this, in the whitelist configuration of the SSL proxy profile, the actual IP address should be provided and not the translated IP addresses.

  For example, consider a destination NAT rule that translates destination IP address **192.0.2.10/24** to **198.51.100.8/24** using the following commands:

  ```
  [edit]
  user@host# set security nat destination pool d1 address 198.51.100.8/24
  user@host# set security nat destination rule-set dst-nat rule r1 match
    destination-address 192.0.2.10/24
  user@host# set security nat destination rule-set dst-nat rule r1 then destination-nat
    pool d1
  ```

In this scenario, to exempt a session from SSL proxy inspection, the following IP address should be added to the whitelist:

```
[edit]
user@host# set security address-book global address ssl-proxy-exempted-addr
  192.0.2.10/24
user@host# set services ssl proxy profile ssl-inspect-profile whitelist
  ssl-proxy-exempted-addr
```

Configuring SSL Proxy Logging

When configuring SSL proxy, you can choose to set the option to receive some or all of the logs. SSL proxy logs contain the logical system name, SSL proxy whitelists, policy information, SSL proxy information, and other information that helps you troubleshoot when there is an error.

You can configure logging of *all* or specific events, such as error, warning, and information events. You can also configure logging of sessions that are whitelisted, dropped, ignored, or allowed after an error occurs.

```
[edit]
user@host# set services ssl proxy profile profile-name actions log all
user@host# set services ssl proxy profile profile-name actions log sessions-whitelisted
user@host# set services ssl proxy profile profile-name actions log sessions-allowed
user@host# set services ssl proxy profile profile-name actions log errors
```

You can use **enable-flow-tracing** option to enable debug tracing.

Configuring Ciphers

You can configure the following ciphers for an SSL proxy profile:

- **preferred-ciphers**—Preferred ciphers allow you to define an SSL cipher that can be used with acceptable key strength. Ciphers are divided in three categories depending on their key strength: strong, medium, or weak.
- **custom-ciphers**—Custom ciphers allow you to define your own cipher list. If you do not want to use one of the three categories, you can select ciphers from each of the categories to form a custom cipher set. To configure custom ciphers, you must set **preferred-ciphers** to custom.

  The following example shows how to create a custom cipher. In this example, you set **prefered-cipher** to custom and add the cipher list (rsa-with-3des-ede-cbc-sha and rsa-with-aes-256-cbc-sha):

  ```
  set services ssl proxy profile profile-name preferred-ciphers custom
  set services ssl proxy profile profile-name custom-ciphers rsa-with-3des-ede-cbc-sha
  set services ssl proxy profile profile-name custom-ciphers rsa-with-aes-256-cbc-sha
  ```

Exporting Certificates to a Specified Location

When a self-signed certificate is generated using a PKI command, the newly generated certificate is stored in a predefined location (**var/db/certs/common/local**).

Use the following command to export the certificate to a specific location (within the device). You can specify the certificate ID, the filename, and the type of file format (DER/PEM):

```
user@host> request security pki local-certificate export certificate-id certificate-id
user@host> request security pki local-certificate export filename filename
user@host> request security pki local-certificate export type der
```

Ignoring Server Authentication

Junos OS allows you to configure an option to ignore server authentication completely. If you configure your system to ignore authentication, then any errors encountered during server certificate verification at the time of the SSL handshake are ignored. Commonly ignored errors include the inability to verify CA signature, incorrect certificate expiration dates, and so forth. If this option is not set, all the sessions where the server sends self-signed certificates are dropped when errors are encountered.

We do not recommend using this option for authentication because configuring it results in websites not being authenticated at all. However, you can use this option to effectively identify the root cause of dropped SSL sessions.

From configuration mode, specify to ignore server authentication:

```
[edit]
user@host# set services ssl proxy profile profile-name actions ignore-server-auth-failure
```

Related Documentation

- SSL Proxy Overview on page 13
- Enabling Debugging and Tracing for SSL Proxy on page 33
- *Understanding Self-Signed Certificates*
- *show services ssl proxy statistics*
- *clear services ssl proxy statistics*

Enabling Debugging and Tracing for SSL Proxy

Debug tracing on both Routing Engine and the Packet Forwarding Engine can be enabled for SSL proxy by setting the following configuration:

```
user@host# set services ssl traceoptions
```

This feature is supported on the SRX550M, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices. Table 7 on page 33 shows the supported levels for trace options.

Table 7: Trace Levels

Cause Type	Description
Brief	Only error traces on both the Routing Engine and the Packet Forwarding Engine.
Detail	Packet Forwarding Engine–Only event details up to the handshake should be traced. Routing Engine–Traces related to commit. No periodic traces on the Routing Engine will be available
Extensive	Packet Forwarding Engine–Data transfer summary available. Routing Engine–Traces related to commit (more extensive). No periodic traces on the Routing Engine will be available.
Verbose	All traces are available.

Table 8 on page 33 shows the flags that are supported.

Table 8: Supported Flags in Trace

Cause Type	Description
cli-configuration	Configuration-related traces only.
initiation	Enable tracing on the SSL-I plug-in.

Table 8: Supported Flags in Trace *(continued)*

Cause Type	Description
proxy	Enable tracing on the SSL-Proxy-Policy plug-in.
termination	Enable tracing on the SSL-T plug-in.
selected-profile	Enable tracing only for profiles that have **enable-flow-tracing** set.

You can enable logs in the SSL proxy profile to get to the root cause for the drop. The following errors are some of the most common:

- Server certification validation error. Check the trusted CA configuration to verify your configuration.
- System failures such as memory allocation failures.
- Ciphers do not match.
- SSL versions do not match.
- SSL options are not supported.
- Root CA has expired. You need to load a new root CA.

You can enable the **ignore-server-auth-failure** option in the SSL proxy profile to ensure that certificate validation, root CA expiration dates, and other such issues are ignored. If sessions are inspected after the **ignore-server-auth-failure** option is enabled, the problem is localized.

Related Documentation

- SSL Proxy Overview on page 13
- Configuring SSL Proxy on page 23

CHAPTER 4

Configuring Pass-Through Authentication

Understanding Pass-Through Authentication

Pass-through user authentication is a form of active authentication; the user is prompted to enter a username and password when pass-through authentication is invoked. If the user's identity is validated, the user is allowed to pass through the firewall and gain access to the requested resources.

When a user attempts to initiate an HTTP, an HTTPS, an FTP, or a Telnet connection request that has a policy requiring authentication, the device intercepts the request and prompts the user to enter a username and password. Depending on the configuration, the device validates the username and password by checking them against those stored in the local database or on an external authentication server.

If an external authentication server is used, after the user's credentials are collected, they are processed through firewall user authentication. The following external authentication servers are supported:

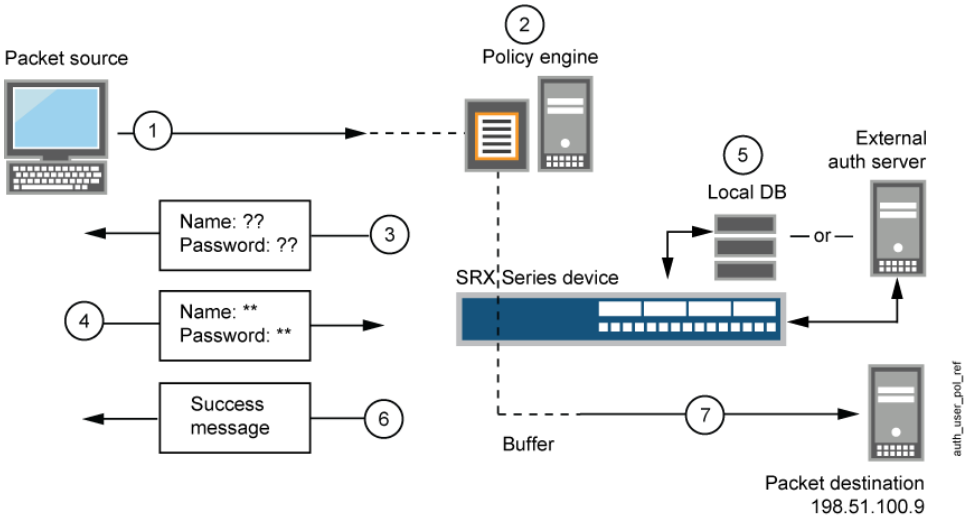
- RADIUS authentication and authorization (compatible with Juniper Steel-Belted Radius servers)

  You can use an external RADIUS server if, in addition to authentication, you want to obtain authorization information about the user's access right (what the user can do on the network).
- LDAP authentication only (supports LDAP version 3, compatible with Windows AD)
- SecurID authentication only (uses an RSA SecurID external authentication server)

A firewall user is a network user who must provide a username and password for authentication when initiating a connection across the firewall. You can put several user accounts together to form a user group, which you can store on the local database or on a RADIUS, an LDAP, or a SecurID server. When you reference an authentication user group and an external authentication server in a policy, the traffic matching the policy triggers an authentication check.

NOTE: You use family inet to assign an IPv4 address. You use family inet6 to assign an IPv6 address. An interface can be configured with both an IPv4 and an IPv6 address. For the sake of brevity, these examples use IPv4 addresses only.

Figure 6: Policy Lookup for a User



The steps in Figure 6 on page 36 are as follows:

1. A client user sends an FTP, an HTTP, an HTTPS, or a Telnet packet to 198.51.100.9.
2. The device intercepts the packet, notes that its policy requires authentication from either the local database or an external authentication server, and buffers the packet.
3. The device prompts the user for login information through FTP, HTTP, HTTPS, or Telnet.
4. The user replies with a username and password.
5. The device either checks for an authentication user account on its local database or sends the login information to the external authentication server as specified in the policy.
6. Finding a valid match (or receiving notice of such a match from the external authentication server), the device informs the user that the login has been successful.
7. For HTTP, HTTPS, or Telnet traffic, the device forwards the packet from its buffer to its destination IP address, 198.51.100.9/24. However, for FTP traffic, after successful authentication, the device closes the session and the user must reconnect to the FTP server at IP address 198.51.100.9/24.

After the device authenticates a user at a particular source IP address, it subsequently permits traffic—as specified in the policy requiring authentication through pass through—from any other user at that same address. This might be the case if the user

originates traffic from behind a NAT device that changes all original source addresses to a single translated address.

NOTE: The pass-through user authentication method is recommended in situations when security has a higher priority than convenience. This authentication method applies only to the session and child sessions matching the policy that triggered it. You can apply this method on Internet-facing links, if used with caution.

Related Documentation

- Firewall User Authentication Overview on page 9
- Understanding Web Authentication on page 51
- Example: Configuring Pass-Through Authentication on page 37

Example: Configuring Pass-Through Authentication

This example shows how to configure pass-through authentication to authenticate firewall users. A firewall user is a network user who must provide a username and password when initiating a connection across the firewall.

Pass-through authentication allows SRX Series administrators to restrict users who attempt to access a resource in another zone using FTP, Telnet, HTTP, or HTTPS. If the traffic matches a security policy whose action is pass-through authentication, the user is required to provide login information.

For HTTPS, to ensure security the HTTPS default certificate key size is 2048 bits. If you do not specify a certificate size, the default size is assumed.

Requirements

Before you begin, define firewall users. See Firewall User Authentication Overview.

This example uses the following hardware and software components:

- SRX Series device
- Firewall user's system
- Packet destination system

Overview

The pass-through authentication process is triggered when a client, referred to as a firewall user, attempts to initiate an FTP, a Telnet, or an HTTP session to access a resource

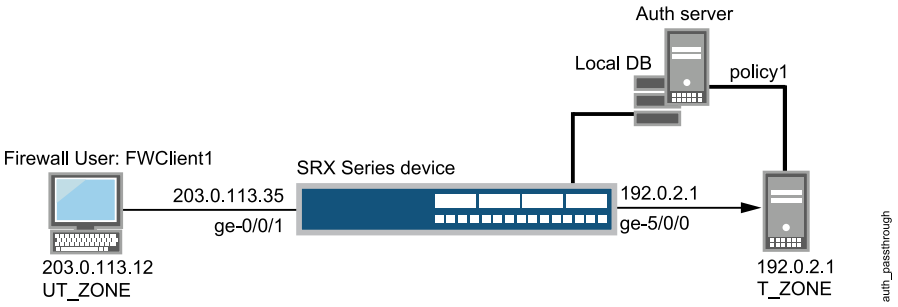
in another zone. The SRX Series firewall acts as a proxy for an FTP, a Telnet, an HTTP, or an HTTPS server so that it can authenticate the firewall user before allowing the user access to the actual FTP, Telnet, or HTTP server behind the firewall.

If traffic generated from a connection request sent by a firewall user matches a security policy rule bidirectionally and that rule specifies pass-through firewall authentication as the action of its **then** clause, the SRX Series device requires the firewall user to authenticate to a Junos OS proxy server.

If the authentication is successful, subsequent traffic from the same source IP address is automatically allowed to pass through the SRX Series device if the traffic matches the security policy tuples.

Figure 7 on page 38 shows the topology used in this example.

Figure 7: Configuring Pass-Through Firewall Authentication



NOTE: Although the topology shows use of an external server, it is not covered in the configuration. It is outside the scope of this example.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/1 unit 0 family inet address 203.0.113.35/24
set interfaces ge-5/0/0 unit 0 family inet address 192.0.2.1/24
set access profile FWAUTH client FWClient1 firewall-user password password
set access firewall-authentication pass-through default-profile FWAUTH
set access firewall-authentication pass-through telnet banner success "WELCOME TO JUNIPER TELNET SESSION"
set security zones security-zone UT-ZONE host-inbound-traffic system-services all
set security zones security-zone UT-ZONE interfaces ge-0/0/1.0 host-inbound-traffic protocols all
set security zones security-zone T-ZONE host-inbound-traffic system-services all
```

```
set security zones security-zone T-ZONE interfaces ge-5/0/0.0 host-inbound-traffic protocols
all
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match source-address any
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match destination-address
any
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match application junos-telnet
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 then permit
firewall-authentication pass-through client-match FWClient1
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure pass-through authentication:

1. Configure two interfaces and assign IP addresses to them.

   NOTE: For this example, it is optional to assign two addresses to the interfaces.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 unit 0 family inet address 203.0.113.35/24
   user@host# set interfaces ge-5/0/0 unit 0 family inet address 192.0.2.1/24
   ```

2. Create the FWAUTH access profile for the FWClient1 user, specify the user's password, and define a success banner for Telnet sessions.

   ```
   [edit access]
   user@host# set access profile FWAUTH client FWClient1 firewall-user password
     pwd
   user@host# set firewall-authentication pass-through default-profile FWAUTH
   user@host# set firewall-authentication pass-through telnet banner success
     "WELCOME TO JUNIPER TELNET SESSION"
   ```

3. Configure security zones.

   NOTE: For this example, it is optional to configure a second interface for a security zone.

   ```
   [edit security zones]
   user@host# set security-zone UT-ZONE host-inbound-traffic system-services all
   user@host# set security-zone UT-ZONE interfaces ge-0/0/1.0 host-inbound-traffic
     protocols all
   user@host# set security-zone T-ZONE host-inbound-traffic system-services all
   user@host# set security-zone T-ZONE interfaces ge-5/0/0.0 host-inbound-traffic
     protocols all
   ```

4. Assign security policy P1 to the security zones.

   ```
   [edit security policies]
   user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match
     source-address any
   ```

```
user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match
  destination-address any
user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match application
  junos-telnet
user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 then permit
  firewall-authentication pass-through client-match FWClient1
```

5. Use Telnet to authenticate the FWClient1 firewall user to host2.

```
user@FWClient1# run telnet 192.0.2.1/24
Trying 192.0.2.1/24...
Connected to 192.0.2.1/24
Escape character is '^]'.
Firewall User Authentication
Username: FWClient1
Password:$ABC123
WELCOME TO JUNIPER TELNET SESSION
Host1 (ttyp0)
login: user
Password: $ABC123
--- JUNOS 10.1R1.1 built 2009-10-12 13:30:18 UTC
%
```

Results From configuration mode, confirm your configuration by entering these commands.

- show interfaces
- show access
- show security zones
- show security policies

If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, the output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
user@host# show interfaces
  ge-0/0/1 {
    unit 0 {
      family inet {
        address 203.0.113.35;
      }
    }
  }
  ge-5/0/0 {
    unit 0 {
      family inet {
        address 192.0.2.1/24;
      }
    }
  }
...
user@host# show access
  profile FWAUTH {
```



```
    authentication-order password;
    client FWClient1 {
      firewall-user {
        password "$ABC123"; ## SECRET-DATA
      }
    }
  }
  firewall-authentication {
    pass-through {
      default-profile FWAUTH;
      telnet {
        banner {
          success "WELCOME TO JUNIPER TELNET SESSION";
        }
      }
    }
  }
```

```
user@host# show security zones
  security-zone UT-ZONE {
    host-inbound-traffic {
      system-services {
        all;
      }
    }
      interfaces {
        ge-0/0/1.0 {
          host-inbound-traffic {
            protocols {
              all;
            }
          }
        }
      }
      security-zone T-ZONE {
        host-inbound-traffic {
          system-services {
            all;
          }
        }
        interfaces {
          ge-5/0/0.0 {
            host-inbound-traffic {
              protocols {
                all;
              }
            }
          }
        }
      }
```

```
user@host# show security policies
...
from-zone UT-ZONE to-zone T-ZONE {
  policy P1 {
    match {
      source-address any;
```

```
        destination-address any;
        application junos-telnet;
      }
      then {
        permit {
          firewall-authentication {
            pass-through {
              client-match FWClient1;
            }
          }
        }
      }
    }
  }
```

If you are done configuring the device, enter commit from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying Firewall User Authentication and Monitoring Users and IP Addresses in the Authentication Table on page 42

Verifying Firewall User Authentication and Monitoring Users and IP Addresses in the Authentication Table

Purpose Display firewall authentication user history and verify the number of firewall users who successfully authenticated and the number of firewall users who failed to log in.

Action From operational mode, enter these **show** commands:

```
user@host> show security firewall-authentication history
History of firewall authentication data:
Authentications: 2
Id Source Ip Date Time Duration Status User
1 203.0.113.12 2010-10-12 21:24:02 0:00:24 Failed FWClient1
2 203.0.113.12 2010-10-12 21:24:48 0:00:22 Success FWClient1

user@host> show security firewall-authentication history identifier 1
Username: FWClient1
Source IP: 203.0.113.12
Authentication state: Success
Authentication method: Pass-through using Telnet
Access start date: 2010-10-12
Access start time: 21:24:02
Duration of user access: 0:00:24
Source zone: UT-ZONE
Destination zone: T-ZONE
Access profile: FWAUTH
Bytes sent by this user: 0
Bytes received by this user: 2660

user@host> show security firewall-authentication users
Firewall authentication data:
Total users in table: 1
```

```
Id Source Ip Src zone Dst zone Profile Age Status User
4 203.0.113.12     UT-ZONE  T-ZONE   FWAUTH     1 Success   FWClient1

user@host> show security firewall-authentication users identifier 3
Username: FWClient1
Source IP: 203.0.113.12
Authentication state: Success
Authentication method: Pass-through using Telnet
Age: 3
Access time remaining: 9
Source zone: UT-ZONE
Destination zone: T-ZONE
Access profile: FWAUTH
Interface Name: ge-0/0/1.0
Bytes sent by this user: 0
Bytes received by this user: 1521
```

Related Documentation

- Firewall User Authentication Overview on page 9
- Understanding Pass-Through Authentication on page 35

Example: Configuring HTTPS Traffic to Trigger Pass-Through Authentication

This example shows how to configure HTTPS traffic to trigger pass-through authentication. HTTPS is more secure than HTTP, so it has become more popular and is more widely used.

Requirements

This example uses the following hardware and software components:

- SRX Series device
- Two PCs running Linux and Open SSL. One PC acts as a client and another as an HTTPS server. The two PCs are used to create key files and to send traffic.
- Junos OS Release 12.1X44-D10 or later for high-end SRX Series Services Gateways and Junos OS Release 15.1X49-D40 or later for vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways.

NOTE: Starting in Junos OS Release 12.1X44-D10, HTTPS-based authentication is introduced on high-end SRX Series Services Gateways.

Starting in Junos OS Release 15.1X49-D40, HTTPS-based authentication is introduced on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways.

Before you begin:

An SRX Series device has to decode HTTPS traffic to trigger pass-through authentication. Then, SSL termination proxy creates and installs a private key file and a certification file. The following list describes the steps to create and install a private key file and a certification key file.

NOTE: If you have an official .crt file and .key file, then you can directly upload and install the files on the SRX Series device. If you do not have a .crt file and .key file, follow the procedure to create and install the files. Instructions specified in Step 1 and Step 2 must be run on a PC with Linux and OpenSSL installed. Instructions specified in Step 3 and Step 4 must be run in operational mode.

To create and install a private key file and a certification file:

1. On a PC create the **.key** file.

   openssl genrsa -out /tmp/device.key 1024

2. On a PC, create the **.crt** file.

   openssl req -new -x509 -days 365 -key /tmp/server.key -out /tmp/device.crt -subj "/C=CN/ST=BJ/L=BJ/O=JNPR/OU=CNRD/CN=203.0.113.11/emailAddress=device@mycompany.com"

3. Upload the **.key** and **.crt** files to an SRX Series device, and install the files on the device using the following command from operational mode:

   user@host> **request security pki local-certificate load filename /var/tmp/device.crt key /var/tmp/device.key certificate-id device**

Overview

Firewall authentication initiates a secure connection to be established across two devices. A network user must provide a username and password for authentication when initiating a connection across the firewall. Firewall authentication supports HTTPS traffic for pass-through authentication. HTTPS can secure HTTP firewall authentication traffic between users and the SRX Series device.

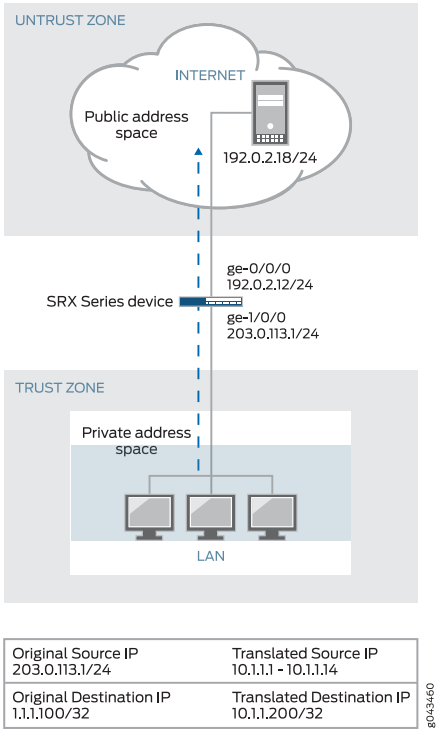
HTTPS is the secure version of HTTP, the protocol over which data is sent between the user and the device that the user is connected to. All communications between the user and the connected devices are encrypted. HTTPS is often used to protect highly confidential online transactions like online banking and online shopping order forms.

In this example, HTTPS traffic is used to trigger pass-through authentication because HTTPS is more secure than HTTP. For HTTPS traffic to trigger pass-through authentication you must first configure the SSL termination profile.

Figure 8 on page 45 shows an example of pass-through authentication using HTTPS traffic. In this example, a host or a user from an untrust zone tries to access resources on the trust zone. The SRX Series device uses HTTPS to collect the username and password

information. Subsequent traffic from the host or user is allowed or denied based on the result of this authentication.

Figure 8: Pass-Through Authentication Using HTTPS Traffic



Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/0 unit 0 family inet address 192.0.2.12/24
set interfaces ge-1/0/0 unit 0 family inet address 203.0.113.1/24
set security policies from-zone trust to-zone untrust policy p1 match source-address any
set security policies from-zone trust to-zone untrust policy p1 match destination-address
    any
set security policies from-zone trust to-zone untrust policy p1 match application any
set security policies from-zone trust to-zone untrust policy p1 then permit
    firewall-authentication pass-through access-profile local_pf
set security policies from-zone trust to-zone untrust policy p1 then permit
    firewall-authentication pass-through ssl-termination-profile ssl_pf
set security policies from-zone trust to-zone untrust policy p1 then log session-init
set security policies from-zone trust to-zone untrust policy p1 then log session-close
set security zones security-zone trust interfaces ge-0/0/0.0 host-inbound-traffic
    system-services all
set security zones security-zone trust interfaces ge-0/0/0.0 host-inbound-traffic protocols
    all
```

```
set security zones security-zone untrust interfaces ge-1/0/0.0 host-inbound-traffic
  system-services all
set security zones security-zone untrust interfaces ge-1/0/0.0 host-inbound-traffic protocols
  all
set access profile local_pf client user1 firewall-user password <password>
set access firewall-authentication pass-through default-profile local_pf
set services ssl termination profile ssl_pf server-certificate device
```

Step-by-Step Procedure

To configure HTTPS traffic to trigger pass-through authentication:

1. Configure interfaces and assign IP addresses.

   ```
   [edit interfaces]
   user@host# set ge-0/0/0 unit 0 family inet address 192.0.2.12/24
   user@host# set ge-1/0/0 unit 0 family inet address 203.0.113.1/24
   ```

2. Configure security policies to permit firewall authenticated traffic from zone trust to zone untrust.

   ```
   [edit security policies]
   user@host# set from-zone trust to-zone untrust policy p1 then permit
     firewall-authentication pass-through access-profile local_pf
   user@host# set from-zone trust to-zone untrust policy p1 then permit
     firewall-authentication pass-through ssl-termination-profile ssl_pf
   ```

3. Specify a policy action to take when a packet matches the criteria.

   ```
   [edit security policies]
   user@host# set from-zone trust to-zone untrust policy p1 match source-address
     any
   user@host# set from-zone trust to-zone untrust policy p1 match destination-address
     any
   user@host# set from-zone trust to-zone untrust policy p1 match application any
   user@host# set from-zone trust to-zone untrust policy p1 then log session-init
   user@host# set from-zone trust to-zone untrust policy p1 then log session-close
   ```

4. Configure security zones and assign interfaces.

   ```
   [edit security zones]
   user@host#  set security-zone trust interfaces ge-0/0/0.0 host-inbound-traffic
     protocols all
   user@host# set security-zone trust interfaces ge-0/0/0.0 host-inbound-traffic
     system-services all
   ```

5. Configure application services for zones.

   ```
   [edit security zones]
   user@host# set security-zone trust host-inbound-traffic system-services all
     protocols all
   user@host# set security-zone untrust host-inbound-traffic system-services all
     protocols all
   ```

6. Create an access profile and configure the client as a firewall user and set the password.

   ```
   [edit access]
   user@host# set profile local_pf client user1 firewall-user password <password>
   ```

7. Configure the type of firewall and the default profile name where the authentication settings are defined.

```
[edit access]
user@host# set firewall-authentication pass-through default-profile local_pf
```

8. Configure the SSL termination profile and enter a local certificate identifier name.

```
[edit services]
user@host# set ssl termination profile ssl_pf server-certificate device
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show security policies**, **show security zones**, **show access**, and **show services ssl termination** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show interfaces
...
interfaces
  ge-0/0/0 {
    unit 0 {
      family inet {
        address 192.0.2.12;
      }
    }
  }
  ge-1/0/0 {
    unit 0 {
      family inet {
        address 203.0.113.1/24;
      }
    }
  }
```

```
user@host# show security policies
...
policies
  from-zone trust to-zone untrust {
    policy p1 {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          firewall-authentication {
            pass-through {
              access-profile local_pf;
              ssl-termination-profile ssl_pf;
            }
          }
        }
        log {
          session-init;
          session-close;
        }
      }
    }
```

```
    }
}
```

user@host# **show security zones**

```
...
zones {
    security-zone trust {
        interfaces {
            ge-0/0/0.0 {
                host-inbound-traffic {
                    system-services {
                        all;
                    }
                    protocols {
                        all;
                    }
                }
            }
        }
    }
    security-zone untrust {
        interfaces {
            ge-1/0/0.0 {
                host-inbound-traffic {
                    system-services {
                        all;
                    }
                    protocols {
                        all;
                    }
                }
            }
        }
    }
```

user@host# **show access**

```
...
access {
    profile local_pf {
        client user1 {
            firewall-user {
                password password;
            }
        }
    }
    firewall-authentication {
        pass-through {
            default-profile local_pf;
        }
    }
```

user@host# **show services ssl termination**

```
...
services {
    ssl {
        termination {
            profile ssl_pf {
```

```
            server-certificate device;
        }
    }
}
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying the Configuration

Purpose Verify that the configuration is correct.

Action From operational mode, enter the **show security firewall-authentication users** command for identifier 1.

```
user@host> show security firewall-authentication users identifier 1
  Username: user1
  Source IP: 203.0.113.1/24
  Authentication state: Success
  Authentication method: Pass-through using HTTPS
  Age: 0
  Access time remaining: 10
  Lsys: root-logical-system
  Source zone: trust
  Destination zone: untrust
  Access profile: local_pf
  Interface Name: ge-0/0/0.0
  Bytes sent by this user: 946
  Bytes received by this user: 0
```

Meaning The **show security firewall-authentication users** command displays the firewall authentication user information for the specified identifier. If the output displays Pass-through using HTTPS in the Authentication method field and Success in the Authentication state field, then your configuration is correct.

Related Documentation

- Firewall User Authentication Overview on page 9
- Understanding Pass-Through Authentication on page 35
- Example: Configuring Pass-Through Authentication on page 37

CHAPTER 5

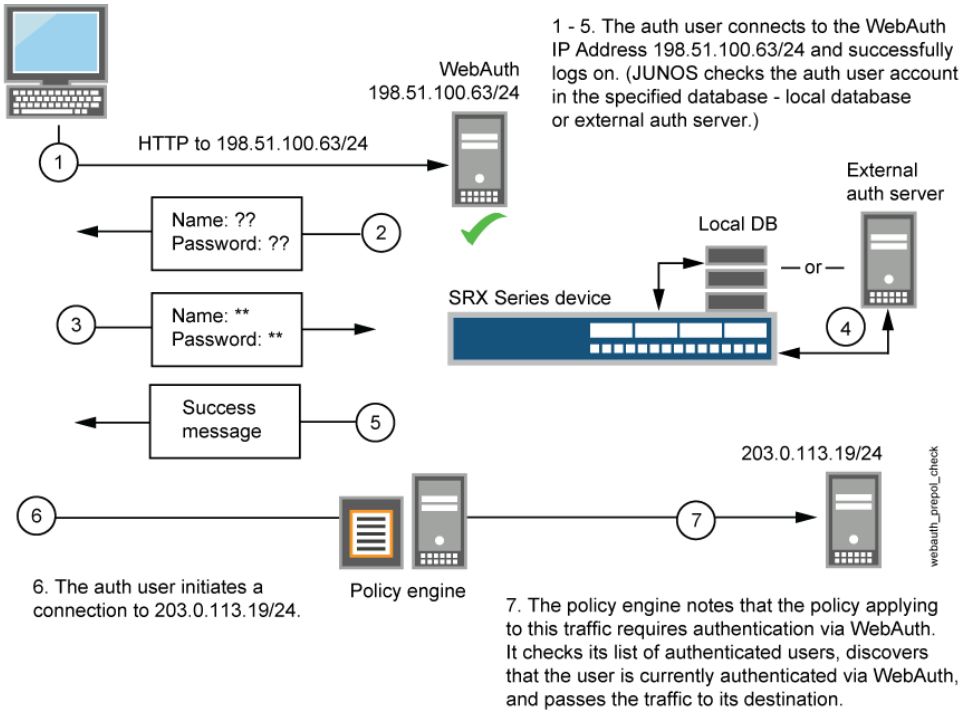
Configuring Web Authentication

Understanding Web Authentication

Web authentication is an alternative to pass-through user authentication. Instead of pointing to the resource that you want to connect to from your client browser, you point the browser to an IP address on the device that is enabled for Web authentication. This initiates an HTTP session to the IP address hosting the Web authentication feature on the device. The device then prompts you for your username and password and caches the result in the device. Later, when traffic encounters a Web authentication policy, you are allowed or denied access based on the prior Web authentication results, as shown in Figure 9 on page 52.

NOTE: You use family inet to assign an IPv4 address. You use family inet6 to assign an IPv6 address. An interface can be configured with both an IPv4 and an IPv6 address. For the sake of brevity, these examples use IPv4 addresses only.

Figure 9: Web Authentication Example



Follow these Web authentication guidelines:

- You can leave the default Web authentication server as the local database or you can choose an external authentication server for the role. The default Web authentication profile determines if the user authenticates using the local database or the external authentication server. An access profile stores usernames and passwords of users or points to external authentication servers where such information is stored.
- The Web authentication address must be in the same subnet as the interface that you want to use to host it. For example, if you want authentication users to connect using Web authentication through ethernet3, which has IP address 203.0.113.1/24, then you can assign Web authentication an IP address in the 203.0.113.0/24 subnet.
- You can put a Web authentication address in the same subnet as the IP address of any physical interface or virtual security interface (VSI). (For information about different types of interfaces, see *Security Zones and Interfaces Overview*.)
- You can put Web authentication addresses on multiple interfaces.
- After a device authenticates a user at a particular source IP address, it subsequently permits traffic—as specified in the policy requiring authentication through Web authentication—from any other user at that same address. This might be the case if the user originates traffic from behind a NAT device that changes all original source addresses to a single translated address.
- With Web authentication enabled, any HTTP traffic to the IP address will get the Web authentication login page instead of the administrator login page. Disabling this option

will show the administrator login page (assuming that **[system services web-management HTTP]** is enabled.

- We recommend that you have a separate primary or preferred IP address, if an address is used for Web authentication.

NOTE: The Web authentication method is recommended in situations when the client devices are immediately adjacent to the security gateway and there is high assurance that the client devices are not multiuser hosts. This authentication method is best applied to wireless links and DMZ, or conference room links.

Related Documentation

- Firewall User Authentication Overview on page 9
- Understanding Pass-Through Authentication on page 35
- Example: Configuring Web Authentication on page 53

Example: Configuring Web Authentication

This example shows how to enable Web authentication and set up a policy that allows access to a user when traffic encounters a policy that has Web authentication enabled.

Requirements

Before you begin:

- Define firewall users. See "Firewall User Authentication Overview" on page 9.
- Add the Web authentication HTTP flag under the interface's address hierarchy to enable Web authentication.

Overview

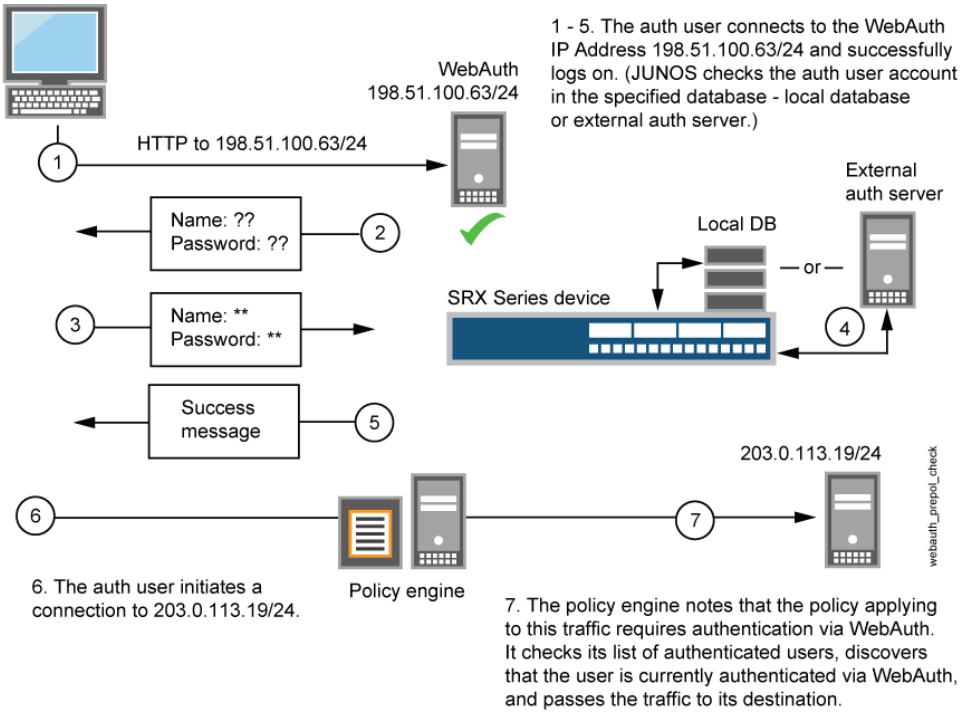
To enable Web authentication, you must specify the IP address of the device hosting the HTTP session. These settings are used if the firewall user accessing a protected resource wants to be authenticated by directly accessing the webserver or by Web authentication. The following instructions show how to set up a policy that allows access to the FWClient1 user when traffic encounters a policy that has Web authentication enabled (Policy-W). (See Figure 10 on page 54.) In this example, FWClient1 has already authenticated through the Web authentication login page.

The FWClient1 firewall user does the following to get authenticated:

a. Points the browser to the Web authentication IP (198.51.100.63/24) to get authenticated first

b. Starts traffic to access resources specified by the policy-W policy

Figure 10: Web Authentication Example



When you configure the device as described in these instructions and the user successfully authenticates, the screen illustrated in Figure 11 on page 54 appears.

Figure 11: Web Authentication Success Banner

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/1 unit 0 family inet address 198.51.100.23/24
set interfaces ge-0/0/1 unit 0 family inet address 198.51.100.23/24 web-authentication http
set interfaces fe-5/0/0 unit 0 family inet address 203.0.113.15/24
set access profile WEBAUTH client FWClient1 firewall-user password pwd
set access firewall-authentication web-authentication default-profile WEBAUTH
set access firewall-authentication web-authentication banner success "WEB AUTH LOGIN SUCCESS"
set security zones security-zone UT-ZONE host-inbound-traffic system-services all
set security zones security-zone UT-ZONE interfaces ge-0/0/1.0 host-inbound-traffic protocols all
```

```
set security zones security-zone T-ZONE host-inbound-traffic system-services all
set security zones security-zone T-ZONE interfaces ge-5/0/0.0 host-inbound-traffic protocols
all
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match source-address any
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match destination-address
any
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 match application any
set security policies from-zone UT-ZONE to-zone T-ZONE policy P1 then permit
firewall-authentication web-authentication client-match FWClient1
set system services web-management http interface ge-0/0/1.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure Web authentication:

1. Configure two interfaces and assign IP addresses to them.

   NOTE: For this example, it is optional to assign two addresses to the interfaces.

   ```
   [edit]

   user@host# set interfaces ge-0/0/1 unit 0 family inet address 198.51.100.23/24

   user@host# set interfaces ge-0/0/1 unit 0 family inet address 198.51.100.63/24
     web-authentication http

   user@host# set interfaces fe-5/0/0 unit 0 family inet address 203.0.113.15/24
   ```

2. Create the WEBAUTH access profile for the FWClient1 user, specify the user's password, and define a success banner.

   ```
   [edit access]

   user@host# set profile WEBAUTH client FWClient1 firewall-user password pwd

   user@host# set firewall-authentication web-authentication default-profile
     WEBAUTH

   user@host# set firewall-authentication web-authentication banner success "WEB
     AUTH LOGIN SUCCESS"
   ```

3. Configure security zones.

   NOTE: For this example, it is optional to configure a second interface for a security zone.

   ```
   [edit security zones]
   ```

```
user@host# set security-zone UT-ZONE host-inbound-traffic system-services all

user@host# set security-zone UT-ZONE interfaces ge-0/0/1.0 host-inbound-traffic
  protocols all

user@host# set security-zone T-ZONE host-inbound-traffic system-services all

user@host# set security-zone T-ZONE interfaces ge-5/0/0.0 host-inbound-traffic
  protocols all
```

4. Assign security policy P1 to the security zones.

```
[edit security policies]

user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match
  source-address any

user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match
  destination-address any

user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 match application
  any

user@host# set from-zone UT-ZONE to-zone T-ZONE policy P1 then permit
  firewall-authentication web-authentication client-match FWClient1
```

5. Activate the HTTP process (daemon) on your device.

```
[edit]

user@host# set system services web-management http interface ge-0/0/1.0
```

Results From configuration mode, confirm your configuration by entering these commands:

- **show interfaces**
- **show access**
- **show security zones**
- **show security policies**
- **show system services**

If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this **show** output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
user@host# show interfaces
...
}
ge-0/0/1{
    unit 0 {
        family inet {
```



```
            address 198.51.100.23/24 {
            address 198.51.100.63/24 {
                web-authentication http;
            }
        }
    }
}
fe-5/0/0 {
    unit 0 {
        family inet {
            address 198.51.100.14/24;
        }
    }
}
...

user@host# show access
profile WEBAUTH {
    client FWClient1 {
        firewall-user {
            password "$ABC123"; ## SECRET-DATA
        }
    }
}
firewall-authentication {
    web-authentication {
        default-profile WEBAUTH;
        banner {
            success "WEB AUTH LOGIN SUCCESS";
        }
    }
}

user@host# show security zones
...
}
security-zone UT-ZONE {
    host-inbound-traffic {
       system-services {
            all;
        }
    }
    interfaces {
        ge-0/0/1.0 {
            host-inbound-traffic {
                protocols {
                    all;
                }
            }
        }
    }
}
security-zone T-ZONE {
    host-inbound-traffic {
       system-services {
            all;
        }
    }
    interfaces {
        ge-5/0/0.0 {
```

```
            host-inbound-traffic {
                protocols {
                    all;
                }
            }
        }
    }
}
```

```
user@host# show security policies
...
from-zone UT-ZONE to-zone T-ZONE {
    policy P1 {
       match {
            source-address any;
            destination-address any;
            application any;
       }
       then {
            permit {
                firewall-authentication {
                    web-authentication {
                        client-match FWClient1;
                    }
                }
            }
       }
    }
}
```

```
user@host# show system services
...
ftp;
ssh;
telnet;
web-management {
    http {
        interface g-0/0/1.0;
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying Firewall User Authentication and Monitoring Users and IP Addresses in the Authentication Table on page 58

Verifying Firewall User Authentication and Monitoring Users and IP Addresses in the Authentication Table

Purpose Display firewall authentication user history and verify the number of firewall users who successfully authenticated and firewall users who failed to log in.

Action From operational mode, enter these **show** commands:

```
user@host> show security firewall-authentication history
user@host> show security firewall-authentication history identifier 1
user@host> show security firewall-authentication users
user@host> show security firewall-authentication users identifier 3
```

```
user@host> show security firewall-authentication history
History of firewall authentication data:
Authentications: 1
Id Source Ip Date Time Duration Status User
5 198.51.100.75      2010-04-24 01:08:57 0:10:30    Success  FWClient1
```

```
user@host> show security firewall-authentication history identifier 1
Username: FWClient1
Source IP: 198.51.100.752
Authentication state: Success
Authentication method: Web-authentication
Access start date: 2010-10-12
Access start time: 21:24:02
Duration of user access: 0:00:24
Source zone: N/A
Destination zone: N/A
Access profile: WEBAUTH
Bytes sent by this user: 0
Bytes received by this user: 2660
```

```
user@host> show security firewall-authentication users
Firewall authentication data:
Total users in table: 1
Id Source Ip Src zone Dst zone Profile Age Status User
4 198.51.100.75     N/A  N/A   WEBAUTH     1 Success   FWClient1
```

```
user@host> show security firewall-authentication users identifier 3
Username: FWClient1
Source IP: 198.51.100.75
Authentication state: Success
Authentication method: Web-authentication
Age: 3
Access time remaining: 9
Source zone: N/A
Destination zone: N/A
Access profile: WEBAUTH
Interface Name: ge-0/0/1.0
Bytes sent by this user: 0
Bytes received by this user: 1521
```

Related Documentation

- Understanding Web Authentication on page 51
- Understanding Firewall Authentication Banner Customization on page 83
- *Security Zones and Interfaces Overview*

Example: Configuring HTTPS Traffic to Trigger Web Authentication

This example shows how to configure HTTPS traffic to trigger Web authentication. HTTPS is widely used for Web authentication because it is more secure than HTTP.

Requirements

Before you begin:

This example uses the following hardware and software components:

- SRX Series device
- Two PCs with Linux and Open SSL installed. One PC acts as a client and another as an HTTPS server. The two PCs are used to create key files and to send traffic.
- Junos OS Release 12.1X44-D10 or later for high-end SRX Series Services Gateways and Junos OS Release 15.1X49-D40 or later for vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways.

An SRX Series device has to decode the HTTPS traffic to trigger Web authentication. The following list describes the steps to create and install a private key file and a certification key file.

NOTE: If you have an official .crt file and .key file, then you can directly upload and install the files on the SRX Series device. If you do not have a .crt file and .key file, then follow the procedure to create and install the files. Instructions specified in Step 1 and Step 2 must be run on a PC which has Linux and OpenSSL installed. Instructions specified in Step 3 and Step 4 must be run in operational mode.

1. From the PC, create the **.key** file.

   openssl genrsa -out /tmp/device.key 1024

2. From the PC, create the **.crt** file.

   openssl req -new -x509 -days 365 -key /tmp/server.key -out /tmp/device.crt -subj "/C=CN/ST=BJ/L=BJ/O=JNPR/OU=CNRD/CN=203.0.113.22/emailAddress=device@mycomany.com"

3. From the SRX Series device, upload the **.key** and **.crt** files and install the files on the device using the following command:

   user@host> **request security pki local-certificate load filename /var/tmp/device.crt key /var/tmp/device.key certificate-id device**

Overview

Firewall authentication initiates a secure connection to be established across two devices. A network user must provide a username and password for authentication when initiating a connection across the firewall. Firewall authentication supports HTTPS traffic for pass-through authentication. HTTPS can secure HTTP firewall authentication traffic between users and the SRX Series device.

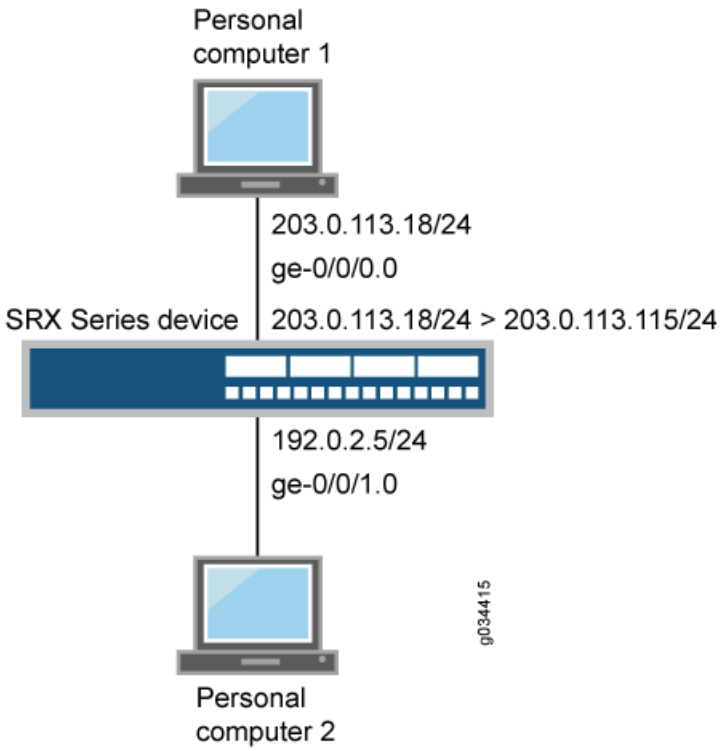
HTTPS is the secure version of HTTP, the protocol over which data is sent between the user and the device that the user is connected to. All communications between the user and the connected devices are encrypted. HTTPS is often used to protect highly confidential online transactions like online banking and online shopping order forms.

In this example, HTTPS traffic is used to trigger Web authentication because HTTPS is more secure than HTTP.

The user uses HTTPS to access an IP address on the device that is enabled for Web authentication. In this scenario, the user does not use HTTPS to access the IP address of the protected resource. The user is prompted for a username and password, which are verified by the device. Subsequent traffic from the user or host to the protected resource is allowed or denied based on the results of this Web authentication.

Figure 12 on page 61 shows an example of Web authentication using HTTPS traffic.

Figure 12: Web Authentication Using HTTPS Traffic



Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set system services web-management https pki-local-certificate device
set interfaces ge-0/0/0 unit 0 family inet address 203.0.113.18/24
set interfaces ge-0/0/0 unit 0 family inet address 203.0.113.115/24 web-authentication
  https
set interfaces ge-0/0/1 unit 0 family inet address 192.0.2.5/24
set security policies from-zone trust to-zone untrust policy p1 match source-address any
set security policies from-zone trust to-zone untrust policy p1 match destination-address
  any
set security policies from-zone trust to-zone untrust policy p1 match application any
set security policies from-zone trust to-zone untrust policy p1 then permit
set access profile local_pf client user1 firewall-user password user1
set access firewall-authentication web-authentication default-profile local_pf
set security policies from-zone trust to-zone untrust policy p1 then permit
  firewall-authentication web-authentication
```

Step-by-Step Procedure

To configure HTTPS traffic to trigger Web authentication:

1. Enable Web-management support to HTTPS traffic.

   ```
   [edit system services]
   user@host# set web-management https pki-local-certificate device
   ```

2. Configure interfaces and assign IP addresses. Enable Web authentication at ge-0/0/0 interface.

   ```
   [edit interfaces]
   user@host# set ge-0/0/0 unit 0 family inet address 203.0.113.18/24
   set ge-0/0/0 unit 0 family inet address 203.0.113.115/24 web-authentication https
   user@host# set ge-0/0/1 unit 0 family inet address 192.0.2.5/24
   ```

3. Configure security policies to permit firewall authenticated traffic from zone trust to zone untrust.

   ```
   [edit security policies]
   user@host# set from-zone trust to-zone untrust policy p1 match source-address
     any destination-address any application any
   user@host# set security policies from-zone trust to-zone untrust policy p1 then
     permit
   ```

4. Create an access profile, configure the client as a firewall user, and set the password.

   ```
   [edit access]
   user@host# set profile local_pf client user1 firewall-user password user1
   ```

5. Configure the type of firewall authentication settings.

   ```
   [edit access]
   user@host# set firewall-authentication web-authentication default-profile local_pf
   ```

6. Specify a policy action to take when a packet matches the criteria.

   ```
   [edit security policies]
   ```

```
user@host# set from-zone trust to-zone untrust policy p1 then permit
  firewall-authentication web-authentication
```

Results From configuration mode, confirm your configuration by entering the **show system services**, **show interfaces**, **show security policies**, and **show access** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show system services
  web-management {
  https {
  pki-local-certificate device;
  }
}
user@host# show interfaces
  ge-0/0/0 {
  unit 0 {
    family inet {
      address 203.0.113.115/24 {
        web-authentication https;
      }
    }
}
  ge-0/0/1 {
    unit 0 {
      family inet {
        address 192.0.2.5/24;
      }
    }
  }
user@host# show security policies
from-zone trust to-zone untrust {
  policy p1 {
    match {
      source-address any;
      destination-address any;
      application any;
    }
    then {
      permit {
        firewall-authentication {
          web-authentication;
        }
      }
    }
  }
}
user@host# show access
  profile local_pf {
    client user1 {
      firewall-user {
        password "user1";
      }
```

```
    }
  }
    firewall-authentication {
      web-authentication {
        default-profile local_pf;
      }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying the Configuration

Purpose Verify that the configuration is correct.

Action From operational mode, enter the **show security firewall-authentication users identifier** ***identifier*** command.

Sample Output

```
user@host> show security firewall-authentication users identifier 1
  Username: user1
  Source IP: 203.1.113.102
  Authentication state: Success
  Authentication method: Web-authentication
  Age: 0
  Access time remaining: 10
  Lsys: root-logical-system
  Source zone: N/A
  Destination zone: N/A
  Access profile: local_pf
  Bytes sent by this user: 0
  Bytes received by this user: 0
```

Meaning The **show security firewall-authentication users identifier** ***identifier*** command displays the firewall authentication user information using the identifier ID of the user. If the authentication method parameter displays Web authentication and the authentication state parameter displays success in your output then your configuration is correct.

Related Documentation

- Firewall User Authentication Overview on page 9
- Understanding Web Authentication on page 51
- Example: Configuring Web Authentication on page 53

CHAPTER 6

Configuring External Authentication Servers

Understanding External Authentication Servers

Authentication, authorization, and accounting (AAA) servers provide an extra level of protection and control for user access in the following ways:

- Authentication determines the firewall user.
- Authorization determines what the firewall user can do.
- Accounting determines what the firewall user did on the network.

You can use authentication alone or with authorization and accounting. Authorization always requires a user to be authenticated first. You can use accounting alone, or with authentication and authorization.

Once the user's credentials are collected, they are processed using firewall user authentication, which supports the following types of servers:

- Local authentication and authorization
- RADIUS authentication and authorization (compatible with Juniper Steel-Belted Radius server)
- LDAP authentication only (supports LDAP version 3 and is compatible with Windows AD)
- SecurID authentication only (using an RSA SecurID external authentication server)

NOTE: Junos OS also supports administrative authentication using local, RADIUS, and TACACS+ servers.

This topic includes the following sections:

Understanding SecurID User Authentication

SecurID is an authentication method that allows users to enter either static or dynamic passwords as their credentials. A dynamic password is a combination of a user's PIN and a randomly generated token that is valid for a short period of time, approximately one minute. A static password is set for the user on the SecurID server. For example, the SecurID server administrator might set a temporary static password for a user who lost his or her SecurID token.

When a user attempts to access a resource protected by a policy and SecurID is configured in the profile **authentication-order** parameter as either the only authentication mode or the first one to be used, the device forwards the user's credentials to the SecurID server for authentication. If the user enters valid values, the user is allowed access to the requested resource.

NOTE: **The SecurID server includes a feature that presents a user with a challenge if the user provides wrong credentials repeatedly. However, Junos OS does not support the challenge feature. Instead, the SecurID server administrator must resynchronize the RSA token for the user.**

For SecurID, you configure information about the Juniper Networks device on the SecurID server, and this information is exported to a file called sdconf.rec.

To install the sdconf.rec file on the device, you must use an out-of-band method such as FTP. Install the file in a directory whose files are not deleted regularly. Do not put it in a temporary directory. For example, you might install it in /var/db/secureid/server1/sdconf.rec.

The sdconf.rec file contains information that provides the Juniper Networks device with the address of the SecurID server. You do not need to configure this information explicitly when you configure the SecurID server to be used as the external authentication server.

Related Documentation

- Firewall User Authentication Overview on page 9
- Example: Configuring RADIUS and LDAP User Authentication on page 66
- Example: Configuring SecurID User Authentication on page 71
- Example: Deleting the SecurID Node Secret File on page 74

Example: Configuring RADIUS and LDAP User Authentication

This example shows how to configure a device for external authentication.

Requirements

Before you begin, create an authentication user group.

Overview

You can put several user accounts together to form a user group, which you can store on the local database or on a RADIUS, an LDAP, or a SecurID server. When you reference an authentication user group and an external authentication server in a policy, the traffic matching the policy provokes an authentication check.

This example shows how access profile Profile-1 is configured for external authentication. Two RADIUS servers and one LDAP server are configured in the access profile. However, the order of authentication specifies RADIUS server only, so if the RADIUS server authentication fails, then the firewall user fails to authenticate. The local database is not accessed.

NOTE: If the firewall clients are authenticated by the RADIUS server, then the group-membership VSA returned by the RADIUS server should contain alpha, beta, or gamma client groups in the RADIUS server configuration or in the access profile, Profile-1. Access profiles store usernames and passwords of users or point to external authentication servers where such information is stored.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set access profile Profile-1 authentication-order radius
set access profile Profile-1 client Client-1 client-group alpha
set access profile Profile-1 client Client-1 client-group beta
set access profile Profile-1 client Client-1 client-group gamma
set access profile Profile-1 client Client-1 firewall-user password pwd
set access profile Profile-1 client Client-2 client-group alpha
set access profile Profile-1 client Client-2 client-group beta
set access profile Profile-1 client Client-2 firewall-user password pwd
set access profile Profile-1 client Client-3 firewall-user password pwd
set access profile Profile-1 client Client-4 firewall-user password pwd
set access profile Profile-1 session-options client-group alpha
set access profile Profile-1 session-options client-group beta
set access profile Profile-1 session-options client-group gamma
set access profile Profile-1 session-options client-idle-timeout 255
set access profile Profile-1 session-options client-session-timeout 4
set access profile Profile-1 ldap-options base-distinguished-name
CN=users,DC=junos,DC=juniper,DC=net
set access profile Profile-1 ldap-options search search-filter sAMAccountName=
```

```
set access profile Profile-1 ldap-options search admin-search distinguished-name
cn=administrator,cn=users,dc=junos,dc=juniper,dc=net
set access profile Profile-1 ldap-options search admin-search password pwd
set access profile Profile-1 ldap-server 203.0.113.39/24
set access profile Profile-1 radius-server 203.0.113.62/24 secret example-secret
set access profile Profile-1 radius-server 203.0.113.62/24 retry 10
set access profile Profile-1 radius-server 203.0.113.27/24 secret juniper
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure a device for external authentication:

1. Specify the RADIUS server for external authentication order.

   ```
   [edit]

   user@host# set access profile Profile-1 authentication-order radius
   ```

2. Configure Client1-4 firewall users and assign the Client-1 firewall user and Client-2 firewall user to client groups.

   ```
   [edit access profile Profile-1]

   user@host# set client Client-1 client-group alpha

   user@host# set client Client-1 client-group beta

   user@host# set client Client-1 client-group gamma

   user@host# set client Client-1 firewall-user password pwd

   user@host# set client Client-2 client-group alpha

   user@host# set client Client-2 client-group beta

   user@host# set client Client-2 firewall-user password pwd

   user@host# set client Client-3 firewall-user password pwd

   user@host# set client Client-4 firewall-user password pwd
   ```

3. Configure client groups in the session options.

   ```
   [edit access profile Profile-1]

   user@host# set session-options client-group alpha

   user@host# set session-options client-group beta

   user@host# set session-options client-group gamma
   ```

```
user@host# set session-options client-idle-timeout 255

user@host# set session-options client-session-timeout 4
```

4. Configure the IP address for the LDAP server and server options.

```
[edit access profile Profile-1]

user@host# set ldap-options base-distinguished-name
  CN=users,DC=junos,DC=mycompany,DC=net

user@host# set ldap-options search search-filter sAMAccountName=

user@host# set ldap-options search admin-search password pwd

user@host# set ldap-options search admin-search distinguished-name
  cn=administrator,cn=users,dc=junos,dc=mycompany,dc=net

user@host# set ldap-server 203.0.113.39/24
```

5. Configure the IP addresses for the two RADIUS servers.

```
[edit access profile Profile-1]

user@host# set radius-server 203.0.113.62/24 secret pwd

user@host# set radius-server 203.0.113.62/24 retry 10

user@host# set radius-server 203.0.113.27/24 secret pwd
```

Results From configuration mode, confirm your configuration by entering the **show access profile Profile-1** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show access profile Profile-1
authentication-order radius;
client Client-1 {
    client-group [ alpha beta gamma ];
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-2 {
    client-group [ alpha beta ];
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-3 {
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-4 {
    firewall-user {
```

```
        password "$ABC123"; ## SECRET-DATA
    }
}
session-options {
    client-group [ alpha beta gamma ];
    client-idle-timeout 255;
    client-session-timeout 4;
}
ldap-options {
    base-distinguished-name CN=users,DC=junos,DC=juniper,DC=net;
    search {
        search-filter sAMAccountName=;
        admin-search {
            distinguished-name cn=administrator,cn=users,dc=junos,
       dc=mycompany,dc=net; password "$ABC123"; ## SECRET-DATA
        }
    }
}
ldap-server {
    203.0.113.39/24 ;
}
radius-server {
    203.0.113.62/24  {
        secret "$ABC123"; ## SECRET-DATA
        retry 10;
    }
    203.0.113.27/24  {
        secret "$ABC123"; ## SECRET-DATA
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Troubleshooting with Logs on page 70

Troubleshooting with Logs

Purpose Use these logs to identify any issues.

Action From operational mode, enter the **show log messages** command and the **show log dcd** command.

Related Documentation

- Understanding External Authentication Servers on page 65
- Example: Configuring SecurID User Authentication on page 71
- Example: Deleting the SecurID Node Secret File on page 74

Example: Configuring SecurID User Authentication

This example shows how to configure SecurID as the external authentication server.

Requirements

Before you begin, create an authentication user group.

Overview

SecurID is an authentication method that allows users to enter either static or dynamic passwords as their credentials. A dynamic password is a combination of a user's PIN and a randomly generated token that is valid for a short period of time, approximately one minute. A static password is set for the user on the SecurID server. For example, the SecurID server administrator might set a temporary static password for a user who lost his or her SecurID token.

When a user attempts to access a resource protected by a policy and SecurID is configured in the profile **authentication-order** parameter as either the only authentication mode or the first one to be used, the device forwards the user's credentials to the SecurID server for authentication. If the user enters valid values, the user is allowed access to the requested resource.

Specify that Server-1 is to be used as the SecurID server and that its configuration file resides on the device in the **/var/db/securid/Server-1/sdconf.rec** file. From configuration mode, enter this command:

```
user@host# set access securid-server Server-1 configuration-file
  "/var/db/securid/Server-1/sdconf.rec"
```

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set access profile Profile-2 authentication-order securid
set access profile Profile-2 client Client-1 client-group alpha
set access profile Profile-2 client Client-1 client-group beta
set access profile Profile-2 client Client-1 client-group gamma
set access profile Profile-2 client Client-1 firewall-user password pwd
set access profile Profile-2 client Client-2 client-group alpha
set access profile Profile-2 client Client-2 client-group beta
set access profile Profile-2 client Client-2 firewall-user password pwd
set access profile Profile-2 client Client-3 firewall-user password pwd
```

```
set access profile Profile-2 client Client-4 firewall-user password pwd
set access profile Profile-2 session-options client-group alpha
set access profile Profile-2 session-options client-group beta
set access profile Profile-2 session-options client-group gamma
set access profile Profile-2 session-options client-idle-timeout 255
set access profile Profile-2 session-options client-session-timeout 4
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure SecurID as the external authentication server:

1. For the Profile-2 profile, configure SecurID as the server to be used for external authentication.

   ```
   [edit]

   user@host# set access profile Profile-2 authentication-order securid
   ```

   To share a single SecurID server across multiple profiles, for each profile set the **authentication-order** parameter to include **securid** as the authentication mode.

2. Configure clients 1 through 4 as firewall users, and assign Client-1 and Client-2 to client groups.

   ```
   [edit access profile Profile-2]

   user@host# set client Client-1 client-group alpha

   user@host# set client Client-1 client-group beta

   user@host# set client Client-1 client-group gamma

   user@host# set client Client-1 firewall-user password pwd

   user@host# set client Client-2 client-group alpha

   user@host# set client Client-2 client-group beta

   user@host# set client Client-2 firewall-user password pwd

   user@host# set client Client-3 firewall-user password pwd

   user@host# set client Client-4 firewall-user password pwd
   ```

3. Configure client groups in the session options.

   ```
   [edit access profile Profile-2]

   user@host# set session-options client-group alpha

   user@host# set session-options client-group beta
   ```

```
user@host# set session-options client-group gamma

user@host# set session-options client-idle-timeout 255

user@host# set session-options client-session-timeout 4
```

Results From configuration mode, confirm your configuration by entering the **show access profile Profile-2** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show access profile Profile-2
authentication-order securid;
client Client-1 {
    client-group [ alpha beta gamma ];
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-2 {
    client-group [ alpha beta ];
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-3 {
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
client Client-4 {
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
session-options {
    client-group [alpha beta gamma];
    client-idle-timeout 255;
    client-session-timeout 4;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Troubleshooting with Logs on page 73

Troubleshooting with Logs

Purpose Use these logs to identify any issues.

Action From operational mode, enter the **show log messages** command and the **show log dcd** command.

Troubleshooting

Troubleshooting Unsuccessful Authentication In a Dynamic VPN Configuration

Problem Device fails to locate client address in a dynamic VPN configuration.

Solution

1. Verify that the device host name, the domain-search, and the name server are configured properly.

   ```
   [edit system]

   user@host# set host-name srxhost.example.net

   user@host# set domain-search domain.example.net

   user@host# set name-server 203.0.113.11
   ```

2. Verify that the device host name is getting resolved on the RSA server.

Related Documentation

Example: Deleting the SecurID Node Secret File

This example shows how to delete the node secret file.

Requirements

Before you begin, confirm that it is necessary to delete the SecurID node secret file.

Overview

When the Juniper Networks device initially communicates successfully with the SecurID server, a node secret file is created for it automatically. The file is created as a result of the handshake between the Juniper Networks device and the SecurID server after the software authenticates the first user successfully. All subsequent communication between the Juniper Networks device and the SecurID server relies on this secret as a representation of trust between the two nodes instead of repeating the handshake with each authentication request.

Under normal circumstances you should not delete the node secret file. In the rare case that you must do so, for example, to debug a serious problem, you can use the **clear** command to remove the file.

WARNING: If you delete the file, you must deselect a box on the SecurID server to indicate that the node secret file for the Juniper Networks device and the SecurID server no longer exists. Otherwise, authentication attempts will fail.

Configuration

Step-by-Step Procedure

To delete the node secret file:

1. Use the **clear** command to remove the node secret file. During subsequent user authentication, the device reestablishes a shared secret with the SecurID server and re-creates the node secret file. From operational mode, enter the **clear network-access** command to clear the **securid-node-secret-file** for the Juniper Networks device.

   ```
   user@host> clear network-access securid-node-secret-file
   ```

2. From operational mode, confirm your deletion by entering the **show network-access securid-node-secret-file** command. If the output does not display, repeat the instructions in this example to correct it.

   ```
   user@host> show network-access securid-node-secret-file
   ```

Verification

Verify the deletion by entering the **show network-access securid-node-secret-file** command.

Related Documentation

- Understanding External Authentication Servers on page 65
- Example: Configuring SecurID User Authentication on page 71

Enabling LDAP Authentication with TLS/SSL for Secure Connections

Beginning with Junos OS Release 15.1X49-D70, SRX Series devices support the Transport Layer Security (TLS) StartTLS extension for LDAP for firewall user authentication and the integrated user firewall authentication for obtaining username and role information through firewall authentication. StartTTLS allows protocol data transfers between the LDAP server and client over the TLS layer after successful negotiation between the peers. StartTLS upgrades an existing insecure LDAP connection to a secure TLS/SSL connection.

NOTE: SRX Series devices support TLSv1.1 and TLS v1.2 to use LDAP authentication with TLS/SSL.

With StartTLS for LDAP, a secure communication can be provided with the following sets of ciphers that provide increasingly strong security:

- High encryption cipher: AES256-SHA,DES-CBC3-SHA
- Medium encryption ciphers: High encryption cipher + RC4-SHA:RC4-MD5:AES128-SHA
- Medium encryption ciphers: Medium encryption ciphers + DES-CBC-SHA:EXP1024-DES-CBC-SHA:EXP1024-RC4-SHA: EXP1024-RC4-MD5:EXP-DES-CBC-SHA:EXP-RC4-MD5

Implementation of StartTLS on LDAP is interoperable with the following standard LDAP servers:

- Windows Active Directory
- Novell e-Directory
- Sun LDAP
- OpenLDAP

By default, LDAP traffic is not transmitted securely. You can set LDAP traffic to be confidential and secure by using Secure Sockets Layer/Transport Layer Security (SSL/TLS) technology.

To configure TLS parameters as a part of LDAP server configuration:

1. Define TLS type as **start-tls** to configure LDAP over StartTLS.

   [edit]

   user@host# **set access profile** *profile-name* **ldap-server** *ip-address* **tls-type start-tls**

2. Configure the peer host name to be authenticated.

   [edit]

   user@host# **set access profile** *profile-name* **ldap-server** *ip-address* **tls-peer-name** *peer-name*

3. Specify the timeout value on the TLS handshake. You can enter 3 through 90 seconds.

   [edit]

   user@host# **set access profile** *profile-name* **ldap-server** *ip-address* **tls-timeout**

4. Specify TLS version (v1.1 and v1.2 are supported) as the minimum protocol version enabled in connections. By default, SRX Series device uses TLS v1.2 to negotiate the TLS connection with the LDAP server:

   [edit]

   user@host# **set access profile** *profile-name* **ldap-server** *ip-address* **tls-min-version** *supported-tls-version*

NOTE: SRX Series devices support an additional check on the LDAP server's certificate during the TLS handshake for LDAP authentication by default. If the validation of the server certificate is not required, you can use the following configuration to ignore the validation of server's certificate and accept the certificate without checking:

```
[edit]
user@host# set access profile profile-name ldap-server ip-address
  no-tls-certificate-check
```

By default, the no-tls-certificate-check remains disabled.

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11
- LDAP Functionality in Integrated User Firewall on page 150

CHAPTER 7

Configuring Client Groups

Understanding Client Groups for Firewall Authentication

To manage a number of firewall users, you can create user or client groups and store the information either on the local Juniper Networks device or on an external RADIUS or LDAP server.

A client group is a list of groups to which the client belongs. As with client-idle timeout, a client group is used only if the external authentication server does not return a value in its response. (For example, LDAP servers do not return such information.)

The RADIUS server sends the client's group information to the Juniper Networks device using Juniper VSA (46). The client-match portion of the policy accepts a string that can be either the username or the groupname to which the client belongs.

The reason to have a single database for different types of clients (except admins) is based on the assumption that a single client can be of multiple types. For example, a firewall user client can also be an L2TP client.

Related Documentation

- Firewall User Authentication Overview on page 9
- Example: Configuring Local Users for Client Groups on page 79

Example: Configuring Local Users for Client Groups

This example shows how to configure a local user for client groups in a profile.

Requirements

Before you begin, create an access profile.

Overview

A client group is a list of groups to which the client belongs. As with client-idle timeout, a client group is used only if the external authentication server does not return a value in its response (for example, LDAP servers do not return such information).

This example shows how to configure a local user called Client-1 for client groups G1, G2, and G3 in a profile called Managers. Within this example, client groups are configured for a client. If a client group is not defined for the client, then the client group under the **access profile session-options** hierarchy is used.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set access profile Managers client Client-1 client-group G1
set access profile Managers client Client-1 client-group G2
set access profile Managers client Client-1 client-group G3
set access profile Managers client Client-1 firewall-user password pwd
set access profile Managers session-options client-group G1
set access profile Managers session-options client-group G2
set access profile Managers session-options client-group G3
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure a local user for client groups in a profile:

1. Configure the firewall user profile Managers, and assign client groups to it.

   ```
   user@host# edit access profile Managers
   [edit access profile Managers]
   user@host# set client Client-1 client-group G1
   user@host# set client Client-1 client-group G2
   user@host# set client Client-1 client-group G3
   user@host# set client Client-1 firewall-user password pwd
   ```

2. Configure client groups in the session options.

   ```
   [edit access profile Managers]
   user@host# set session-options client-group G1
   user@host# set session-options client-group G2
   user@host# set session-options client-group G3
   ```

Results

Confirm your configuration by entering the **show access profile Managers** command from configuration mode. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show access profile Managers

client Client-1 {
    client-group [ G1 G2 G3 ];
```

```
    firewall-user {
        password "$ABC123"; ## SECRET-DATA
    }
}
session-options {
    client-group [ G1 G2 G3 ];
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Troubleshooting with Logs on page 81

Troubleshooting with Logs

Purpose Use these logs to identify any issues.

Action From operational mode, enter the **show log messages** command and the **show log dcd** command.

Related Documentation

- Understanding Client Groups for Firewall Authentication on page 79

CHAPTER 8

Customizing the Firewall Authentication Banner

Understanding Firewall Authentication Banner Customization

A banner is a message that appears on a monitor in different places depending on the type of login.

Figure 13: Banner Customization

- At the top of a browser screen after a user has successfully logged into a Web authentication address as shown Figure 13 on page 83.
- Before or after a Telnet, an FTP, an HTTP, or and HTTPS login prompt, success message, and fail message for users

All banners, except for a console login banner, have default messages. You can customize the messages that appear on the banners to better suit the network environment in which you use the device.

Related Documentation

- Firewall User Authentication Overview on page 9
- Example: Customizing a Firewall Authentication Banner on page 83

Example: Customizing a Firewall Authentication Banner

This example shows how to customize the banner text that appears in the browser.

Requirements

Before you begin, create an access profile.

Overview

A banner is a message that appears on a monitor in different places depending on the type of login. This example shows how to change the banner that appears in the browser to indicate that a user has successfully authenticated after successfully logging in through Web authentication. The new message is “Web authentication is successful.” If the authentication fails, then the new message reads “Authentication failed.”

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set access firewall-authentication pass-through default-profile Profile-1
set access firewall-authentication pass-through ftp banner fail “ Authentication failed”
set access firewall-authentication web-authentication default-profile Profile-1
set access firewall-authentication web-authentication banner success “ Web authentication is successful”
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To customize the banner text that appears in the browser:

1. Specify the banner text for failed pass-through authentication through FTP.

   ```
   [edit]

   user@host# set access firewall-authentication pass-through default-profile Profile-1

   user@host# set access firewall-authentication pass-through ftp banner fail “
     Authentication failed”
   ```

2. Specify the banner text for successful Web authentication.

   ```
   [edit]

   user@host# set access web-authentication default-profile Profile-1

   user@host# set access web-authentication banner success “ Web authentication
     is successful”
   ```

Results

From configuration mode, confirm your configuration by entering the **show access firewall-authentication** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show access firewall-authentication
```

```
pass-through {
    default-profile Profile-1;
    ftp {
        banner {
            fail "Authentication failed";
        }
    }
}
web-authentication {
    default-profile Profile-1;
    banner {
        success "Web authentication is successful";
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Related Documentation

- Understanding Firewall Authentication Banner Customization on page 83

PART 3

Configuring Infranet Authentication

CHAPTER 9

Configuring UAC in a Junos OS Environment

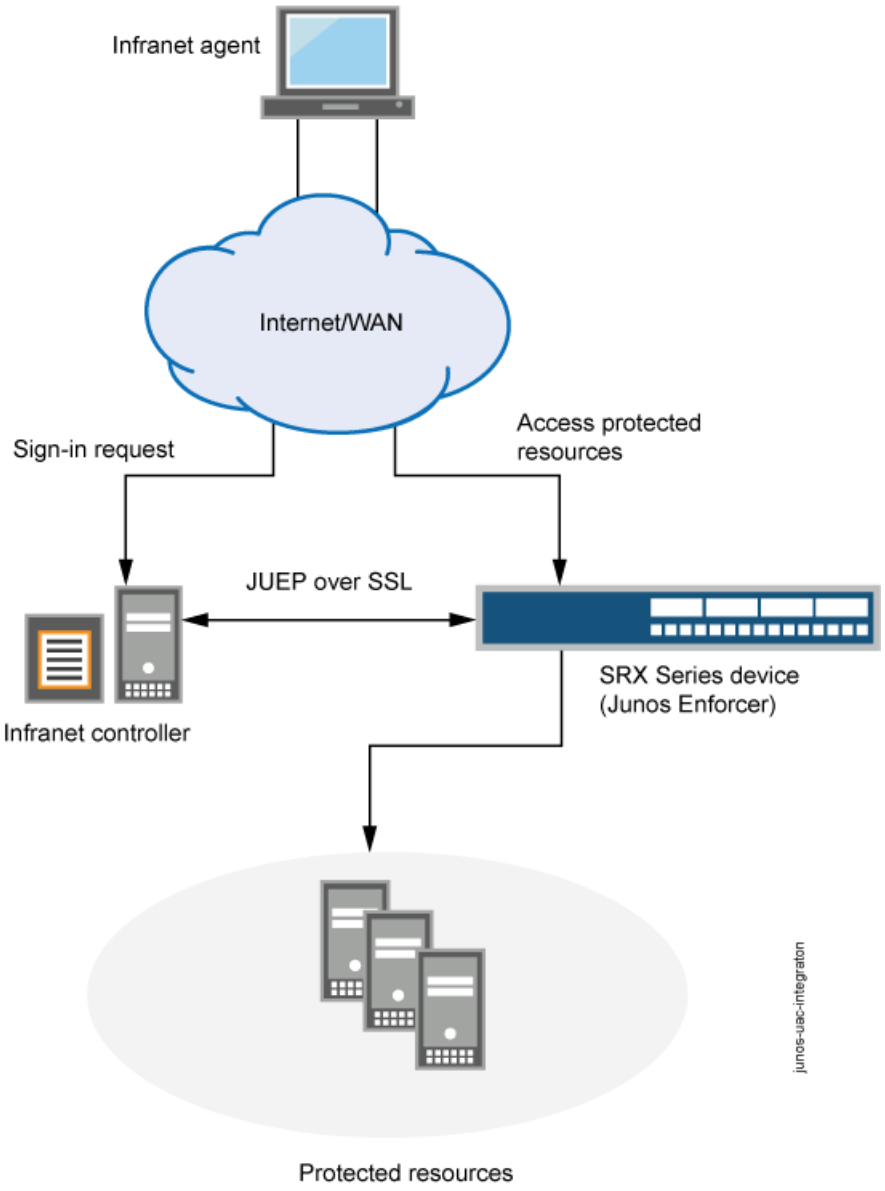
Understanding UAC in a Junos OS Environment

A Unified Access Control (UAC) deployment uses the following components to secure a network and ensure that only qualified end users can access protected resources:

- IC Series UAC Appliances—An IC Series appliance is a policy decision point in the network. It uses authentication information and policy rules to determine whether or not to provide access to individual resources on the network. You can deploy one or more IC Series appliances in your network.
- Infranet Enforcers—An Infranet Enforcer is a policy enforcement point in the network. It receives policies from the IC Series appliance and uses the rules defined in those policies to determine whether or not to allow an endpoint access to a resource. You deploy the Infranet Enforcers in front of the servers and resources that you want to protect.
- Infranet agents—An Infranet agent is a client-side component that runs directly on network endpoints (such as users' computers). The agent checks that the endpoint complies to the security criteria specified in Host Checker policies and relays that compliance information to the Infranet Enforcer. The Infranet Enforcer then allows or denies the endpoint access based on the compliance results.

An SRX Series device can act as an Infranet Enforcer in a UAC network. Specifically, it acts as a Layer 3 enforcement point, controlling access by using IP-based policies pushed down from the IC Series appliance. When deployed in a UAC network, an SRX Series device is called a Junos OS Enforcer. See Figure 14 on page 90.

Figure 14: Integrating a Junos OS Security Device into a Unified Access Control Network



NOTE: You can use the Junos OS Enforcer with the IC Series appliance and Secure Access devices in an IF-MAP Federation network. In a federated network, multiple IC Series appliances and Secure Access devices that are not directly connected to the Junos OS Enforcer can access resources protected by the security device. There are no configuration tasks for IF-MAP Federation on the Junos OS Enforcer. You configure policies on IC Series appliances that can dynamically create authentication table entries on the Junos OS Enforcer.

Related Documentation

- Enabling UAC in a Junos OS Environment (CLI Procedure) on page 91

Enabling UAC in a Junos OS Environment (CLI Procedure)

Junos OS security policies enforce rules for transit traffic, defining what traffic can pass through the Juniper Networks device. The policies control traffic that enters from one zone (from-zone) and exits another (to-zone). To enable an SRX Series device as a Junos OS Enforcer in a UAC deployment, you must:

- Identify the source and destination zones through which UAC traffic will travel. It also needs the list of interfaces, including which zones they are in. The IC Series UAC Appliance uses the destination zone to match its own IPsec routing policies configured on IC Series appliance.
- Identify Junos OS security policies that encompass those zones, and enable UAC for those policies.

Before you begin:

1. Set up the interfaces through which UAC traffic should enter the SRX Series device.
2. Group interfaces with identical security requirements into zones. See *Example: Creating Security Zones*.
3. Create security policies to control the traffic that passes through the security zones. See *Example: Configuring a Security Policy to Permit or Deny All Traffic*.

To configure UAC through a Junos OS security policy, enter the following configuration statement:

user@host# **set security policies from-zone *zone-name* to-zone *zone-name* policy match then permit application-services uac-policy**

CHAPTER 10

Establishing Communications Between Devices

Understanding Communications Between the Junos OS Enforcer and the IC Series UAC Appliance

When you configure an SRX Series device to connect to an IC Series UAC Appliance, the SRX Series device and the IC Series appliance establish secure communications as follows:

1. If more than one IC Series device are configured as Infranet Controllers on the SRX Series device, a round-robin algorithm determines which of the configured IC Series devices is the active Infranet Controller. The others are failover devices. If the active Infranet Controller becomes inoperative, the algorithm is reapplied to the remaining IC Series devices that are configured to establish the new active Infranet Controller.
2. The active IC Series appliance presents its server certificate to the SRX Series device. If configured to do so, the SRX Series device verifies the certificate. (Server certificate verification is not required; however, as an extra security measure you can verify the certificate to implement an additional layer of trust.)
3. The SRX Series device and the IC Series appliance perform mutual authentication using the proprietary challenge-response authentication. For security reasons, the password is not included in the message sent to the IC Series appliance.
4. After successfully authenticating with the SRX Series device, the IC Series appliance sends its user authentication and resource access policy information. The SRX Series device uses this information to act as the Junos OS Enforcer in the UAC network.
5. Thereafter, the IC Series appliance and the Junos OS Enforcer can communicate freely with one another over the SSL connection. The communications are controlled by a proprietary protocol called *Junos UAC Enforcer Protocol (JUEP)*.

Related Documentation

- Understanding UAC in a Junos OS Environment on page 89
- Configuring Communications Between the Junos OS Enforcer and the IC Series UAC Appliance (CLI Procedure) on page 94

Understanding Communications Between Junos OS Enforcer and a Cluster of IC Series UAC Appliances

You can configure a Junos OS Enforcer to work with more than one IC Series UAC Appliance in a high availability configuration known as an IC Series appliance cluster. The Junos OS Enforcer communicates with only one IC Series appliance at a time; the other IC Series appliances are used for failover. If the Junos OS Enforcer cannot connect to the first IC Series appliance you added to a cluster, it tries to connect to the failed IC Series appliance again. Then it fails over to the other IC Series appliances in the cluster. It continues trying to connect to IC Series appliances in the cluster until a connection occurs.

When the Junos OS Enforcer cannot establish a connection to an Infranet Enforcer, it preserves all its existing authentication table entries and Unified Access Control (UAC) policies and takes the timeout action that you specify. Timeout actions include:

- **close**—Close existing sessions and block any further traffic. This is the default option.
- **no-change**—Preserve existing sessions and require authentication for new sessions.
- **open**—Preserve existing sessions and allow new sessions access.

Once the Junos OS Enforcer can reestablish a connection to an IC Series appliance, the IC Series appliance compares the authentication table entries and UAC policies stored on the Junos OS Enforcer with the authentication table entries and policies stored on the IC Series appliance and reconciles the two as required.

NOTE: The IC Series appliances configured on a Junos OS Enforcer should all be members of the same IC Series appliance cluster.

Configuring Communications Between the Junos OS Enforcer and the IC Series UAC Appliance (CLI Procedure)

To configure an SRX Series device to act as a Junos OS Enforcer in a UAC deployment, and therefore to enforce IC Series UAC Appliance policies, you must specify an IC Series appliance to which the SRX Series device should connect.

Before you begin:

1. Enable UAC through the relevant Junos OS security policies. See "Enabling UAC in a Junos OS Environment (CLI Procedure)" on page 91.
2. (Optional) Create a profile for the certificate authority (CA) that signed the IC Series appliance's server certificate, and import the CA certificate onto the SRX Series device. See *Example: Loading CA and Local Certificates Manually*.
3. Configure user authentication and authorization by setting up user roles, authentication and authorization servers, and authentication realms on the IC Series appliance.
4. Configure resource access policies on the IC Series appliance to specify which endpoints are allowed or denied access to protected resources.

To configure an SRX Series device to act as a Junos OS Enforcer:

1. Specify the IC Series appliance(s) to which the SRX Series device should connect.
   - To specify the IC Series appliance hostname:

     user@host# **set services unified-access-control infranet-controller *hostname***
   - To specify the IC Series appliance IP address:

     user@host# **set services unified-access-control infranet-controller *hostname* address *ip-address***

   NOTE: When configuring access to multiple IC Series appliances, you must define each separately. For example:

   ```
   user@host# set services unified-access-control infranet-controller IC1
   user@host# set services unified-access-control infranet-controller IC2
   user@host# set services unified-access-control infranet-controller IC3

   user@host# set services unified-access-control infranet-controller IC1
     address 10.10.10.1
   user@host# set services unified-access-control infranet-controller IC2
     address 10.10.10.2
   user@host# set services unified-access-control infranet-controller IC3
     address 10.10.10.3
   ```

   Make sure that all of the IC Series appliances are members of the same cluster.

   NOTE: By default, the IC Series appliance should select port 11123.
2. Specify the Junos OS interface to which the IC Series appliance should connect:

   user@host# **set services unified-access-control infranet-controller *hostname* interface *interface-name***

3. Specify the password that the SRX Series device should use to initiate secure communications with the IC Series appliance:

NOTE: Any change in the Unified Access Control's (UAC) contact interval and timeout values in the SRX Series device will be effective only after the next reconnection of the SRX Series device with the IC Series appliance.

user@host# **set services unified-access-control infranet-controller** *hostname* **password** *password*

4. (Optional) Specify information about the IC Series appliance's server certificate that the SRX Series device needs to verify the certificate.

   - To specify the server certificate subject that the SRX Series device checks:

     user@host# **set services unified-access-control infranet-controller** *hostname* **server-certificate-subject** *certificate-name*

   - To specify the CA profile associated with the certificate:

     user@host# **set services unified-access-control infranet-controller** *hostname* **ca-profile** *ca-profile*

NOTE: An IC Series appliance server certificate can be issued by an intermediate CA. There are two types of CAs—root CAs and intermediate CAs. An intermediate CA is secondary to a root CA and issues certificates to other CAs in the public key infrastructure (PKI) hierarchy. Therefore, if a certificate is issued by an intermediate CA, you need to specify the complete list of CA profiles in the certification chain.

CHAPTER 11

Configuring Policy Enforcement

Understanding Junos OS Enforcer Policy Enforcement

Once the SRX Series device has successfully established itself as the Junos OS Enforcer, it secures traffic as follows:

1. First, the Junos OS Enforcer uses the appropriate Junos OS security policy to process the traffic. A *security policy* uses criteria such as the traffic's source IP address or the time of day that the traffic was received to determine whether or not the traffic should be allowed to pass.
2. Once it determines that the traffic may pass based on the Junos OS security policy, the Junos OS Enforcer maps the traffic flow to an authentication table entry. The Junos OS Enforcer uses the source IP address of the first packet in the flow to create the mapping.

   An *authentication table entry* contains the source IP address and user role(s) of a user who has already successfully established a UAC session. A *user role* identifies a group of users based on criteria such as type (for instance, "Engineering" or "Marketing") or status (for instance, "Antivirus Running"). The Junos OS Enforcer determines whether to allow or deny the traffic to pass based on the authentication results stored in the appropriate authentication table entry.

   The IC Series UAC Appliance pushes authentication table entries to the Junos OS Enforcer when the devices first connect to one another and, as necessary, throughout the session. For example, the IC Series appliance might push updated authentication table entries to the Junos OS Enforcer when the user's computer becomes noncompliant with endpoint security policies, when you change the configuration of a user's role, or when you disable all user accounts on the IC Series appliance in response to a security problem such as a virus on the network.

   If the Junos OS Enforcer drops a packet because of a missing authentication table entry, the device sends a message to the IC Series appliance, which in turn may

provision a new authentication table entry and send it to the Junos OS Enforcer. This process is called dynamic authentication table provisioning.

3. Once it determines that the traffic may pass based on the authentication table entries, the Junos OS Enforcer maps the flow to a resource. The Junos OS Enforcer uses the destination IP address specified in the flow to create the mapping. Then the device uses that resource as well as the user role specified in the authentication table entry to map the flow to a resource access policy.

   A *resource access policy* specifies a particular resource to which you want to control access based on user role. For instance, you might create a resource access policy that allows only users who are members of the Engineering and Antivirus Running user roles access to the Engineering-Only server. Or you might create a resource access policy that allows members of the No Antivirus Running user role access to the Remediation server on which antivirus software is available for download.

   The IC Series appliance pushes resource access policies to the Junos OS Enforcer when the devices first connect to one another and when you modify your resource access policy configurations on the IC Series appliance.

   If the Junos OS Enforcer drops the packet because of a “deny” policy, the Junos OS Enforcer sends a message to the IC Series appliance, which in turn sends a message to the endpoint’s Odyssey Access Client (if available). (The IC Series appliance does not send “deny” messages to the agentless client.)

4. Once it determines that the traffic may pass based on the resource access policies, the Junos OS Enforcer processes the traffic using the remaining application services defined in the Junos OS policy. The Junos OS Enforcer runs the remaining services in the following order: Intrusion Detection and Prevention (IDP), URL filtering, and Application Layer Gateways (ALGs).

Configuring Junos OS Enforcer Failover Options (CLI Procedure)

To configure IC Series UAC Appliance failover processing, you must configure the Junos OS Enforcer to connect to a cluster of IC Series appliances. The Junos OS Enforcer communicates with one of these IC Series appliances at a time and uses the others for failover processing.

Before you begin:

1. Enable UAC through the relevant Junos OS security policies.
2. Configure the SRX Series device as a Junos OS Enforcer. During the configuration, define a cluster of IC Series appliances to which the Junos OS Enforcer should connect. See “Enabling UAC in a Junos OS Environment (CLI Procedure)” on page 91.

To configure failover processing:

1. Specify how often (in seconds) the Junos OS Enforcer should expect a heartbeat signal from the IC Series appliance indicating an active connection:

   user@host# **set services unified-access-control interval *seconds***

2. Specify the interval (in seconds) at which the Junos OS Enforcer should consider the current connection timed out:

NOTE: Any change in the Unified Access Control's (UAC) contact interval and timeout values in the SRX Series device will be effective only after the next reconnection of the SRX Series device with the IC Series appliance.

```
user@host# set services unified-access-control timeout seconds
```

3. Specify how the Junos OS Enforcer should handle all current and subsequent traffic sessions when its connection to an IC Series appliance cluster times out:

```
user@host# set services unified-access-control timeout-action (close | no-change |
  open)
```

Related Documentation

- Understanding Junos OS Enforcer Policy Enforcement on page 97
- Testing Junos OS Enforcer Policy Access Decisions Using Test-Only Mode (CLI Procedure) on page 99
- Verifying Junos OS Enforcer Policy Enforcement on page 100

Testing Junos OS Enforcer Policy Access Decisions Using Test-Only Mode (CLI Procedure)

When configured in test-only mode, the SRX Series device enables all UAC traffic to go through regardless of the UAC policy settings. The device logs the UAC policy's access decisions without enforcing them so you can test the implementation without impeding traffic.

Before you begin:

1. Enable UAC through the relevant Junos OS security policies. See "Enabling UAC in a Junos OS Environment (CLI Procedure)" on page 91
2. Configure the SRX Series devices as a Junos OS Enforcer. See "Configuring Communications Between the Junos OS Enforcer and the IC Series UAC Appliance (CLI Procedure)" on page 94.
3. If you are connecting to a cluster of IC Series UAC Appliances, enable failover options. See "Configuring Junos OS Enforcer Failover Options (CLI Procedure)" on page 98.

To activate or deactivate test-only mode, enter the following configuration statement:

```
user@host#  set services unified-access-control test-only-mode (true | false)
```

Verifying Junos OS Enforcer Policy Enforcement

Displaying IC Series UAC Appliance Authentication Table Entries from the Junos OS Enforcer

Purpose Display a summary of the authentication table entries configured from the IC Series UAC Appliance.

Action Enter the **show services unified-access-control authentication-table** CLI command.

Displaying IC Series UAC Appliance Resource Access Policies from the Junos OS Enforcer

Purpose Display a summary of UAC resource access policies configured from the IC Series UAC Appliance.

Action Enter the **show services unified-access-control policies** CLI command.

CHAPTER 12

Classifying Traffic with User Roles

Understanding Unified Access Control

In Junos OS Release 12.1 and later, user role firewall security policies let you classify traffic based on the roles to which a user is assigned. Based on match criteria, which includes the user's role, you create policies to apply services that allow or block access to resources. The user role firewall is similar to the identity-based network access control (NAC) solution available with UAC on the SRX Series device. A user role firewall, however, does not require the Junos Pulse/Odyssey installation, and it supports agentless transparent authentication.

User role information can be collected in several ways: locally on the SRX Series device, from a Junos Pulse Access Control Service device, or by relaying authentication data from a third-party authentication server through a Junos Pulse Access Control Service device to the SRX Series device.

Related Documentation

- Acquiring User Role Information from an Active Directory Authentication Server on page 101

Acquiring User Role Information from an Active Directory Authentication Server

Networks have used the IP address as a way of identifying users and servers. The strategy is based on the assumption that users or groups of users connect to the network from fixed locations and use one device at a time.

Wireless networking and mobile devices require a different strategy. Individuals can connect to the network using multiple devices simultaneously. The way in which devices connect to the network changes rapidly. It is no longer possible to identify a user with a group of statically allocated IP addresses.

In Junos OS Release 12.1 and later, user role firewall security policies let you classify traffic based on the roles to which a user is assigned. Based on match criteria, which includes the user's role, you create policies to apply services that allow or block access to resources. The user role firewall is similar to the identity-based network access control (NAC)

solution available with UAC on the SRX Series device. A user role firewall, however, does not require the Junos Pulse/Odyssey installation, and it supports agentless transparent authentication.

User role information can be collected in several ways: locally on the SRX Series device, from a Junos Pulse Access Control Service device, or by relaying authentication data from a third-party authentication server through a Junos Pulse Access Control Service device to the SRX Series device.

Incorporating a third-party authentication server into a user role firewall configuration can also provide single sign-on (SSO) support. This allows a browser-based user to authenticate once and have that authentication communicated to other trusted servers in the domain as needed.

Requirements

This solution uses the following hardware and software components:

- One MAG Series Junos Pulse Gateway device with software release 4.2 or later
- The MAGx600-UAC-SRX license installed on the MAG Series device
- One SRX Series device with Junos OS Release 12.1 or later
- One Microsoft Active Directory server using version 2008

NOTE: **Microsoft Windows 2003 is also compatible with this functionality, but terminology, pathways, and settings might differ from what is presented in this document.**

Before you begin:

- Ensure that the MAG Series device is configured as an Access Control Service and is accessible to the network. See the *MAG Series Junos Pulse Gateway Hardware Guide* for configuration details.
- Ensure that the MAGx600-UAC-SRX license is installed on the MAG Series device.
- Ensure that the SRX Series device is configured and initialized with Junos OS version 12.1 or later.
- Ensure that the Active Directory authentication server is configured for standard Junos Pulse Access Control Service authentication. See your third-party documentation.
- Ensure that the administrator has the appropriate capabilities for configuring the roles, users, and device interactions.

Overview

In this solution an SRX Series device obtains user role information dynamically from a Microsoft Active Directory authentication server. Authentication verification and user role information from the Active Directory server is relayed by the Access Control Service on the MAG Series device to the SRX Series device.

Users within the same domain are connected to a LAN segment. They are associated with user role groups, such as developer or manager, depending on their work in the organization. When a user authenticates to the AD authentication server, the user should be able to access protected resources without having to authenticate a second time.

The SRX Series device is configured as an enforcer for the MAG Series device. It receives user role information from the MAG Series device and applies user role firewall policies accordingly to incoming and outgoing traffic.

When the SRX Series device has no user role information for a user, the user's browser is redirected to the MAG Series device. Transparently to the user, the MAG Series device requests verification from the browser. The browser retrieves a token from the Active Directory server confirming authentication and passes it to the MAG Series device. With the information provided by the token, the MAG Series device retrieves user role information for the user from the Active Directory server and creates an authentication table entry consisting of the current IP address and the user role data. The MAG Series device pushes the updated table to the SRX Series device and redirects the browser back to the SRX to request access again. This time, the table does contain user role information which is then retrieved and used as part of the match criteria for applying user role firewall services.

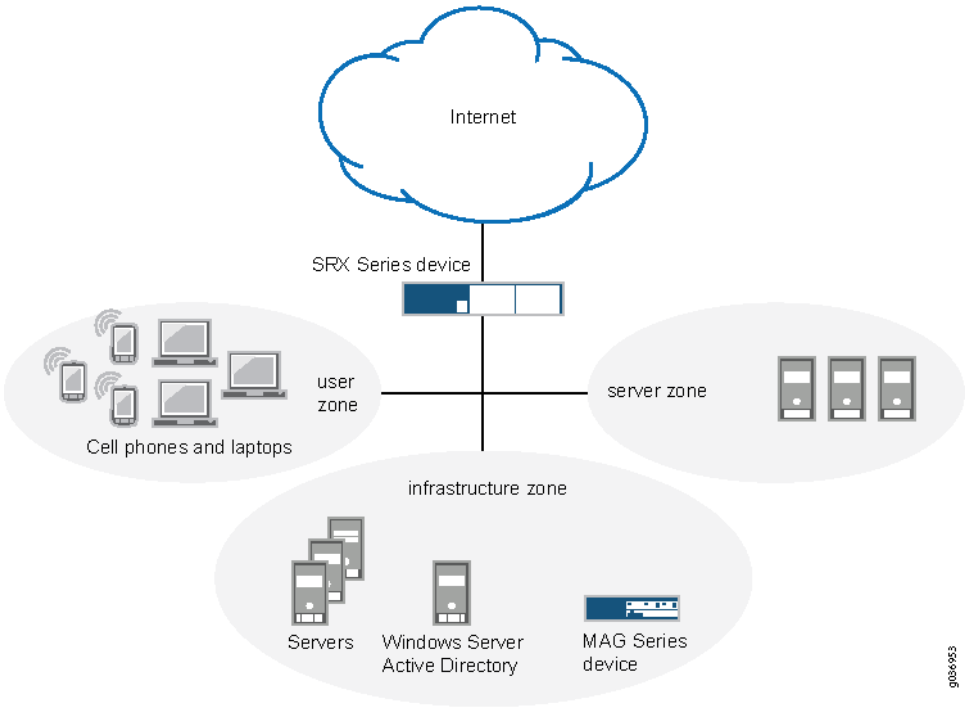
The user is not aware of the process unless the Active Directory (AD) server has no current authentication for the user. When that is the case, the server prompts the user for name and password. Once authentication occurs, the server returns a token to the browser.

The procedure documented here initially configures the MAG Series device as the authenticator. The configuration is later modified to retrieve authentication information from the AD server. This solution uses SPNEGO negotiation and Kerberos authentication to secure communications among the SRX Series device, the MAG Series device, the browser, and the authentication server.

Topology

Figure 15 on page 104 shows the topology for this deployment in which the MAG Series device is used initially as the authenticationn source. Later, the AD server is used transparently unless the user is not authenticated, in which case he is prompted for a user name and password.

Figure 15: Single Sign-On Support Topology



A user's request to access another resource is controlled by roles and groups associated with the user. For example, a user belonging to a group of developers named Dev might have access to a particular test server. The same user might also be the manager and belong to the Mgr group that can access certain HR resources. A contractor working for this manager might require access to the test server as well but not to the HR resources. In this case, the user would be added to the Dev group and perhaps a Contractor group, but not the Mgr group.

User role firewall policies defined on the SRX Series device control the groups and user roles that can access various resources. In this configuration, if user role data does not exist for a user requesting access, a policy redirects the user's browser to the MAG Series device to authenticate the user and retrieve any associated user role data.

A token exchange among the Access Control Service, the browser, and the Active Directory server remains transparent to the user while it verifies the user's authentication. The exchange uses SPNEGO negotiation and Kerberos authentication for encrypting and decrypting messages among the devices.

With information obtained from the response token, the MAG Series device retrieves the user's roles and groups directly from the Active Directory server. It then creates an authentication table entry and passes it to the SRX Series device.

Configuration

Configure the devices for this solution by performing the following tasks.

- Connect the SRX Series device and the MAG Series device in an enforcer configuration.
- Configure the Access Control Service on the MAG Series device for local user authentication and verify that authentication information is transferred between the devices.
- Configure a captive portal policy on the SRX Series device to redirect any unauthenticated user to the Access Control Service and verify that redirection is functioning properly.
- Configure the Microsoft Active Directory authentication server to interact with the Access Control Service and the endpoints.
- Reconfigure the Access Control Service for remote authentication by the Active Directory server and redefine Active Directory groups for the SRX Series device.
- Configure endpoint browsers for the SPNEGO protocol

NOTE: **Configuring the Access Control Service using local authentication is not necessary for this solution. However, by configuring local authentication first you can verify the captive portal interaction between the MAG Series device and the SRX Series device.**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

Connecting the SRX Series Device to the Access Control Service

Step-by-Step Procedure

In an enforcer configuration, the Access Control Service on the MAG Series device and the SRX Series device communicate over a secure channel. When the SRX Series device first connects with the Access Control Service, the devices exchange information to ensure secure communication. Optionally, you can use digital security certificates as an enhanced mechanism for establishing trust.

See the *Unified Access Control Administration Guide* for details about configuring certificate trust between the SRX Series device and the Access Control Service.

To connect the SRX Series device and the Access Control Service on the MAG Series device:

1. Configure the SRX Series device.

   a. Configure the zones and interfaces of the devices.

      ```
      user@host# set security zones security-zone user interfaces ge-0/0/0
      user@host# set security zones security-zone infrastructure interfaces ge-0/0/1
      user@host# set security zones security-zone untrust interfaces ge-0/0/2
      ```

   b. Configure the IP addresses of the interfaces.

      ```
      user@host# set interfaces ge-0/0/0 unit 0 family inet address 10.12.12.1/8
      user@host# set interfaces ge-0/0/1 unit 0 family inet address 10.0.0.22/24
      user@host# set interfaces ge-0/0/2 unit 0 family inet address 203.0.113.19/24
      ```

c. Identify the Access Control Service as a new Infranet Controller, and configure the interface for the connection to it.

```
[edit]
user@host# set services unified-access-control infranet-controller mag123
  address 10.0.0.22
user@host# set services unified-access-control infranet-controller mag123
  interface fxp0.0
```

d. Specify the password for securing interactions between the Access Control Service and the SRX Series device.

```
[edit]
user@host# set services unified-access-control infranet-controller mag123
  password pwd
```

NOTE: The same password must be configured on both devices.

e. (Optional) Specify the full name of the Access Control Service certificate that the SRX Series device must match during connection.

```
user@host# set services unified-access-control infranet-controller mag123
  ca-profile ca-mag123-enforcer
```

f. If you are done configuring the SRX Series device, enter commit from configuration mode.

2. Configure the Access Control Service from the administrator console on the MAG Series device.

   a. Navigate to the Infranet Enforcer page, and click **New Enforcer**.

   b. Select Junos, enter the password set previously on the SRX Series device (InSub321), and enter the serial number of the SRX Series device.

   c. Click **Save Changes**.

Results When both devices are configured, the SRX Series device connects automatically to the Access Control Service.

- From the Access Control Service, select **System>Status>Overview** to view the status of the connection to the SRX Series device. The diode in the display is green if the connection is functioning. To display additional information, click the device name.
- From operational mode on the SRX Series device, confirm your connection by entering the **show services unified-access-control status** command. If the output does not display the intended configuration, repeat the instructions in this section to correct the configuration.

```
user@host> show services unified-access-control status

Host           Address         Port   Interface     State
       mag123        10.0.0.22        11123  fxp0.0        connected
```

Configuring the Access Control Service for Local User Authentication

Step-by-Step Procedure

When a user is authenticated, the Access Control Service on the MAG Series device updates its authentication table with the IP address and associated roles of the user, and pushes the updated table to the SRX Series device. If this user data is deleted or modified, the Access Control Service updates the authentication table with the new information and again pushes it to the SRX Series device.

To test the successful transfer and content of the authentication table, this task configures the Access Control Service on the MAG Series device for local authentication. Within this configuration you can test the user role firewall from the SRX Series device without affecting other network operations. A later task modifies this configuration to provide user role retrieval from the remote Active Directory server.

NOTE: It is not a requirement to configure the Access Control Service for local user authentication. It is provided so that you can test each task in the configuration.

To configure the Access Control Service for local authentication:

1. Define roles on the Access Control Service.
   a. From the administrator console of the Access Control Service, select **Users>User Roles>New User Role**.
   b. Enter **dev** as the role name.

      In this solution, use the default values for other role settings.
   c. Click **Save Changes**.

NOTE: This solution assumes that the MAGx600-UAC-SRX license is installed on the Access Control Service. If the full-feature license is installed, you will need to disable OAC Install and enable Agentless Access.

2. Configure the default authentication server.
   a. Select **Authentication>Auth. Servers**.
   b. Select **System Local**. This establishes the MAG Series device as the default authentication server.
3. Create users.
   a. Select the **Users** tab, and click **New**.
   b. Create **user-a** by entering the following details.

- Username
- User's full name
- Password
- Password confirmation

c. Repeat the previous step to create **user-b**.

d. Click **Save Changes**.

4. Create a realm.

a. Select **Users>User Realms>New User Realm**.

b. Enter **REALM6** as the realm name.

c. Select **System Local** in the Authentication box.

d. Click **Save Changes**.

5. From the same page, create role mapping rules.

a. Select the **Role Mapping** tab, and click **New Rule**.

b. Define two rules with the following details.

- Enter username user-a, and assign it to role dev.
- Enter username user-b, and assign it to role dev.

c. Click **Save Changes**.

6. Set up the default sign-in page.

a. Select **Authentication>Signing In>Sign-in Policies**.

b. Click the default **Sign-in policy (*/)**.

c. In the **Sign-in URL** box, enter the IP address of this device.

d. In **Authentication realm, Available realms**, select REALM6.

e. Click **Save Changes**.

Results Verify the results of the configuration. If the output does not display the intended configuration, repeat the instructions in this section to correct the configuration.

1. Verify that local authentication on the Access Control Service is functioning properly.

- Open a browser window from an endpoint in the network.
- Enter the fully qualified domain name for the Access Control Service.

  The default sign-in page should display.
- Sign in as user-a, and provide the defined password.

2. From operational mode on the SRX Series device:

   a. Confirm that the authentication table on the SRX Series device was updated with **user-a**.

      user@host> **show services unified-access-control authentication-table**

      ```
      Id    Source IP       Username     Age      Role identifier
      1     203.0.113.102    user-a       0        0000000001.000005.0
      Total: 1
      ```

   b. Confirm that the correct role has been associated with the role identifier.

      user@host> **show services unified-access-control roles**

      ```
      Name                Identifier

      dev                 0000000001.000005.0
      ```

   c. List all roles associated with user-a.

      user@host> **show services unified-access-control authentication-table detail**

      ```
      Identifier: 1
        Source IP: 203.0.113.102
        Username: user-a
        Age: 0
        Role identifier       Role name
          0000000001.000005.0 dev
      ```

Configuring Redirection from the SRX Series Device to the Access Control Service

Step-by-Step Procedure

Local authentication, as configured in the previous task, requires users to log on to the Access Control Service directly to gain access to network resources. The SRX Series device can be configured to automatically redirect the browser of an unauthenticated user to the Access Control Service if a user requests access to a protected resource directly. You can define a user role firewall policy to redirect an unauthenticated user to a captive portal on the Access Control Service for sign-in.

NOTE: Other services, such as IDP, UTM, AppFW, and AppQoS, can be configured as well as the UAC captive portal implementation. The solution focuses on captive portal for authentication for user role implementation only.

To configure redirection from the SRX Series device to the Access Control Service:

1. From configuration mode on the SRX Series device, configure the profile for the captive portal acs-device.

   ```
   [edit]
   user@host# set services unified-access-control captive-portal acs-device
     redirect-traffic unauthenticated
   ```

2. Add either the redirection URL for the Access Control Service or a default URL.

```
[edit]
user@host# set services unified-access-control captive-portal acs-device
redirect-url “https://%ic-url%/?target=%dest-url%&enforcer=%enforcer-id%”
```

   This command specifies the default target and enforcer variables so that the browser is returned to the SRX Series device after authentication.

3. Allow traffic to the Acitve Directory (AD) server, the Access Control Service, and the other infrastructure servers.

```
[edit]
user@host# set security policies from-zone user to-zone infrastructure policy
  Allow-AD-UAC match source-address any
user@host# set security policies from-zone user to-zone infrastructure policy
  Allow-AD-UAC match destination-address any
user@host# set security policies from-zone user to-zone infrastructure policy
  Allow-AD-UAC application any
user@host# set security policies from-zone user to-zone infrastructure policy
  Allow-AD-UAC then permit
```

4. Configure a security policy that redirects HTTP traffic from zone user to zone untrust if the source-identity is unauthenticated-user.

```
[edit]
user@host# set security policies from-zone user to-zone untrust policy user-role-fw1
  match source-address any
user@host# set security policies from-zone user to-zone untrust policy user-role-fw1
  match destination-address any
user@host# set security policies from-zone user to-zone untrust policy user-role-fw1
  match application http
user@host# set security policies from-zone user to-zone untrust policy user-role-fw1
  match source-identity unauthenticated-user
```

5. Configure the action to be taken when traffic matches the criteria for user-role-fw1.

   In this case, traffic meeting the specified criteria is allowed access to the UAC captive portal defined by the acs-device profile.

```
user@host# set security policies from-zone user to-zone untrust policy user-role-fw1
  then permit application-services uac-policy captive-portal acs-device
```

6. Configure a security policy allowing access to any HTTP traffic from zone user to zone untrust.

```
[edit]
user@host# set security policies from-zone user to-zone untrust policy user-role-fw2
  match source-address any
user@host# set security policies from-zone user to-zone untrust policy user-role-fw2
  match destination-address any
user@host# set security policies from-zone user to-zone untrust policy user-role-fw2
  match application http
user@host# set security policies from-zone user to-zone untrust policy user-role-fw2
  match source-identity any
user@host# set security policies from-zone user to-zone untrust policy user-role-fw2
  then permit
```

NOTE: It is important to position the redirection policy for unauthenticated users before a policy for “any” user so that UAC authentication is not shadowed by a policy intended for authenticated users.

7. If you are done configuring the policies, commit the changes.

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

Results Confirm your configuration with the following procedures. If the output does not display the intended configuration, repeat the instructions in this section to correct the configuration.

1. From configuration mode, confirm your captive portal profile configuration by entering the **show services** command.

```
[edit]
user@host# show services

...
unified-access-control {
 captive-portal acs-device {
  redirect-traffic unauthenticated;
  redirect-url “https://%ic-url%/?target=%dest-url%&enforcer=%enforcer-id%”
...
```

2. From configuration mode, confirm your policy configuration by entering the **show security policies** command.

```
user@host# show security policies

...
from-zone user to-zone infrastructure {
 policy Allow-AD-UAC  {
  match {
   source-address any;
   destination-address any;
   application any;
  }
  then {
   permit
  }
 }
}
from-zone user to-zone untrust {
 policy user-role-fw1 {
  match {
   source-address any;
   destination-address any;
   application http;
   source-identity unauthenticated-user
  }
  then {
   permit {
    application-services {
```

```
      uac-policy {
       captive-portal acs-device;
      }
     }
    }
   }
  }
 }
}
from-zone user to-zone untrust {
 policy user-role-fw2 {
  match {
   source-address any;
   destination-address any;
   application http;
   source-identity any
  }
  then {
   permit
  }
 }
}
...
```

3. Verify that the redirection policy is functioning correctly.

   a. Open a browser window from a second endpoint in the network.

   b. Enter a third-party URL, such as www.google.com.

      The default sign-in page from the Access Control Service prompts for a user and password.

   c. Enter the username **user-b** and its password.

      The browser should display the requested URL.

      NOTE: If a pop-up blocker is set on the endpoint, it could interfere with this functionality.

   d. From operational mode on the SRX Series device, verify that the authentication data and roles from the Access Control Service were pushed to the SRX Series device successfully.

      user@host> **show services unified-access-control authentication-table**

```
Id    Source IP       Username     Age      Role identifier
1     203.0.113.112    user-a       0        0000000001.000005.0
2     203.0.113.15    user-b       0        0000000001.000005.0
Total: 2
```

Configuring Active Directory Settings

Step-by-Step Procedure

SPNEGO negotiation and Kerberos authentication are transparent to the user and network administrator, but certain configuration options enable the use of these protocols. This section identifies configuration requirements when using Active Directory as the authentication server. To interact in SPNEGO negotiation, the Access Control Service requires a keytab file created by Active Directory. Refer to your third-party documentation for more information about enabling SPNEGO and Kerberos usage.

This section is not intended to be a tutorial for Active Directory. However, there are specific configuration details required for this solution. See your third-party documentation to set up Active Directory as a domain controller.

To configure the Active Directory authentication server:

1. Add a DNS entry as the UAC service account in the **Forward Lookup Zones**. In this way clients can refer to the MAG Series device by name or by IP address.

   This UAC service account name will be used in the next section when reconfiguring the UAC service on the MAG Series device.

2. Single sign-on authentication requires that the UAC service account password never expires. To modify user settings:

   a. From the Active Directory Users and Computers application in DNS, select **Users>New>User** and select the UAC service account created in step 1.

   b. Select the **Account** tab.

   c. In user settings, click **Password Never Expires**.

3. On the Domain Controller, open a command line, and enter the **ktpass** command to create the SPNEGO keytab file.

   The keytab file created on the Active Directory server contains the full service principal name (SPN) and other encryption information from the server. The keytab file is then uploaded to the Access Control Service on the MAG Series device. This shared information identifies one device to the other whenever encrypted messages and responses are sent.

   Use the following syntax.

   ktpass -out *output-file-name* -mapuser *uac-service-acccount-name* -prin *service://fqdn@REALM*

   ktpass—Third-party Kerberos utility that maps an SPN to a user, in this case, to the UAC service account. The executable is available for download. Refer to your third-party documentation for the source for this utility.

   -out ***output-file-name***—The name for the SPNEGO keytab file you are creating.

   -mapuser ***uac-service-acccount-name***—The name of the UAC service account created in step 1.

-prin *service://fqdn*@*REALM*—The service principal name. The Kerberos authentication uses the SPN in its communication. It does not use an IP address.

service—The HTTP service.

fqdn—The hostname of the Junos Pulse Access Control Service. The *service://FQDN* portion of the name is provided by the Access Control Service when registering with the Active Directory server.

REALM—The realm of the Active Directory authentication server. It is the same as the domain name. The Kerberos realm name is always in uppercase letters following the recommendation in RFC 1510. This affects interoperability with other Kerberos-based environments.

The following command creates an SPNEGO keytab file named ic.ktpass.

```
ktpass -out ic.ktpass -mapuser icuser@UCDC.COM -princ
  HTTP/mag123.ucdc.com@UCDC.COM -pass Doj73096
```

This file is copied to the Access Control Service on the MAG Series device in the next section when SPNEGO is configured for remote authentication.

Reconfiguring Remote Authentication on the Access Control Service

Step-by-Step Procedure

This section reconfigures the Access Control Service on the MAG Series device to query the remote Active Directory server instead of the local authentication table when authenticating a user. The following steps add services and authentication options to the Access Control Service on the MAG Series device. The configuration of the SRX Series device remains unchanged.

When you reconfigure the realm's authentication server, the Access Control Service displays all roles or groups from the configured domain controller and its trusted domains. Establishing role mapping rules equates the authentication server's roles or groups to those defined on the Access Control Service.

To reconfigure remote authentication on the Access Control Service:

1. From the administrator console of the Access Control Service on the MAG Series device, select **Authentication>Auth. Servers**.
2. Choose the **Active Directory/Windows NT** server type, and click **Add New Server**.
3. Enter the profile of the new authentication server.
   a. Name the Active Directory server.
   b. Enter its NetBIOS domain name in the domain box.

NOTE: You might receive the following message: "Either the server is not a domain controller of the domain, or the NetBIOS name of the domain is different from the Active Directory (LDAP) name." This message is informational and does not affect the processing of the authentication.

c. Enter the Kerberos Realm name.

   The Kerberos realm name is the FQDN of the Active Directory domain. For example, if “mycompany” is the domain or NetBIOS name, mycompany.com is the Kerberos realm name.

d. In the Domain Join Configuration section, enter the username and password of the UAC services account which has permission to join computers to the Active Directory domain.

   Select the Save credentials box.

e. Enter the Container name.

   This is the name of the container in Active Directory where you created the UAC services account for the Access Control Service.

f. Enter the Computer Name.

   Specify the machine ID that the Access Control Service uses to join the specified Active Directory domain as a computer. This name is derived from the licence hardware ID of the Access Control Service in the following format: 0161MT2L00K2C0.

g. Verify that the join operation has succeeded.

   The Join Status indicator provides a color-coded status for the domain join operation as follows:

   - Gray: Not started
   - Yellow: In progress
   - Red: Failed to join
   - Green: Joined the domain

h. Select **Kerberos** and **NTLM v2** as the authentication protocols.

i. In the Trusts section, select the Allow trusted domains box.

j. Select **Enable SPNEGO**.

k. Use the Browse button to upload the keytab file that you created in the previous section.

l. Click **Save Changes** and **Test Configuration**.

4. Ensure that SSO is enabled.

   a. Select **Users>User Realms** and the realm name.

   b. Select the Active Directory server name from the **Auth Server** list.

   c. Select the **Authentication Policy** tab.

d. Verify that the **SSO** option is selected.

e. Click **Save Changes**.

5. Create role-mapping policies for groups acquired from the authentication server.

   Groups from the Active Directory authentication server need to be mapped to roles on the Access Control Service. You first need to create roles, and then map one or more groups to the appropriate role.

   a. Select the Role Mapping tab.

   b. Click **New Rule**, enter a role name, and click **Save Changes**.

      You do not need to add users to the role. Create as many roles as needed to map the groups from the Active Directory authentication server.

   c. Click **Groups**, and select **Search** to list the groups defined in the domain controller.

   d. Select the group names that you want to map to the new role.

   e. Repeat steps b through d to create and map other groups.

   f. Click **Save Changes**.

Configuring Endpoint Browsers for the SPNEGO

Step-by-Step Procedure

Ensure that endpoint browsers have SPNEGO enabled. For further information, see your third-party documentation.

- Internet Explorer

  From **Security>Local Intranet>Sites>Advanced** add the trusted URL.

  IE performs SPNEGO without any further endpoint configuration but the user is prompted for a username and password. The username and password can be cached.

  To provide single sign-on support, an Internet Explorer configuration can be pushed by configuring a group policy on the Active Directory server. See your third-party documentation for further information.

  Integrated Windows Authentication must be enabled. Use the **Tools>Internet Options>Advanced>Security>Enable Integrated Windows Authentication** path to verify that IWA is enabled.

- Firefox (Windows and MacOS)

  The configuration is in a hidden location. For the URL, type **about:config** and search for the word **trusted**. The required key is the comma separated parameter named **network.negotiate-auth.trusted-uris**.

NOTE: You need to specify the URL of the resource (in this solution, the FQDN or domain controller value UCDC.com).

- Chrome

  Use the Internet Explorer setting. From **Security>Local Intranet>Sites>Advanced** add the trusted URL.

  An internet Explorer configuration can also be pushed by configuring a group policy on the Active Directory server. This configuration is honored by Chrome.

After successful authentication, the standard agentless page is shown along with a second window with the protected resource (unless a pop-up blocker prevents this).

Related Documentation

- *Authentication and Integrated User Firewalls Feature Guide for Security Devices*

CHAPTER 13

Configuring Endpoint Security

Understanding Endpoint Security Using the Infranet Agent with the Junos OS Enforcer

An Infranet agent helps you secure traffic on your network starting with the endpoints that initiate communications as follows:

1. The Infranet agent, which runs directly on the endpoint, checks that the endpoint is compliant with your Unified Access Control (UAC) Host Checker policies. You can use a wide variety of criteria within a UAC Host Checker policy to determine compliance. For example, you can configure the Host Checker policy to confirm that the endpoint is running antivirus software or a firewall or that the endpoint is not running specific types of malware or processes.
2. The Infranet agent transmits the compliance information to the Junos OS Enforcer.
3. The Junos OS Enforcer allows or denies the endpoint access to the resources on your network based on the Host Checker compliance results.

Because the Infranet agent runs directly on the endpoint, you can use the Infranet agent to check the endpoint for security compliance at any time. For instance, when a user tries to sign into the IC Series UAC Appliance, you can require the Infranet agent to send compliance results immediately—the user will not even see the sign-in page until the Infranet agent returns positive compliance results to the IC Series appliance. You can also configure the Infranet agent to check for compliance after the user signs in or periodically during the user session.

If the endpoints running the Infranet agent have appropriate access, they will automatically send their compliance results to the IC Series appliance, and the IC Series appliance will update the authentication table entries accordingly and push them to the Junos OS Enforcer. The Junos OS Enforcer supports connections with the Odyssey Access Client and “agentless” Infranet agents.

Configuring Endpoint Security Using the Infranet Agent with the Junos OS Enforcer

To integrate the Infranet agent into a Junos OS-UAC deployment, no special configuration is required on the Junos OS Enforcer. You simply need to create security policies enabling access to the appropriate endpoints as you would for any other Junos OS-UAC deployment.

Related Documentation

- Understanding Endpoint Security Using the Infranet Agent with the Junos OS Enforcer on page 119
- *Security Basics Guide for Security Devices*
- Understanding User Authentication for Security Devices on page 3

CHAPTER 14

Configuring IPsec

Understanding Junos OS Enforcer Implementations Using IPsec

To configure an SRX Series device to act as a Junos OS Enforcer using IPsec, you must:

- Include the identity configured under the security IKE gateway. The identity is a string such as “gateway1.mycompany.com”, where gateway1.mycompany.com distinguishes between IKE gateways. (The identities specify which tunnel traffic is intended.)
- Include the preshared seed. This generates the preshared key from the full identity of the remote user for Phase 1 credentials.
- Include the RADIUS shared secret. This allows the IC Series UAC Appliance to accept RADIUS packets for extended authentication (XAuth) from the Junos OS Infranet Enforcer.

When configuring IPsec between the IC Series appliance, the Odyssey Access Client, and the SRX Series device, you should note that the following are IKE (or Phase 1) proposal methods or protocol configurations that are supported from the IC Series appliance to the Odyssey Access Client:

- IKE proposal: **authentication-method pre-shared-keys** (you must specify **pre-shared-keys**)
- IKE policy:
  - **mode aggressive** (you must use aggressive mode)
  - **pre-shared-key ascii-text *key*** (only ASCII text preshared-keys are supported)
- IKE gateway: dynamic
  - **hostname *identity*** (you must specify a unique identity among gateways)
  - **ike-user-type group-ike-id** (you must specify **group-ike-id**)
  - **xauth access-profile *profile*** (you must specify **xauth**)

The following are IPsec (or Phase 2) proposal methods or protocol configurations that are supported from the IC Series appliance to the Odyssey Access Client.

- IPsec proposal: **protocol esp** (you must specify **esp**)
- IPsec VPN: **establish-tunnels immediately** (you must specify **establish-tunnels immediately**)

NOTE:

- Only one IPsec VPN tunnel is supported per from-zone to to-zone security policy. This is a limitation on the IC Series appliance.
- Junos OS security policies enable you to define multiple policies differentiated by different source addresses, destination addresses, or both. The IC Series appliance, however, cannot differentiate such configurations. If you enable multiple policies in this manner, the IC Series appliance could potentially identify the incorrect IKE gateway.

Example: Configuring the Device as a Junos OS Enforcer Using IPsec (CLI)

To configure an SRX Series device to act as a Junos OS Enforcer using IPsec:

1. Set system and syslog information using the following configuration statements:

```
system {
host-name test_host;
   domain-name test.mycompany.com;
   host-name test_host;
   root-authentication {
   encrypted-password "$ABC123";
   }
   services {
   ftp;
      ssh;
      telnet;
      web-management {
      http {
         interface ge-0/0/0.0;
      }
   }
}
syslog {
user * {
   any emergency;
      }
      file messages {
      any critical;
      authorization info;
      }
      file interactive-commands {
      interactive-commands error;
   }
```

```
    }
    max-configurations-on-flash 5;
    max-configuration-rollbacks 5;
    license {
    autoupdate {
      url https://ae1.mycompany.com/junos/key_retrieval;
    }
    }
    ntp {
    boot-server 1.2.3.4;
    server 1.2.3.4;
}
}
```

NOTE: On SRX Series devices, the factory default for the maximum number of backup configurations allowed is five. Therefore, you can have one active configuration and a maximum of five rollback configurations. Increasing this backup configuration number will result in increased memory usage on disk and increased commit time.

To modify the factory defaults, use the following commands:

```
root@host# set system max-configurations-on-flash number
root@host# set system max-configuration-rollbacks number
```

where **max-configurations-on-flash** indicates backup configurations to be stored in the configuration partition and **max-configuration-rollbacks** indicates the maximum number of backup configurations.

2. Configure the interfaces using the following configuration statements:

```
interfaces {
ge-0/0/0 {
  unit 0 {
    family inet {
      address 10.64.75.135/16;
    }
  }
}
ge-0/0/1 {
  unit 0 {
    family inet {
      address 10.100.54.1/16;
    }
  }
}
ge-0/0/2 {
  unit 0 {
    family inet {
      10.101.54.1/16;
    }
  }
}
```

3. Configure routing options using the following configuration statements:

```
routing-options {
static {
  route 0.0.0.0/0 next-hop 10.64.0.1;
    route 10.11.0.0/16 next-hop 10.64.0.1;
    route 172.0.0.0/8 next-hop 10.64.0.1;
  route 10.64.0.0/16 next-hop 10.64.0.1;
}
}
```

4. Configure security options using the following configuration statements:

```
security {
ike {
  traceoptions {
    file ike;
    flag all;
  }
  proposal prop1 {
  authentication-method pre-shared-keys;
    dh-group group2;
    authentication-algorithm sha1;
  encryption-algorithm 3des-cbc;
  }
  policy pol1 {
  mode aggressive;
    proposals prop1;
  pre-shared-key ascii-text "$ABC123";
  }
  gateway gateway1 {
  ike-policy pol1;
    dynamic {
    hostname gateway1.mycompany.com;
      connections-limit 1000;
    ike-user-type group-ike-id;
    }
    external-interface ge-0/0/0;
  xauth access-profile infranet;
  }
  gateway gateway2 {
  ike-policy pol1;
    dynamic {
    hostname gateway2.mycompany.com;
      connections-limit 1000;
    ike-user-type group-ike-id;
    }
    external-interface ge-0/0/0;
  xauth access-profile infranet;
}
}
```

5. Configure IPsec parameters using the following configuration statements:

```
ipsec {
proposal prop1 {
  protocol esp;
    authentication-algorithm hmac-sha1-96;
    encryption-algorithm 3des-cbc;
```

```
    lifetime-seconds 86400;
    }
    policy pol1 {
    proposals prop1;
    }
    vpn vpn1 {
    ike {
      gateway gateway1;
      ipsec-policy pol1;
    }
    }
    vpn vpn2 {
    ike {
      gateway gateway2;
      ipsec-policy pol1;
    }
  }
  }
```

6. Configure screen options using the following configuration statements:

```
  screen {
  ids-option untrust-screen {
    icmp {
      ping-death;
      }
      ip {
      source-route-option;
      tear-drop;
      }
      tcp {
      syn-flood {
        alarm-threshold 1024;
          attack-threshold 200;
          source-threshold 1024;
          destination-threshold 2048;
          queue-size 2000;
        timeout 20;
        }
      land;
    }
  }
  }
```

7. Configure zones using the following configuration statements:

```
  zones {
  security-zone trust {
    tcp-rst;
      host-inbound-traffic {
      system-services {
        all;
        }
        protocols {
        all;
      }
      }
```

```
    interfaces {
    ge-0/0/0.0;
  }
  }
  security-zone untrust {
  host-inbound-traffic {
    system-services {
      all;
      }
      protocols {
      all;
    }
    }
    interfaces {
    ge-0/0/1.0;
  }
  }
  security-zone zone101 {
  host-inbound-traffic {
    system-services {
      all;
      }
      protocols {
      all;
    }
    }
    interfaces {
    ge-0/0/2.0;
  }
}
}
```

8. Configure policies for UAC using the following configuration statements:

```
policies {
from-zone trust to-zone trust {
  policy default-permit {
    match {
      source-address any;
        destination-address any;
      application any;
      }
      then {
      permit;
    }
  }
  }
  from-zone trust to-zone untrust {
  policy default-permit {
    match {
      source-address any;
        destination-address any;
      application any;
      }
      then {
      permit;
```

```
    }
    }
    policy default-deny {
    match {
      source-address any;
        destination-address any;
      application any;
      }
      then {
      permit;
    }
    }
    policy pol1 {
    match {
      source-address any;
        destination-address any;
      application any;
      }
      then {
      permit {
        tunnel {
          ipsec-vpn vpn1;
          }
          application-services {
          uac-policy;
        }
        }
        log {
        session-init;
        session-close;
      }
    }
  }
  }
  from-zone untrust to-zone trust {
  policy pol1 {
    match {
      source-address any;
        destination-address any;
      application any;
      }
      then {
      permit;
        log {
        session-init;
        session-close;
      }
    }
  }
  }
  from-zone trust to-zone zone101 {
  policy pol1 {
    match {
      source-address any;
        destination-address any;
      application any;
```

```
        }
        then {
        permit {
          tunnel {
            ipsec-vpn vpn2;
            }
            application-services {
            uac-policy;
          }
          }
          log {
          session-init;
          session-close;
        }
      }
      }
      policy test {
      match {
        source-address any;
          destination-address any;
        application any;
        }
        then {
        permit;
      }
    }
    }
    default-policy {
    deny-all;
  }
  }
  }
```

9. Configure RADIUS server authentication access using the following configuration statements:

```
access {
profile infranet {
  authentication-order radius;
    radius-server {
    10.64.160.120 secret "$ABC123";
  }
}
}
```

10. Configure services for UAC using the following configuration statements:

```
services {
unified-access-control {
  infranet-controller IC27 {
    address 3.23.1.2;
      interface ge-0/0/0.0;
    password "$ABC123";
    }
    infranet-controller prabaIC {
    address 10.64.160.120;
      interface ge-0/0/0.0;
```

```
        password "$ABC123";
        }
        certificate-verification optional;
        traceoptions {
        flag all;
    }
}
}
```

CHAPTER 15

Configuring Captive Portal

Understanding the Captive Portal on the Junos OS Enforcer

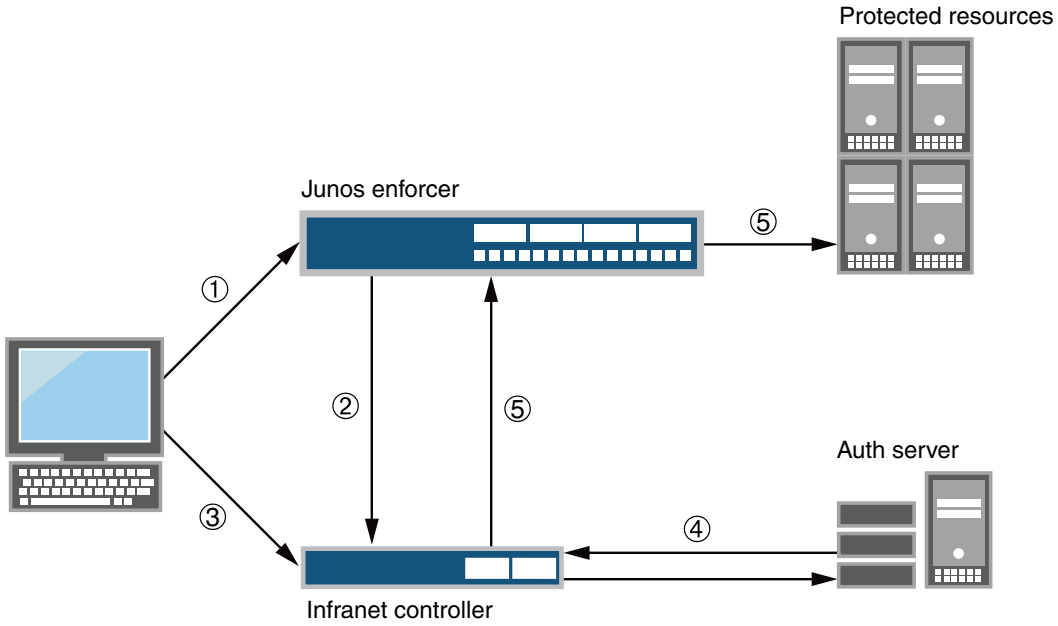
In a Unified Access Control (UAC) deployment, users might not be aware that they must first sign in to the IC Series UAC Appliance for authentication and endpoint security checking before they are allowed to access a protected resource behind the Junos OS Enforcers. To help users sign in to the IC Series appliance, you can configure the captive portal feature. The captive portal feature allows you to configure a policy in the Junos OS Enforcer that automatically redirects HTTP traffic destined for protected resources to the IC Series appliance or to a URL configured in the Junos OS Enforcer.

You can configure a captive portal for deployments that use either source IP enforcement or IPsec enforcement, or a combination of both enforcement methods.

Figure 16 on page 132 shows the captive portal feature enabled on a Junos OS Enforcer. Users accessing protected resources are automatically redirected to the IC Series appliance:

1. Users point to a protected resource using the browser.
2. The Junos OS Enforcer determines that the user is not authenticated and redirects the request to the IC Series appliance or another server.
3. Users enter their Infranet username and password to log in.
4. The IC Series appliance passes the user credentials to an authentication server.
5. After authentication, the IC Series appliance redirects the users to the protected resource they wanted to access.

Figure 16: Enabling the Captive Portal Feature on a Junos OS Enforcer



By default, the Junos OS Enforcer encodes and forwards to the IC Series appliance the protected resource URL that the user entered. The IC Series appliance uses the protected resource URL to help users navigate to the protected resource. The manner in which the IC Series appliance uses the protected resource URL depends on whether or not the user's endpoint is running the Odyssey Access Client or Junos Pulse. If the user's endpoint is not running the Odyssey Access Client or Junos Pulse (that is, it is in an agentless or Java agent configuration), the IC Series appliance automatically opens a new browser window and uses HTTP to access the protected resource after the user signs in. If the endpoint is using the Odyssey Access Client, the IC Series appliance inserts a hypertext link in the webpage that automatically opens after the user signs in. The user must then click that hypertext link to access the protected resource by means of HTTP in the same browser window.

The Junos OS Enforcer supports the captive portal feature only for HTTP traffic. If you attempt to access a protected resource by using HTTPS or a non-browser application (such as an e-mail application), the Junos OS Enforcer does not redirect the traffic. When using HTTPS or a non-browser application, you must manually sign in to the IC Series appliance first before attempting to access protected resources.

Understanding Captive Portal Configuration on the Junos OS Enforcer

To configure the captive portal feature, you create a security policy on the Junos OS Enforcer and then specify a redirection option for the captive portal security policy. You can choose to redirect traffic to an external server or to the IC Series UAC Appliance. You can also choose to redirect all traffic or unauthenticated traffic only.

- Redirecting traffic to an external webserver—You can configure the Junos OS Enforcer to redirect HTTP traffic to an external webserver instead of the IC Series appliance. For example, you can redirect HTTP traffic to a webpage that explains to users the requirement to sign in to the IC Series appliance before they can access the protected resource. You could also include a link to the IC Series appliance on that webpage to help users sign in.
- Redirecting unauthenticated traffic—Select this option if your deployment uses source IP only or a combination of source IP and IPsec. The Junos OS Enforcer redirects clear-text traffic from unauthenticated users to the currently connected IC Series appliance or to an IP address or domain name that you specify in a redirect URL. After a user signs in to the IC Series appliance and the user's endpoint system meets the requirements of the IC Series appliance security policies, the Junos OS Enforcer allows the user's clear-text traffic to pass through in source IP deployments. For IPsec deployments, the Odyssey Access Client creates a VPN tunnel between the user and the Junos OS Enforcer. The Junos OS Enforcer then applies the VPN policy, allowing the encrypted traffic to pass through.
- Redirecting all traffic—Specify this option if you want to redirect all traffic to the URL that you specify in a redirect URL.
- Redirecting traffic with multiple IC Series appliances—You can configure multiple IC Series appliances on your Junos OS Enforcer, but it is connected to only one IC Series appliance at any given time. If the connection to the IC Series appliance fails, the Junos OS Enforcer tries to connect to next configured IC Series appliance. As a result, you cannot be sure which IC Series appliance is connected to the Junos OS Enforcer at any given time. To ensure that the Junos OS Enforcer redirects traffic to the connected IC Series appliance, configure the default redirect URL or the **%ic-ip%** option in the URL.

Related Documentation

- Example: Creating a Captive Portal Policy on the Junos OS Enforcer on page 134

Understanding the Captive Portal Redirect URL Options

By default, after you configure a captive portal policy, the Junos OS Enforcer redirects HTTP traffic to the currently connected IC Series UAC Appliance by using HTTPS. To perform the redirection, the Junos OS Enforcer uses the IP address or domain name that you specified when you configured the IC Series appliance instance on the Junos OS Enforcer. The format of the URL that the Junos OS Enforcer uses for default redirection is:

```
https://%ic-ip%/?target = %dest-url%  &enforcer = %enforcer-id%  &policy =
%policy-id%  &dest-ip = %dest-ip%
```

If you configured your Junos OS Enforcer to work with multiple IC Series appliances in a cluster, and the current IC Series appliance becomes disconnected, the Junos OS Enforcer automatically redirects HTTP traffic to the next active IC Series appliance in its configuration list. The Junos OS Enforcer redirects traffic to only one IC Series appliance at a time.

Otherwise, the browser displays a certificate warning to users when they sign in. You do not need to override the default redirection destination except in these situations:

- You are using a VIP for a cluster of IC Series appliances, and the Junos OS Enforcer is configured to connect to the IC Series appliance physical IP addresses.
- You want to redirect traffic to a webserver instead of the IC Series appliance.
- If, because of split DNS or IP routing restrictions at your site, the Junos OS Enforcer uses a different address for the IC Series appliance than endpoints, you must specify the domain name or IP address that endpoints must use to access the IC Series appliance.

NOTE: If a captive portal policy is configured with the IC Series UAC Appliance URL as the target, then use only HTTPS to redirect traffic.

Table 9 on page 134 lists different options that you can configure in the redirect URL string.

Table 9: Redirect URL String Options

URL String	Description
%dest-url%	Specifies the protected resource which the user is trying to access.
%enforcer-id%	Specifies the ID assigned to the Junos OS Enforcer by the IC Series appliance.
%policy-id%	Specifies the encrypted policy ID for the captive portal security policy that redirected the traffic.
%dest-ip%	Specifies the IP address or hostname of the protected resource which the user is trying to access.
%ic-ip%	Specifies the IP address or hostname of the IC Series appliance to which the Junos OS Enforcer is currently connected.

Example: Creating a Captive Portal Policy on the Junos OS Enforcer

This example shows how to create a captive portal policy on the Junos OS Enforcer. In this example, you deploy a Junos OS Enforcer in front of the data center resources you want to protect and configure the captive portal feature on the Junos OS Enforcer. The

Junos OS Enforcer then automatically redirects HTTP traffic destined for the protected resource to the IC Series UAC Appliance for authentication.

Requirements

Before you begin:

- Deploy the IC Series appliance in the network so that users can access the device. Use the internal port on the IC Series appliance to connect users, the Junos OS Enforcer, and authentication servers. See "Configuring Communications Between the Junos OS Enforcer and the IC Series UAC Appliance (CLI Procedure)" on page 94.
- Set up security zones and interfaces on the Junos OS Enforcer. Make sure that end users are in a different security zone than protected resources. For example, protected resources in the data center are configured in the trusted zone and users in an untrusted zone. See *Example: Creating Security Zones*.
- Add individual users to either an external authentication server or the local authentication server. Set up roles and realms for individual users. You can provision access to protected resources based on your network security needs.

Overview

In this example, you want to protect the trusted zone from users on the LAN by making sure that only compliant and authenticated users are granted access. New users join your network every month. You want to configure the captive portal feature on your system so that unauthenticated users are redirected to the IC Series appliance automatically without requiring new users to remember to log in to the IC Series appliance.

The configuration instructions in this topic describe how to create a security policy called **my-policy**, specify a match condition for this policy, specify the captive portal policy as a part of the UAC policy, and set criteria for redirecting traffic to the IC Series appliance. In this example, the policy **my-policy**:

- Specifies the match condition to include any traffic from a previously configured zone called **trust** to another previously configured zone called **untrust**.
- Specifies the captive portal policy called **my-captive-portal-policy** as part of the UAC policy.
- Specifies the redirect-traffic criteria as **unauthenticated**.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy

and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set security policies from-zone untrust to-zone trust policy my-policy match
  destination-address any source-address any application any
set security policies from-zone untrust to-zone trust policy my-policy then permit
  application-services uac-policy captive-portal my-captive-portal-policy
set services unified-access-control captive-portal my-captive-portal-policy redirect-traffic
  unauthenticated
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.

To create a captive portal policy on the Junos OS Enforcer:

1. Specify the match condition for the policy.

   ```
   [edit security policies from-zone untrust to-zone trust policy my-policy]
   user@host# set match destination-address any source-address any application
     any
   ```

2. Specify the captive portal policy as part of the UAC policy to be applied on the traffic that matches the conditions specified in the security policy.

   ```
   [edit security policies from-zone untrust to-zone trust policy my-policy]
   user@host# set then permit application-services uac-policy captive-portal
     my-captive-portal-policy
   ```

3. Redirect all unauthenticated traffic to the IC Series appliance.

   ```
   [edit services unified-access-control]
   user@host# set captive-portal my-captive-portal-policy redirect-traffic
     unauthenticated
   ```

Results

Confirm your configuration by entering the **show services** and **show security policies** command from configuration mode. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
[edit]
  user@host# show services
  unified-access-control {
  captive-portal my-captive-portal-policy {
    redirect-traffic unauthenticated;
  }
}

[edit]
  user@host# show security policies
  ...
  from-zone untrust to-zone trust {
  policy my-policy {
```

```
        match {
          source-address any;
          destination-address any;
          application any;
        }
        then {
          permit {
            application-services {
              uac-policy {
                captive-portal my-captive-portal-policy;
              }
            }
          }
        }
      }
    }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To confirm that the configuration is working properly, perform this task:

- Verifying the Captive Portal Policy on page 137

Verifying the Captive Portal Policy

Purpose Verify that the captive portal policy was created.

Action From operational mode, enter the **show security policies detail** command.

Related Documentation

- Understanding Captive Portal Configuration on the Junos OS Enforcer on page 133

Example: Configuring a Redirect URL for Captive Portal

This example shows how to redirect traffic to the currently connected IC Series UAC Appliance or to an IP address or domain name that you specify in a redirect URL. We recommend the default configuration that redirects traffic to the IC Series appliance for authentication.

- Requirements on page 137
- Overview on page 138
- Configuration on page 138
- Verification on page 138

Requirements

Before you specify the redirect URL, make sure you configure the captive portal policy. For information about creating the captive portal policy, see "Example: Creating a Captive Portal Policy on the Junos OS Enforcer" on page 134.

Overview

In this example, you configure the URL to redirect traffic to the IC Series appliance and after authentication to forward the traffic automatically to the protected resource. The configuration instructions in this topic describe how to set the URL to https://my-website.com.

You can redirect traffic to the currently connected IC Series appliance or to an IP address or domain name that you specify in a redirect URL. We recommend the default configuration that redirects traffic to the IC Series appliance for authentication.

If you need to override the default redirection destination, you can specify any combination of redirect options:

- **https://*IP or domain name/URL path*/target=%dest-url%**—Forwards users to the protected resource automatically after authentication with the IC Series appliance or webserver. The Junos OS Enforcer replaces the **%dest-url%** parameter with the protected resource URL and then forwards the protected resource URL in encrypted form to the IC Series appliance.
- **https://*IP or domain name/*target=*URL path***—Forwards users to the specified URL automatically after authentication with the IC Series appliance or webserver.
- **https://*IP or domain name/URL path***—Redirects users to the IC Series appliance authentication page but not be forwarded to the protected resource after authentication. Users must manually open a new browser window and enter the protected resource URL again after signing in.
- **redirect-all**—Redirects all traffic to the URL that you specify in a redirect URL.

Configuration

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure the redirect URL for the captive portal feature on the Junos OS Enforcer:

1. Specify the redirect URL for the preconfigured captive portal policy.

```
[edit services unified-access-control]
user@host# set captive-portal my-captive-portal-policy redirect-url
  https://192.168.0.100/target=my-website.com
```

2. If you are done configuring the device, commit the configuration.

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

Verification

To verify the configuration is working properly, enter the **show services unified-access-control captive-portal my-captive-portal-policy** command.

PART 4

Configuring Integrated User Firewall

CHAPTER 16

Understanding Integrated User Firewall

Overview of Integrated User Firewall

This topic includes the following sections:

Integrated User Firewall and Authentication Sources

The SRX Series device already supports Unified Access Control (UAC) integration with Network Access Control (NAC) and a user firewall that can derive its authentication source from Windows Active Directory via the UAC MAG Series Junos Pulse Gateway. However, many customers want simple user firewall functionality without full NAC, and do not want the additional cost or complexity of user role firewall (which has Active Directory dependencies such as Kerberos, SPNEGO on Browsers, Active Directory DNS/Certs, and UAC configuration).

The integrated user firewall feature fulfills the requirement for simplicity. It retrieves user-to-IP address mappings from the Windows Active Directory to use in firewall policies as match criteria. This feature consists of the SRX Series polling the event log of the Active Directory controller to determine, by username and source IP address, who has logged in to the SRX Series device. Then the username and group information are queried

from the LDAP service in the Active Directory controller. Once the SRX Series has the IP address, username, and group relationship information, it generates authentication entries. With the authentication entries, the SRX Series UserFW module enforces user-based and group-based policy control over traffic.

For a comparison of integrated user firewall, user role firewall, and UAC NAC, see "Understanding the Three-Tiered User Firewall Features" on page 3.

Benefits of Integrated User Firewall

The integrated user firewall feature introduces an authentication source via integration with Microsoft Active Directory technology.

- It provides visibility into who is accessing the SRX Series and best-effort security for access to the SRX Series.
- It is a single-box solution, requiring only an SRX Series.
- It requires fewer configuration steps than the UAC integration with NAC, which uses the UAC MAG Series.
- It does not require the configuration of a captive portal, although that option is available to enforce on users who do not authenticate.
- It is ideal for small-to-medium businesses and low-scale deployments.
- It supports High Availability (HA).

How the Integrated User Firewall Works

At a high level, this feature involves the UserID process in the SRX Series Routing Engine, which reads the Windows event log from the Active Directory controller and abstracts IP address-to-user mapping information. The process correlates users to the groups to which they belong, via the LDAP protocol with LDAP service in the Active Directory controller. Thus, the process has gathered enough information to generate authentication entries. The network administrator then references the authentication entries in user firewall security policies to control traffic.

A more detailed explanation of how this feature works is as follows:

1. The SRX Series reads the Active Directory event log to get source IP address-to-username mapping information. To do so, a process in the SRX Series Routing Engine implements a Windows Management Instrumentation (WMI) client with Microsoft Distributed COM/Microsoft RPC stacks and an authentication mechanism to communicate with a Windows Active Directory controller in an Active Directory domain. Using event log information retrieved from the Active Directory controller, the process knows the IP addresses of active Active Directory users and abstracts IP-to-Active Directory username mapping information. The process monitors Active Directory event log changes via the same WMI DCOM interface to adjust local mapping information to reflect any change in the Active Directory server.
2. The process uses LDAP to query the LDAP service interface of the Active Directory to identify the groups to which users belong. Having the IP address, the Active Directory user, and the groups, the process can generate authentication entries accordingly.

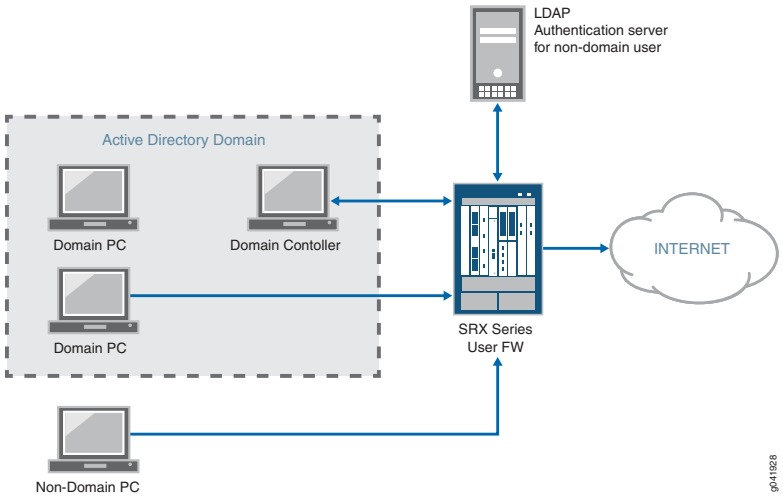
3. The process pushes the authentication entries to the Packet Forwarding Engine authentication table. The Packet Forwarding Engine uses the entries and user policy to apply user firewall access control to traffic.

This feature supports two domains and up to 10 Active Directory controllers in a domain.

Deployment Scenario for User Firewall Integration with Windows Active Directory

Figure 17 on page 143 illustrates a typical scenario where the integrated user firewall feature is deployed. Users in the Active Directory domain and users outside the Active Directory domain want access to the Internet through an SRX Series device. The domain controller might also act as the LDAP server.

Figure 17: Scenario for Integrated User Firewall



The SRX Series device reads and analyzes the event log of the domain controller and generates an authentication table as an Active Directory authentication source for this feature. The user firewall is aware of any domain user on an Active Directory domain device via the Active Directory authentication source. The SRX Series device administrator configures a user firewall policy that enforces the desired user-based or group-based access control.

For any non-domain user or domain user on a non-domain machine, the administrator specifies a captive portal to force the user to do firewall authentication (if the SRX Series supports captive portal for the traffic type). After the user enters a name and password and passes firewall authentication, the SRX Series gets firewall authentication user/group information and can enforce user firewall policy to control the user accordingly.

In addition to captive portal, if the IP address or user information is not available from the event log, the user can again log in to the Windows PC to generate an event log entry. Then the system generates the user's authentication entry accordingly.

Limitations

- Windows Active Directory controllers older than Windows 2003 are not supported.
- Tracking the status of non-Windows Active Directory users is not supported.
- IPv6 addresses are not supported.
- Logical systems are not supported.
- The WMIC does not support multiple users logged onto the same PC.
- Domain controllers and domain PCs must be running Windows OS. The minimum support for a Windows client is Windows XP. The minimum support for a server is Windows Server 2003.
- You cannot use the Primary Group, whether by its default name of Domain Users, or any other name (if you happened to have changed it), in integrated user firewall configurations.

  When a new user is created in Active Directory, the user is added to the global security group Primary Group which is by default called Domain Users. The Primary Group is less specific than other groups created in Active Directory because all users belong to it. Consequently, it can become very large.

Related Documentation

- Understanding the Three-Tiered User Firewall Features on page 3
- Understanding How the WMIC Reads the Event Log on the Domain Controller on page 169
- Understanding Active Directory Authentication Tables on page 144
- Understanding Integrated User Firewall Domain PC Probing on page 172
- Example: Configuring Integrated User Firewall on page 153
- **user-identification (Services) on page 430**

Understanding Active Directory Authentication Tables

This topic includes the following sections:

Active Directory Authentication as an Authentication Source

On an SRX Series device, user information tables serve as the authentication source for information required by firewall security policies. The SRX Series device supports various user information tables including local, user firewall, and Unified Access Control (UAC)

types. The integrated user firewall feature introduces another type of authentication source—Active Directory authentication.

The integrated user firewall feature gathers user and group information for Active Directory authentication by reading domain controller event logs, probing domain PCs, and querying Lightweight Directory Access Protocol (LDAP) services within the configured Windows domain. Up to two Windows domains are supported.

From the user and group information, the integrated user firewall feature generates an Active Directory authentication table on the Routing Engine of the SRX Series device, which then pushes the authentication table to the Packet Forwarding Engine. Security policies use the information in the table to authenticate users and to provide access control for traffic through the firewall.

Active Directory Authentication Tables

The Active Directory authentication table contains the IP address, username, and group mapping information that serves as the authentication source for the SRX Series integrated user firewall feature. Information in the table is obtained by reading Window Active Directory domain controller event logs, probing domain PCs, and querying LDAP services within a specified Windows domain.

Reading domain controller event logs generates a list of IP address-to-user mapping information that is used to create entries in the Active Directory authentication table. Once entries have been added in the table, a query is sent to the LDAP server for user-to-group mapping information.

The LDAP server returns all group information; this includes not only information about the groups you directly belong to, but also all the parent (and parent of the parent and so on) groups that you belong to. Group information returned from the LDAP server is compared with the source identity in security policies. If there is a match, Active Directory authentication table entries are updated to include only the group information provided in the security policy. In this way, only relevant group information is listed in the authentication table. Whenever source identity is updated, the authentication table is also updated to reflect the up-to-date relevant group information for all listed users.

When user traffic arrives at the firewall, the Active Directory authentication table is searched for an entry corresponding to the source IP address of the traffic. If an entry exists, policies matching that entry are applied to the traffic and access is allowed or denied.

Table 10 on page 145 lists Active Directory authentication table support by SRX Series devices. Platform support depends on the Junos OS release in your installation.

Table 10: Active Directory Authentication Table Support by SRX Series Devices

SRX Series Devices	Active Directory Authentication Table Entries	Domains	Active Directory Controllers
SRX100, SRX110, SRX210, SRX220	500	1	5

Table 10: Active Directory Authentication Table Support by SRX Series Devices *(continued)*

SRX Series Devices	Active Directory Authentication Table Entries	Domains	Active Directory Controllers
SRX240	1000	1	5
SRX300	500	1	5
SRX320	500	1	5
SRX340, 345	1000	1	5
SRX550M	5000	2	10
SRX550, SRX650	5000	2	10
SRX1400	20,000	2	10
SRX1500	20,000	2	10
SRX3000 line	50,000	2	10
SRX5000 line	100,000	2	10
vSRX	5000	2	10

Once the maximum number of authentication table entries is reached, no additional entries are created.

To be compliant with the Active Directory authentication table, entries must adhere to the following parameters:

- Usernames are limited to 64 characters.
- Group names are limited to 64 characters.
- Each entry can be associated with up to 200 relevant groups (configured in the source identity field). For example, if you belong to 1000 groups in LDAP and out of these, no more than 200 groups are configured in the source identity field, you are compliant with the Active Directory authentication table.

The Active Directory Authentication table must be enabled as the authentication source for integrated user firewall information retrieval.

```
user@host# set security user-identification authentication source
  active-directory-authentication-table priority priority
```

NOTE: The priority option specifies the sequence in which user information tables are checked. Using the lowest setting for the Active Directory authentication source specifies the highest priority, meaning that the Active Directory authentication source is searched first.

State Information for Active Directory Authentication Table Entries

Active Directory authentication table entries can be in one of four states:

Initial—Specifies that IP address-to-user mapping information was obtained by reading domain controller event logs and an entry was added to the authentication table. Entries in this state are changed to valid when the table is pushed from the Routing Engine to the Packet Forwarding Engine.

Valid—Specifies that a valid entry was obtained by reading domain controller event logs or that a valid response was received from a domain PC probe and the user is a valid domain user.

Invalid—Specifies that an invalid response was received from a domain PC probe and the user is an invalid domain user.

Pending—Specifies that a probe event generated an entry in the authentication table, but no probe response has been received from the domain PC. If a probe response is not received within 90 seconds, the entry is deleted from the table.

For a list of probe responses, see "Understanding Integrated User Firewall Domain PC Probing" on page 172.

To display Active Directory authentication entries, along with their state information, use the following command:

```
user@host>show services user-identification active-directory-access
  active-directory-authentication-table all

Domain: www.example1.net
Total count: 2
Source IP       Username      Groups          State
 192.168.10.2     u2           r1, r3, r4      initial
 192.168.10.3     u3           r5, r6, r4      pending


Domain: www.example2.net
Total count: 2
Source IP       Username      Groups          State
 10.1.1.2         u4           r1, r3, r4      valid
 10.1.1.3         u5           r5, r6, r4      invalid
```

Command options allow you to display information by **user** or **group**, and to define additional output levels—**brief**, **domain**, **extensive**, **node**.

Active Directory Authentication Table Management

Windows domain environments are constantly changing as users log in and out of the network and as network administrators modify user group information. The integrated user firewall feature manages changes in the Windows domain by periodically reading domain controller event logs and querying the LDAP server for user-to-group mapping information. That information is used in updating the Active Directory authentication table as appropriate.

Additionally, a probe function is provided to address changes that occur between reading event logs, or to address the case where event log information is lost. An on-demand probe is triggered when client traffic arrives at the firewall but a source IP address for that client cannot be found in the table. And at any point, manual probing is available to probe a specific IP address.

Changes to the active Directory Authentication table also occur due to source identity changes in the security policy configuration.

Table 11 on page 148 describes events that trigger an Active Directory authentication table update.

Table 11: Events Triggering Active Directory Authentication Table Updates

Event	Active Directory Authentication Table Update
A domain controller event log is read at configured intervals.	New IP address-to-user entries are added in the authentication table in initial state. Group information is retrieved from the LDAP server. When the authentication entry is pushed to Packet Forwarding Engine, the state is changed to valid.
An on-demand or manual probe is sent to a domain PC.	An entry is added in the authentication table in pending state. If a probe response is not returned within 90 seconds, the state of the entry is deleted.
An on-demand or manual probe response is received from a domain PC.	Based on the response, entries in pending state are changed to valid or invalid. For valid responses, the group information is retrieved from the LDAP server. For invalid responses, the entry is marked as invalid.
An LDAP server query identifies new user-to-group mapping information.	Entries are updated with the group information.
An LDAP server query identifies deleted user information.	Entries associated with that user are deleted from the table.
An LDAP server query identifies deleted group information.	The affected group information is updated. For example, user2 belongs to group2, and group2 belongs to group1. And, group1 is listed as a source-identity for group2. For any authentication entry of user2, group1 is listed in its relevant groups. However, if group2 is removed from the LDAP server, user2 loses the connection with group1, and as a result, group1 is removed from the user2 authentication table.

Table 11: Events Triggering Active Directory Authentication Table Updates *(continued)*

Event	Active Directory Authentication Table Update
An LDAP server query identifies added group information.	If the group is referenced in a security policy, entries associated with this group are updated to add the group information.
The source identity information is removed from a security policy configuration.	Entries associated with the source identity are deleted from Active Directory authentication table.

NOTE: If an entry is deleted from the table, any sessions attached to that entry are also deleted. If an entry in the table is updated to add or remove group information, there is no impact to existing sessions for that entry.

NOTE: When you use the CLI to delete an active directory authentication entry, the system closes the related session and writes a session-close message to the log file. However, the session-close message does not contain the source identity information for the user, that is, the user and user group information."

To manually delete an entry from the table, use the **request services user-identification active-directory-access active-directory-authentication-table** command. Options exist for deleting a specific IP address, domain, group, or user.

To clear the contents of the Active Directory authentication table, use the **clear services user-identification active-directory access active-directory-authentication-table** command.

Timeout Interval for Table Entries

When a user is no longer active, a timer is started for that user's entry in the Active Directory authentication table. When time is up, the user's entry is removed from the table. Entries in the table remain active as long as there are sessions associated with the entry.

To set the timeout value, use the following statement:

```
user@host# set services user-identification active-directory-access
    authentication-entry-timeout minutes
```

The default **authentication-entry-timeout** interval is 30 minutes. To disable timeouts, set the interval to 0.

NOTE: We recommend that you disable timeouts when disabling on-demand probing in order to prevent someone from accessing the Internet without logging in again.

To view timeout information for Active Directory authentication table entries, use the following command:

```
user@host>show services user-identification active-directory-access
    active-directory-authentication-table all extensive

Domain: www.example1.net
Total entries: 2
Source IP: 192.168.1.2
Username: u2
Groups: r1, r3, r4
State: initial
Access start date: 2014-03-22
Access start time: 10:56:58
Age time: 20 min

Source IP: 192.168.1.3
Username: u3
Groups: r5, r6, r4
State: pending
Access start date: 2014-03-22
Access start time: 10:46:58
Age time: 10 min
```

This example shows that the timer has started for two entries—the entry for user u2 will time out in 20 minutes, while the entry for user u3 will time out in 10 minutes. When session traffic is associated with an entry, the age time value changes to “infinite.”

Related Documentation

- Overview of Integrated User Firewall on page 141
- Understanding Integrated User Firewall Domain PC Probing on page 172
- LDAP Functionality in Integrated User Firewall on page 150
- **active-directory-authentication-table on page 296**
- **user-identification (Services) on page 430**

LDAP Functionality in Integrated User Firewall

This topic includes the following sections:

Role of LDAP in Integrated User Firewall

In order to get the user and group information necessary to implement the Integrated User Firewall feature, the SRX Series device uses the Lightweight Directory Access Protocol (LDAP). The SRX Series acts as an LDAP client communicating with an LDAP server. In a common implementation scenario of the integrated user firewall feature, the

domain controller acts as the LDAP server. The LDAP module in the SRX Series, by default, queries the Active Directory in the domain controller.

The SRX Series downloads user and group lists from the LDAP server. The device also queries the LDAP server for user and group updates. The SRX Series downloads a first-level, user-to-group mapping relationship and then calculates a full user-to-group mapping.

The use of "LDAP" in this section applies specifically to LDAP functionality within the integrated user firewall feature.

LDAP Server Configuration and Base Distinguished Name

Most of the LDAP server configuration is optional, leveraging the common implementation scenario where the domain controller acts as the LDAP server. The SRX Series periodically (every two minutes) queries the LDAP server to get the user and group information changed since the last query.

LDAP's Authentication Method

By default, the LDAP authentication method uses simple authentication. The client's username and password are sent to the LDAP server in plaintext. Keep in mind that the password is clear and can be read from the network.

To avoid exposing the password, you can use simple authentication within an encrypted channel [namely Secure Sockets layer (SSL)], as long as the LDAP server supports LDAP over SSL (LDAPS). After enabling SSL, the data sent from the LDAP server to the SRX Series is encrypted. To enable SSL, see the **user-group-mapping** statement.

LDAP Server's Username, Password, and Server Address

The LDAP server's username, password, IP address, and port are all optional, but they can be configured.

- If the username and password are not configured, the system uses the configured domain controller's username and password.
- If the LDAP server's IP address is not configured, the system uses the address of one of the configured Active Directory domain controllers.
- If the port is not configured, the system uses port 389 for plaintext or port 636 for encrypted text.

Caching and Calculation of User-to-Group Mappings

The SRX Series device caches user-to-group mappings in its local database when the **show services user-identification active-directory-access user-group-mapping** operation is performed. This command displays the users who belong to a group or the groups to which a user belongs.

Three events cause a user-to-group mapping to be removed from the cache:

- A source-identity is removed from a referenced firewall policy (because only source-identities referenced in a policy are stored in the authentication table).
- The LDAP configuration is deleted from the customer's configuration, so all cached Active Directory user-to-group mappings for the domain are removed.
- The user-to-group mapping is deleted from the LDAP server.

The SRX periodically queries to get user and group information from the LDAP server in real time. The user list and the group list show only cached users or groups, not all users or groups in the LDAP server. From this information, the SRX Series calculates one-level mapping relationships. The user list, group list, and mapping are cached in the local database.

Updating Group Information in the Authentication Entry Table

The SRX Series device queries to get the changed users and groups based on the prior query results from the LDAP server. The SRX Series updates the local database and triggers an authentication entry update. Only user/group mappings that are already cached are updated. Other users and groups that are not in the database do not have their mapping relationships cached.

LDAP Server Status and Statistics

You can verify the LDAP connection status by issuing the **show services user-identification active-directory-access user-group-mapping status** command.

You can see counts of queries made to the LDAP server by issuing the **show services user-identification active-directory-access statistics user-group-mapping** command.

Active Directory Autodiscovery

The integrated user firewall feature provides the IP address and Active Directory name of the domain. The auto-discovery feature can use the Active Directory's global catalog feature and then query DNS for a list of global catalogs. The global catalogs in the list are typically provided in a weighted order based on criteria such as network location, system-set weights based on global catalog server size, and so on. Once the customer has the list of Active Directories, the customer can configure it for both event log reading and LDAP search.

Related Documentation

- Overview of Integrated User Firewall on page 141
- Understanding How the WMIC Reads the Event Log on the Domain Controller on page 169
- show services user-identification active-directory-access statistics on page 529
- show services user-identification active-directory-access user-group-mapping on page 510
- user-group-mapping on page 428

Example: Configuring Integrated User Firewall

This example shows how to implement the integrated user firewall feature by configuring a Windows Active Directory domain, an LDAP base, unauthenticated users to be directed to captive portal, and a security policy based on a source identity.

Requirements

This example uses the following hardware and software components:

- One SRX Series device
- Junos OS Release 12.1X47-D10 or later for SRX Series devices

No special configuration beyond device initialization is required before configuring this feature.

Overview

In a typical scenario for the integrated user firewall feature, domain and non-domain users want to access the Internet through an SRX Series device. The SRX Series device reads and analyzes the event log of the domain controllers configured in the domain. Thus, the SRX Series device detects domain users on an Active Directory domain controller. Active Directory domain generates an authentication table as the Active Directory authentication source for the integrated user firewall. The SRX Series device uses this information to enforce the policy to achieve user-based or group-based access control.

For any non-domain user or domain user on a non-domain device, the network administrator can specify a captive portal to force the user to submit to firewall authentication (if the SRX Series device supports captive portal for the traffic type. For example, HTTP). After the user enters a name and password and passes firewall authentication, the SRX Series device gets firewall authentication user-to-group mapping information from the LDAP server and can enforce user firewall policy control over the user accordingly.

NOTE: You cannot use the Primary Group, whether by its default name of Domain Users or any other name, if you changed it, in integrated user firewall configurations.

When a new user is created in Active Directory (AD), the user is added to the global security group Primary Group which is by default Domain Users. The Primary Group is less specific than other groups created in AD because all users belong to it. Also, it can become very large.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set services user-identification active-directory-access domain example.net
  user-group-mapping ldap base DC=example,DC=net
set services user-identification active-directory-access domain example.net user
  administrator password pwd
set services user-identification active-directory-access domain example.net
  domain-controller ad1 address 192.0.2.15
set access profile profile1 authentication-order ldap
set access profile profile1 authentication-order password
set access profile profile1 ldap-options base-distinguished-name
  CN=Users,DC=example,DC=com
set access profile profile1 ldap-options search search-filter sAMAccountName=
set access profile profile1 ldap-options search admin-search distinguished-name
  CN=Administrator,CN=Users,DC=example,DC=com
set access profile profile1 ldap-options search admin-search password password
set access profile profile1 ldap-server 192.0.2.3
set security policies from-zone trust to-zone untrust policy p1 match source-address any
set security policies from-zone trust to-zone untrust policy p1 match destination-address
  any
set security policies from-zone trust to-zone untrust policy p1 match application any
set security policies from-zone trust to-zone untrust policy p1 match source-identity
  unauthenticated-user
set security policies from-zone trust to-zone untrust policy p1 match source-identity
  unknown-user
set security policies from-zone trust to-zone untrust policy p1 then permit
  firewall-authentication user-firewall access-profile profile1
set security policies from-zone trust to-zone untrust policy p2 match source-address any
set security policies from-zone trust to-zone untrust policy p2 match destination-address
  any
set security policies from-zone trust to-zone untrust policy p2 match application any
set security policies from-zone trust to-zone untrust policy p2 match source-identity
  “example.com\user1”
set security policies from-zone trust to-zone untrust policy p2 then permit
set security user-identification authentication source active-directory-authentication-table
  priority 125
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To establish a Windows Active Directory domain, to configure captive portal, and to configure another security policy, perform the steps in this section.

Once configured, when traffic arrives, the SRX Series device consults the user firewall process, which in turn consults the Active Directory authentication source to determine whether the source is in its authentication table. If the user firewall hits an authentication entry, the SRX Series device checks the policy configured in Step 4 for further action. If the user firewall does not hit any authentication entry, the SRX Series device checks the policy configured in Step 3 to enforce the user to do captive portal.

1. Configure the LDAP base distinguished name.

```
[edit services user-identification]
user@host# set active-directory-access domain example.com user-group-mapping
  ldap base DC=example,DC=com
```

2. Configure a domain name, the username and password of the domain, and the name and IP address of the domain controller in the domain.

```
[edit services user-identification]
user@host# set active-directory-access domain example.com user administrator
  password xxxxx
user@host# set active-directory-access domain example.net domain-controller
  ad1 address 192.0.2.15
```

3. Configure an access profile and set the authentication order and LDAP options.

```
[edit access profile profile1]
user@host# set authentication-order ldap
user@host# set authentication-order password
user@host# set ldap-options base-distinguished-name
  CN=Users,DC=example,DC=com
user@host# set ldap-options search search-filter sAMAccountName=
user@host# set ldap-options search admin-search distinguished-name
  CN=Administrator,CN=Users,DC=example,DC=com
user@host# set ldap-options search admin-search password pwd
user@host# set ldap-server 192.0.2.3
```

4. Configure a policy for the source-identity “unauthenticated-user” and “unknown-user” and enable the firewall authentication captive portal. Configuring the source identity is required in case there is no authentication sources configured, it is disconnected.

```
[edit security policies from-zone trust to-zone untrust policy p1]
user@host# set match source-address any
user@host# set match destination-address any
user@host# set match application any
user@host# set match source-identity unauthenticated-user
user@host# set match source-identity unknown-user
user@host# set then permit firewall-authentication user-firewall access-profile
  profile1
user@host#set then permit firewall-authentication user-firewall domain
  example.userfw.com
```

5. Configure a second policy to enable a specific user.

```
[edit security policies from-zone trust to-zone untrust policy p2]
user@host# set match source-address any
user@host# set match destination-address any
user@host# set match application any
user@host# set match source-identity ""example.com\user1""
user@host# set then permit
```

NOTE: When you specify a source identity in a policies statement, prepend the domain name and a backslash to the group name or username. Enclose the combination in quotation marks.

6. Set the Active Directory authentication table as the authentication source for integrated user firewall information retrieval and specify the sequence in which user information tables are checked.

```
[edit security ]
user@host# set user-identification authentication source
  active-directory-authentication-table priority 125
```

NOTE: You must set the Active Directory authentication table as the authentication source for integrated user firewall information retrieval and specify the sequence in which user information tables are checked using the command **set security user-identification authentication source active-directory-authentication-table priority** *value*.

The default value of this option is 125. The default priority for all the authentication sources is as follows:

- Local authentication: 100
- Integrated user firewall: 125
- User role firewall: 150
- Unified Access Control (UAC): 200

The field **priority** specifies the sources for the Active Directory authentication table. The value set determines the sequence for searching among various supported authentication tables to retrieve a user role. Note that these are the only currently supported values. You can enter any value from 0 through 65,535. The default priority of the Active Directory authentication table is 125. This means that even if you do not specify a priority value, the Active Directory authentication table will be searched starting at sequence of value 125 (integrated user firewall).

For more details, see "Understanding Active Directory Authentication Tables" on page 144 and active-directory-authentication-table.

(Optional) Configuration of PKI and SSL Forward Proxy to Authenticate Users

Step-by-Step Procedure

Optionally, for non-domain users, you can configure public key infrastructure (PKI) to validate integrity, confidentiality, and authenticity of traffic. PKI includes digital certificates issued by the Certificate Authority (CA), certificate validity and expiration dates, details about the certificate owner and issuer, and security policies.

NOTE: For any non-domain user or domain user on a non-domain machine, the administrator specifies a captive portal to force the user to do firewall authentication (if the SRX Series device supports captive portal for the traffic type). After the user enters a name and password and passes firewall authentication, the SRX Series device gets firewall authentication user/group information and can enforce the user firewall policy to control the user accordingly. In addition to captive portal, if the IP address or user information is not available from the event log, the user can again log in to the Windows PC to generate an event log entry. Then the system generates the user's authentication entry accordingly.

To enable the SRX Series device to authenticate the users through HTTPs, the SSL forward proxy must be configured and enabled. You need to generate a local certificate, add an SSL termination profile, add an SSL proxy profile, and reference the SSL proxy profile in the security policy. If the SSL forward proxy is not enabled, the SRX Series device cannot authenticate users who are using HTTPS, but for users who are using HTTP, FPT, and Telnet, the authentication can be performed as expected.

To generate PKI and enable SSL forward proxy, perform the following steps:

1. Generate a PKI public/private key pair for a local digital certificate.

   ```
   user@host# request security pki generate-key-pair certificate-id ssl-inspect-ca size
     2048 type rsa
   ```

2. Manually generate a self-signed certificate for the given distinguished name.

   ```
   user@host# request security pki local-certificate generate-self-signed certificate-id
     ssl-inspect-ca domain-name www.mycompany.net subject
     "CN=www.mycompany.com,OU=IT,O=MY COMPANY,L=Sunnyvale,ST=CA,C=US"
     email security-admin@mycompany.net
   ```

3. Define the access profile to be used for SSL termination services. This option is available only on high-end SRX Series devices.

   ```
   user@host# set services ssl termination profile for_userfw server-certificate
     ssl-inspect-ca
   ```

4. Configure the loaded certificate as root-ca in the SSL proxy profile. This option is available only on high-end SRX Series devices.

   ```
   user@host# set services ssl proxy profile ssl-inspect-profile root-ca ssl-inspect-ca
   ```

5. Specify the **ignore-server-auth-failure** option if you do not want to import the entire CA list and you do not want dropped sessions. This option is available only on high-end SRX Series devices.

   ```
   user@host# set services ssl proxy profile ssl-inspect-profile actions
     ignore-server-auth-failure
   ```

6. Add an SSL termination profile into security policies. This option is available only on high-end SRX Series devices.

   ```
   user@host# set security policies from-zone untrust to-zone trust policy p1 then
     permit firewall-authentication user-firewall ssl-termination-profile for_userfw
   ```

Results

From configuration mode, confirm your integrated user firewall configuration by entering the **show services user-identification active-directory-access** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show services user-identification active-directory-access
  domain example.com {
    user {
      administrator;
      password "$ABC123"; ## SECRET-DATA
    }
    domain-controller ad1 {
      address 192.0.2.15;
    }
    user-group-mapping {
      ldap {
        base DC=example,DC=com;
      }
    }
  }
```

From configuration mode, confirm your policy configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show security policies
  from-zone trust to-zone untrust {
    policy p1 {
      match {
        source-address any;
        destination-address any;
        application any;
        source-identity unauthenticated-user;
        source-identity unknown-user;
      }
      then {
        permit {
          firewall-authentication {
            user-firewall {
              access-profile profile1;
            }
```

```
            }
          }
        }
      }
      policy p2 {
        match {
          source-address any;
          destination-address any;
          application any;
          source-identity “example.com\user1”;
        }
        then {
          permit;
        }
      }
    }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

Verifying Connectivity to a Domain Controller

Purpose Verify that at least one domain controller is configured and connected.

Action From operational mode, enter the **show services user-identification active-directory-access domain-controller status** command.

Meaning The domain controller is shown to be connected or disconnected.

Verifying the LDAP Server

Purpose Verify that the LDAP server is providing user-to-group mapping information.

Action From operational mode, enter the **show services user-identification active-directory-access user-group-mapping status** command.

Meaning The LDAP server address, port number, and status are displayed.

Verifying Authentication Table Entries

Purpose See which groups users belong to and the users, groups, and IP addresses in a domain.

Action From operational mode, enter the **show services user-identification active-directory-access active-directory-authentication-table all** command.

Meaning The IP addresses, usernames, and groups are displayed for each domain.

Verifying IP-to-User Mapping

Purpose Verify that the event log is being scanned.

Action From operational mode, enter the **show services user-identification active-directory-access statistics ip-user-mapping** command.

Meaning The counts of the queries and failed queries are displayed.

Verifying IP Probe Counts

Purpose Verify that IP probes are occurring.

Action From operational mode, enter the **show services user-identification active-directory-access statistics ip-user-probe** command.

Meaning The counts of the IP probes and failed IP probes are displayed.

Verifying User-to-Group Mapping Queries

Purpose Verify that user-to-group mappings are being queried.

Action From operational mode, enter the **show services user-identification active-directory-access statistics user-group-mapping** command.

Meaning The counts of the queries and failed queries are displayed.

Related Documentation

- Overview of Integrated User Firewall on page 141
- Understanding the Three-Tiered User Firewall Features on page 3
- policies on page 366
- show services user-identification active-directory-access active-directory-authentication-table on page 522
- show services user-identification active-directory-access domain-controller status on page 526
- show services user-identification active-directory-access statistics on page 529

- show services user-identification active-directory-access user-group-mapping on page 510

Example: Configuring Integrated User Firewall to Use Web-Redirect for Unauthenticated Users Requesting Access to HTTP-Based Resources

This example shows how to use web-redirect for unauthenticated users requesting access to HTTP-based resources

Requirements

This example uses the following hardware and software components:

- One SRX Series device
- Junos OS Release 15.1X49-D70 or later for SRX Series devices

No special configuration beyond device initialization is required before configuring this feature.

Overview

The fwauth access profile redirects **web-redirect** requests of pass-through traffic to HTTP webauth (in JWEB httpd server). Once authentication is successful, fwauth creates a firewall authentication for the user firewall.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [**edit**] hierarchy level, and then enter **commit** from configuration mode.

```
set system services web-management http
set interfaces ge-0/0/1 unit 0 family inet address 3.0.0.25/24 web-authentication http
set security policies from-zone trust to-zone untrust policy p1 match source-address any
set security policies from-zone trust to-zone untrust policy p1 match destination-address
    any
set security policies from-zone trust to-zone untrust policy p1 match application any
set security policies from-zone trust to-zone untrust policy p1 match source-identity
    unauthenticated-user
set security policies from-zone trust to-zone untrust policy p1 match source-identity
    unknown-user
set security policies from-zone trust to-zone untrust policy p1 then permit
    firewall-authentication user-firewall access-profile profile1 web-redirect
set security policies from-zone trust to-zone untrust policy p1 then permit
    firewall-authentication user-firewall domain ad03.net
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure the integrated user firewall to use web-redirect for unauthenticated users requesting access to HTTP-based resources:

1. Enable web-management support for HTTP traffic.

```
[edit system services]
user@host# set system services web-management http
```

2. Configure interfaces and assign IP addresses. Enable Web authentication on ge-0/0/1 interface.

```
[edit interfaces]
user@host# set interfaces ge-0/0/1 unit 0 family inet address 3.0.0.25/24
  web-authentication http
```

3. Configure security policies that specifies an unauthenticated-user or unknown-user as the source-identity.

```
[edit security policies from-zone trust to-zone untrust policy p1]
user@host# set match source-address any
user@host# set match destination-address any
user@host# set match application any
user@host# set match source-identity unauthenticated-user
user@host# set match source-identity unknown-user
```

4. Configure a security policy that permits firewall authentication of a user firewall with **web-redirect** as the action and specifies a pre configured access profile for the user.

```
[edit security policies from-zone trust to-zone untrust policy p1]
user@host# set then permit firewall-authentication user-firewall access-profile
  profile1 web-redirect
```

5. Configure a security policy that specifies the domain name.

```
[edit security policies from-zone trust to-zone untrust policy p1]
user@host# set then permit firewall-authentication user-firewall domain ad03.net
```

Results

From configuration mode, confirm your configuration by entering the **show system services** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show system services
  web-management {
    http {
      port 123;
    }
  }
```

From configuration mode, confirm your integrated user-firewall configuration by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show interfaces
```

```
ge-0/0/1 {
  unit 0 {
    family inet {
      address 46.20.0.1/24 {
        web-authentication http;
      }
      address 3.0.0.25/24 {
        web-authentication http;
      }
    }
    family inet6 {
      address 3046:20::1/64;
      }
    }
  }
```

From configuration mode, confirm your integrated user-firewall configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show security policies
  from-zone trust to-zone untrust {
    policy p1 {
      match {
        source-address any;
        destination-address any;
        application any;
        source-identity unauthenticated-user;
        source-identity unknown-user;
      }
      then {
        permit {
          firewall-authentication {
            user-firewall {
              access-profile profile1;
              web-redirect;
              domain ad03.net;
            }
          }
        }
      }
    }
  }
```

From configuration mode, confirm your policy configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verify the Configuration.

Purpose Verify that the configuration is correct.

Action From operational mode, enter the **show security policies** command.

Sample Output

```
user@host> show security policies
  Default policy: permit-all
  From zone: PCzone, To zone: Tunnelzone
    Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
        Source addresses: any
        Destination addresses: any
        Applications: junos-ftp, junos-tftp, junos-dns-tcp, junos-dns-udp
        Action: permit
```

Meaning Display the security policy that permits firewall authentication of a user firewall with web-redirect as the action.

Related Documentation

- Overview of Integrated User Firewall on page 141
- Example: Configuring Integrated User Firewall on page 153

Example: Configuring Integrated User Firewall to Use Web-Redirect-to-HTTPS for Unauthenticated Users Requesting Access to HTTPS-Based Resources

This example shows how to use web-redirect-to-https for unauthenticated users requesting access to HTTPS-based resources.

Requirements

This example uses the following hardware and software components:

- One SRX Series device
- Junos OS Release 15.1X49-D70 or later for SRX Series devices

No special configuration beyond device initialization is required before configuring this feature.

Overview

The fwauth access profile redirects **web-redirect-to-https** requests of pass-through traffic to HTTPS webauth (in JWEB httpd server). Once authentication is successful, fwauth creates a firewall authentication for the user firewall.

Configuration

CLI Quick Configuration To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy

and paste the commands into the CLI at the [**edit**] hierarchy level, and then enter **commit** from configuration mode.

```
set system services web-management https pki-local-certificate my-test-cert
set interfaces ge-0/0/1 unit 0 family inet address 3.0.0.25/24 web-authentication https
set security policies from-zone trust to-zone untrust policy p1 match source-address any
set security policies from-zone trust to-zone untrust policy p1 match destination-address
  any
set security policies from-zone trust to-zone untrust policy p1 match application any
set security policies from-zone trust to-zone untrust policy p1 match source-identity
  unauthenticated-user
set security policies from-zone trust to-zone untrust policy p1 match source-identity
  unknown-user
set security policies from-zone trust to-zone untrust policy p1 then permit
  firewall-authentication user-firewall access-profile profile1 web-redirect-to-https
set security policies from-zone trust to-zone untrust policy p1 then permit
  firewall-authentication user-firewall domain ad03.net
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure the web-redirect-to-https for unauthenticated users requesting access to HTTP/HTTPS-based resources:

1. Enable web-management support for HTTPS traffic.

   ```
   [edit system services]
   user@host# set system services web-management https pki-local-certificate
     my-test-cert
   ```

2. Configure interfaces and assign IP addresses. Enable Web authentication on ge-0/0/1 interface.

   ```
   [edit interfaces]
   user@host# set interfaces ge-0/0/1 unit 0 family inet address 3.0.0.25/24
     web-authentication https
   ```

3. Configure security policies that specifies an unauthenticated-user or unknown-user as the source-identity.

   ```
   [edit security policies from-zone trust to-zone untrust policy p1]
   user@host# set match source-address any
   user@host# set match destination-address any
   user@host# set match application any
   user@host# set match source-identity unauthenticated-user
   user@host# set match source-identity unknown-user
   ```

4. Configure a security policy that permits firewall authentication of a user firewall with **web-redirect-to- https** as the action and specifies a pre configured access profile for the user.

   ```
   [edit security policies from-zone trust to-zone untrust policy p1]
   user@host# set then permit firewall-authentication user-firewall access-profile
     profile1 web-redirect-to-https
   ```

5. Configure a security policy that specify the domain name.

```
[edit security policies from-zone trust to-zone untrust policy p1]
user@host# set then permit firewall-authentication user-firewall domain ad03.net
```

Results From configuration mode, confirm your configuration by entering the **show system services** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show system services
  web-management {
    http {
      port 123;
    }
    https {
      pki-local-certificate pki-local-certificate;
    }
  }
```

From configuration mode, confirm your integrated user-firewall configuration by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show interfaces
  ge-0/0/1 {
    unit 0 {
      family inet {
        address 46.20.0.1/24 {
          web-authentication {
            http;
            https;
          }
        }
      }
      family inet6 {
        address 3046:20::1/64;
        }
      }
    }
```

From configuration mode, confirm your integrated user-firewall configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show security policies
  from-zone trust to-zone untrust {
    policy p1 {
      match {
        source-address any;
        destination-address any;
        application any;
        source-identity unauthenticated-user;
        source-identity unknown-user;
      }
      then {
        permit {
          firewall-authentication {
```



```
            user-firewall {
              access-profile profile1;
              web-redirect;
              web-redirect-to-https;
              domain ad03.net;
            }
          }
        }
      }
    }
```

From configuration mode, confirm your policy configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verify the Configuration.

Purpose Verify that the configuration is correct.

Action From operational mode, enter the **show security policies** command.

Sample Output

```
user@host> show security policies
  Default policy: permit-all
  From zone: PCzone, To zone: Tunnelzone
    Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
        Source addresses: any
        Destination addresses: any
        Applications: junos-ftp, junos-tftp, junos-dns-tcp, junos-dns-udp
        Action: permit
```

Meaning Display the security policy permit that permits firewall authentication of a user firewall with web-redirect-to-https as the action.

Related Documentation

- Overview of Integrated User Firewall on page 141
- Example: Configuring Integrated User Firewall on page 153

CHAPTER 17

Managing Event Logs

Understanding How the WMIC Reads the Event Log on the Domain Controller

This topic includes the following sections:

Windows Management Instrumentation Client

When you configure the integrated user firewall feature on an SRX Series device, the SRX Series establishes a Windows Management Instrumentation (WMI)/Distributed Component Object Module (DCOM) connection to the domain controller. The SRX Series acts as a WMI client (WMIC). It reads and monitors the security event log on the domain controller. The SRX Series analyzes the event messages to generate IP address-to-user mapping information.

All configuration regarding the WMIC is optional; it will function with default values. After the domain is configured (by the **set services user-identification active-directory-access domain** statement), the WMIC starts to work. The WMIC connection to the domain controller uses the same user credentials as those configured for the domain.

CAUTION: Integrated user firewall uses NTLMv2 as the default WMIC authentication protocol for security reasons. NTLMv1 exposes the system to attacks in which authentication hashes could be extracted from NTLMv1 authentication responses.

For compatibility with integrated user firewall, you must apply the latest version of the Microsoft SP2 patch if you are running an older version of Windows OS, including Windows 2000, Windows XP, and Windows 2003.

WMIC Reads the Event Log on the Domain Controller

The following SRX Series behaviors apply to reading the event log:

- The SRX Series monitors the event log at a configurable interval, which defaults to 10 seconds.
- The SRX Series reads the event log for a certain timespan, which you can configure. The default timespan is one hour. Each time at WMIC startup, the SRX Series checks the last timestamp and the timespan. If the last timestamp is older than the current timespan, then the timespan takes effect. After the WMIC and the UserID process start working, the timespan does not apply; the SRX Series simply reads the latest event log.
- During WMIC startup, the SRX Series has a maximum count of events it will read from the event log, and that maximum is not configurable.
  - On SRX Series branch devices, the maximum count is 100,000.
  - On SRX high-end devices, the maximum count is 200,000.

  During WMIC startup, this maximum is used with the timespan setting, so that if either limit is reached, the WMIC stops reading the event log.
- After a failover, the SRX Series reads the event log from the latest event log timestamp.
- In a chassis cluster environment, the WMIC works on the primary node only.

Specifying IP Filters to Limit IP-to-User Mapping

You can specify IP filters to limit the IP address-to-user mapping information that the SRX Series generates from the event log.

To understand when a filter is useful for such mapping, consider the following scenario. A customer deploys 10 SRX Series devices in one domain, and each SRX Series controls a branch. All 10 SRX Series devices read all 10 branch user login event logs in the domain controller. However, the SRX Series is configured to detect only whether the user is authenticated on the branch it controls. By configuring an IP filter on the SRX Series, the SRX Series reads only the IP event log under its control.

You can configure a filter to include or exclude IP addresses or prefixes. You can specify a maximum of 20 addresses for each filter.

Event Log Verification and Statistics

You can verify that the authentication table is getting IP address and user information by issuing the **show services user-identification active-directory-access active-directory-authentication-table all** command. A list of IP address-to-user mappings is displayed for each domain. The table contains no group information until LDAP is running.

You can see statistics about reading the event log by issuing the **show services user-identification active-directory-access ip-user-mapping statistics domain** command.

Related Documentation

- show services user-identification active-directory-access active-directory-authentication-table on page 522
- Overview of Integrated User Firewall on page 141
- LDAP Functionality in Integrated User Firewall on page 150
- Using Firewall Authentication as an Alternative to WMIC on page 171

Using Firewall Authentication as an Alternative to WMIC

This topic includes the following sections:

WMIC Limitations

The primary method for the integrated user firewall feature to get IP address-to-user mapping information is for the SRX Series device to act as a WMI client (WMIC). However, the WMIC has limitations, such as the following:

- On Windows XP or Server2003, the Windows firewall does not allow the WMIC request to pass through because of the dynamic port allocation of the Distributed Component Object Model (DCOM). Therefore, for these operating systems when Windows firewall is enabled, the PC does not respond to the WMIC probe.
- Because the event-log-reading and PC probe functions both use WMI, using a global policy to disable the WMI-to-PC probe also affects event log reading.

Because these cases might result in the failure of the PC probe, a backup method for getting IP address-to-user mappings is needed. That method is to use firewall authentication to identify users.

Firewall Authentication as a Backup Method for IP Address-to-User Mappings

If you want to use firewall authentication to identify users for the integrated user firewall feature, specify a domain name in the **set security policies from-zone trust to-zone untrust policy <policy-name> then permit firewall-authentication user-firewall domain <domain-name>** statement.

If a domain is configured in that statement, fwauth recognizes that the domain is for a domain authentication entry, and will send the domain name to the fwauth process along with the authentication request. After it receives the authentication response, fwauth deletes that domain authentication entry. The fwauth process sends the source IP address, username, domain, and other information to the USERID process, which verifies that it is a valid domain user entry. The subsequent traffic will hit this user firewall entry.

NOTE: The Active Directory authentication entry that comes from the fwauth process is not subject to the IP filters.

Related Documentation

- *user-firewall*
- Overview of Integrated User Firewall on page 141
- Understanding How the WMIC Reads the Event Log on the Domain Controller on page 169

Understanding Integrated User Firewall Domain PC Probing

This topic includes the following sections:

Overview of Domain PC Probing

At a high level, the integrated user firewall feature gathers IP address, user, and group information from Windows Active Directory domain controller event logs and LDAP services. This information is used to generate Active Directory authentication table entries on an SRX Series device. Authentication entries serve as the authentication source for security policies that enforce user-based or group-based access control.

PC probing acts as a supplement of event log reading. When a user logs in to the domain, the event log contains that information. The PC probe is triggered only when there is no IP-to-address mapping from the event log.

Domain information constantly changes as users log in and out of domain PCs. The integrated user firewall probe functionality provides a mechanism for tracking and verifying information in the authentication tables by directly probing domain PCs for IP address-to-user mapping information. New and changed information identified by the probe serves to update Active Directory authentication table entries, which is critical to maintaining firewall integrity.

The IP address filter also impacts the PC probe. Once you configure the IP address filter, only the IP address specified in the filter is probed.

Probing Domain PCs for User Information

The integrated user firewall feature tracks the online status of users by probing domain PCs. If a user is not online or is not an expected user, the Active Directory authentication table is updated as appropriate. The following probe behaviors apply:

On-demand probing—On-demand probing occurs when a packet is dropped due to a missing entry in the Active Directory authentication table. In this case, an entry is added in pending state to the authentication table, and the domain PC identified by the source IP field of the dropped packet is probed for IP address and user information. The entry remains in pending state until a response is received from the probe.

Manual probing—Manual probing is used to verify and troubleshoot the online status of a user or a range of users, and is at the discretion of the system administrator. To initiate a manual probe, use the **request services user-identification active-directory-access ip-user-probe address ip-address** *address* **domain** *domain-name* command. If a domain name is not specified, the probe looks at the first configured domain for the IP address. To specify a range, use the appropriate network address.

NOTE: Manual probing can cause entries to be removed from the Active Directory authentication table. For example, if there is no response from your PC due to a network issue, such as when the PC is too busy, the IP address entry of the PC is marked as *invalid* and your access is blocked.

If the SRX Series device cannot access a domain PC for some reason, such as a network configuration or Windows firewall issue, the probe fails.

Probe Response

Based on the domain PC probe response, updates are made to the Active Directory authentication table, and associated firewall policies take effect. If no response is received from the probe after 90 seconds, the authentication entry times out. The timed-out authentication entry is the pending state authentication entry, which is generated when you start the PC probe.

If the probe is successful, the state of the authentication entry is updated from pending to valid. If the probe is unsuccessful, the state of the authentication entry is marked as invalid. The invalid entry has the same lifetime as a valid entry and is overwritten by upcoming fwauth (firewall authentication process) authentication results or by the event log. Table 12 on page 173 lists probe responses and corresponding authentication table actions.

Table 12: Probe Responses and Associated Active Directory Authentication Table Actions

Probe Response from Domain PC	Active Directory Authentication Table Action
Valid IP address and username	Add IP-related entry.
Logged on user changed	Update IP-related entry.
Connection timeout	Update IP-related entry as invalid.
Access denied	Update IP-related entry as invalid.
Connection refused	Update IP-related entry as invalid.
Authentication failed (The configured username and password have no privilege to probe the domain PC.)	Update IP-related entry as invalid.

Probe Configuration

On-demand probing is enabled by default. To disable on-demand probing, use the **set services user-identification active-directory-access no-on-demand-probe** statement. Delete this statement to reenable probing. When on-demand probing is disabled, manual probing is available.

The probe timeout value is configurable. The default timeout is 10 seconds. To configure the timeout value, use the following statement:

user@host# **set services user-identification active-directory-access wmi-timeout** ***seconds***

If no response is received from the domain PC within the **wmi-timeout** interval, the probe fails and the system either creates an invalid authentication entry or updates the existing authentication entry as invalid. If an authentication table entry already exists for the probed IP address, and no response is received from the domain PC within the **wmi-timeout** interval, the probe fails and that entry is deleted from the table.

NOTE: To probe domain PCs, you must configure the integrated user firewall feature with the username and password credentials. You do not necessarily need a username and password account for each PC; instead you could set up one administrator account with privileges to access information on multiple PCs.

Probe Rate and Statistics

The maximum probe rate for the integrated user firewall feature is set by default and cannot be changed. For high-end SRX Series devices, the probe rate is 600 times per minute. For branch SRX Series devices, the probe rate is 100 times per minute. Probe functionality supports 5000 users, or up to 10 percent of the total supported authentication entries, whichever is smaller. Supporting 10 percent means that at any time, the number of IP addresses waiting to be probed cannot exceed 10 percent. For more information about the number of supported Active Directory authentication table entries, see "Understanding Active Directory Authentication Tables" on page 144.

High-level statistics covering probe activity are available for the total number of probes and the number of failed probes. Table 12 on page 173 describes the reasons for probe failures. To display probe statistics, use the **show services user-identification active-directory-access statistics ip-user-probe** command.

```
user@host> show services user-identification active-directory-access statistics ip-user-probe
Domain: www.example1.net
    Total user probe number                     : 176116
    Failed user probe number                    : 916

Domain: www.example2.net
    Total user probe number                     : 17632
    Failed user probe number                    : 342
```

Related Documentation

- Overview of Integrated User Firewall on page 141
- Understanding How the WMIC Reads the Event Log on the Domain Controller on page 169
- Understanding Active Directory Authentication Tables on page 144

CHAPTER 18

Logging User Identity Information Based on Zones

Understanding How to Include User Identity Information in the Session Log File Based on the Source Zone

This topic covers the integrated user firewall feature that allows you to configure the system to write to the session log the user's identity by user name or group name without having to use the source identity (source-identity) tuple in the security policy. Knowing the user's identity by name, as written to the log, not just by the IP address of the user's device, gives you clearer visibility into their activity and allows you to resolve security problems faster and more easily. Relying on the source zone (from-zone) to trigger user identity logging rather than on the source identity widens the scope of users whose source identity is logged.

Typically, for each security policy, you must specify in the policy the source and destination IP addresses and the zones against which traffic is matched. You must also specify an application that the traffic is matched to. If traffic matches these criteria, then the security policy's action is applied to the traffic issued from the user's device. However, no user identity information is written to the session log.

Optionally, instead of relying exclusively on the IP address of the user's device to identify the source of the traffic, you can specify the user identity—that is, the user name or the group name—in the source-identity tuple of a security policy. This approach gives you greater control over resource access by narrowing down application of the security policy's actions to a single, identified user or a group of users, if other security policy matching conditions are met. However, use of the source-identity tuple constrains application of the policy to traffic from a single user or user group.

It may happen that you want the system to write to the session log the user identity for all users from whom traffic originated based on the zone to which they belong (from-zone). In this case, you do not want to narrow the traffic match and security policy

application to a single user or a user group, which configuring the source-identity tuple would do.

The zone-based user identity feature allows you to direct the system to write to the log user identity information for any user who belongs to a zone that is configured with the source-identity-log statement when that zone is used as the source zone in a matching security policy.

NOTE: For the source-identity-log feature to take effect, you must also configure logging of the session initialize (session-init) and session end (session-close) events as part of the security policy's actions.

Table 13 on page 178 identifies the platforms that support this feature.

Table 13: Supported Platforms

Supported SRX Series Device Platforms
SRX300, SRX320, SRX340, SRX345
SRX550 High Memory (HM)
SRX1500
SRX4100 and SRX4200
SRX5400, SRX5600, SRX5800

- SRX5000 series devices.
- SRX320 device.
- SRX1500 series devices.
- SRX4000 series devices.
- SRX5000 series devices.
- vSRX

Related Documentation

- Example: Configuring Integrated User Firewall to Write User Identity to the Session Log Based On the Source Zone on page 179
- **source-identity-log (Security) on page 397**
- Overview of Integrated User Firewall on page 141
- Example: Configuring Integrated User Firewall on page 153

Example: Configuring Integrated User Firewall to Write User Identity to the Session Log Based On the Source Zone

This example shows how to configure the integrated user firewall zone-based user identity feature that directs the system to log user identity information based on the source zone (from-zone) configured in the security policy. The zone-based user identity feature widens the scope of users whose identity information is written to the log to include all users who belong to the zone whose traffic matches the security policy.

Requirements

This feature is supported starting with Junos OS 15.1X49-D60. You can configure and run this feature on any of the currently supported SRX Series devices beginning with Junos OS 15.1X49-D60.

Overview

This example shows how to configure integrated user firewall to log user identity information in the session log based on the source zone in the security policy. For this to occur, the zone specified as the source zone must be configured for source identity logging. For zone-based user identity logging, the security policy's actions must include session create (session-init) and session close (session-init) events.

When all conditions are met, the user's name is written to the log at the beginning of the session (or session initialization) and at the beginning of the close of the session (or session tear-down). Note that if a security policy denies the user access to the resource, an entry identifying the user by name is written to the log, that is, if session close is configured.

When you use the zone-based user identity feature, it is the source zone (from-zone) in the security policy that initiates the user identity logging event.

Prior to introduction of this feature, it was necessary to include the source identity tuple (source-identity) in a security policy to direct the system to write user identity information to the log—that is, the user name or the group name. The user identity was written to the log if the source-identity tuple was configured in any of the policies in a zone pair that matched the user's traffic and the session close log was configured.

However, the source identity feature is specific to an individual user or a group of users, and it constrains application of the security policy in that regard.

It is the user name that is stored in the local Active Directory table which the system writes to the log when the policy's source zone is configured for user-identity logging. The SRX Series device previously obtained the user identity information by reading the

domain controller event log. The SRX Series device stored that information in its Active Directory table.

You can use the source-identity tuple in a security policy that also specifies as its source zone a zone that was configured for user identity logging. Because integrated user firewall collects the names of the groups that a user belongs to from Microsoft Domain Controllers only when integrated user firewall relies on the source identity tuple, if you use the zone-based user identity logging feature without also configuring source-identity, the log will contain only the name of the user requesting access and not the groups that the user belongs to.

After you configure a zone to support source identity logging, the zone is reusable as the from-zone specification in any security policy for which you want user identity information logged.

To summarize, the user's name is written to the log if:

- The user belongs to the zone configured for source identity logging.
- The user Issues a resource access request whose generated traffic matches a security policy whose source zone (from-zone) tuple specifies a qualifying zone.
- The security policy includes as part of its actions the session initialize (session-init) and session end (session-close) events.

The source identity log function benefits include the ability to:

- Cover a wide range of users in a single specification—that is, all users who belong to a zone that is configured for source identity logging.
- Continue to use an address range for the source address in a security policy without forfeiting user identity logging.
- Reuse a zone that is configured for source identity logging in more than one security policy.

  Because it is configured independent of the security policy, you can specify the zone as the source zone in one or more policies.

NOTE: **The user identity is not logged if you specify a zone configured for zone-based user identity logging as the destination zone rather than as the source zone.**

For this function to work, you must configure the following information:

- The source identity log statement configured for a zone that is used as the source zone (from-zone) in the intended security policy.
- A security policy that specifies:
  - A qualifying zone as its source zone.
  - The session-init and the session-close events as part of its actions.

Configuration

To configure the source identity logging feature, perform these tasks:

- Configuring a Zone to Support Source Identity Logging and Using It in a Security Policy on page 181
- Results on page 182

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security zones security-zone trust source-identity-log
set security policies from-zone trust to-zone untrust policy appfw-policy1 match
    source-address any destination-address any application junos-ftp
set security policies from-zone trust to-zone untrust policy appfw-policy1 then permit
set security policies from-zone trust to-zone untrust policy appfw-policy1 then log
    session-init
set security policies from-zone trust to-zone untrust policy appfw-policy1 then log
    session-close
```

Configuring a Zone to Support Source Identity Logging and Using It in a Security Policy

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

1. Configure source identity logging for the trust zone. When this zone is used as the source zone in a security policy, the system writes the user identity information to the session log for all users to whom the security policy applies.

   ```
   [edit security]
   user@host# set zones security-zone trust source-identity-log
   ```

2. Configure a security policy called appfw-policy1 that specifies the zone trust as the term for its source zone. Source identity logging is applied to any user whose traffic matches the security policy's tuples.

   This security policy allows the user to access the junos-ftp service. When the session is established for the user, the user's identity is logged. It is also logged at the close of the session.

   ```
   [edit security]
   user@host# set policies from-zone trust to-zone untrust policy appfw-policy1 match
       source-address any destination-address any application junos-ftp
   user@host#  set policies from-zone trust to-zone untrust policy appfw-policy1 then
       permit
   ```

3. Configure the appfw-policy1 security policy's actions to include logging of the session initiation and session close events.

NOTE: You must configure the security policy to log session initiation and session close events for the source identity log function to take effect. The user identity information is written to the log in conjunction with these events.

```
[edit security]
user@host# set policies from-zone trust to-zone untrust policy appfw-policy1 then
    log session-init
user@host# set policies from-zone trust to-zone untrust policy appfw-policy1 then
    log session-close
```

Results

From configuration mode, confirm your configuration by entering the **show security zones** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Verification

This section shows the session log generated for the user session. The log output:

- Shows the user name, user1, which appears at the outset of session open and then again at the outset of session close.

  The security policy configuration that caused the user name to be written to the log specifies the zone trust as its source zone. The zone trust was configured for source identity logging.

- Includes information obtained from the user's request traffic, the policy matching criteria, and the NAT setup.

- Contains identity information about the user, which is obtained from the Active Directory database. That information includes the role parameter for "MyCompany/Administrator", which shows the groups that the user belongs to.

In this scenario, the user requested access to the Juniper Networks junos-ftp service, which the log also records. Table 14 on page 182 calls out the parts of the log that are specific to the source identity log function configuration:

Table 14: Session Log Components Specific to the Source Identity Log Function

session create This is the session initiation which begins the first section of the log that records the session setup information. The user's name, user1, is displayed at the beginning of the session create log recording.	user1 RT_FLOW_SESSION_CREATE
Session create is followed by standard information that defines the session based on the user's traffic that matches security policy tuples.	

Table 14: Session Log Components Specific to the Source Identity Log Function *(continued)*

source address, the source port, the destination address, the destination port.	source-address="198.51.100.13/24" source-port="635" destination-address="198.51.100.10/24" destination-port="51"
application service This is the application service that the user requested access to and which the security policy permitted.	service-name="junos-ftp"
source zone, destination zone Further down the log are the zone specifications which show trust as the source zone and untrust as the destination zone as defined.	source-zone-name="trust" destination-zone-name="untrust"
session close This is the session close initiation, which begins the second part of the log record that covers session tear-down and close. The user's name, user1, is displayed at the beginning of the session close record.	user1 RT_FLOW - RT_FLOW_SESSION_CLOSE

- Verify that the User Identity Information Was Logged on page 183

Verify that the User Identity Information Was Logged

Purpose Note that integrated user firewall collects groups configured as the source-identity only from Microsoft Domain Controllers. If you use the zone-based user-identity feature without configuring source-identity, the log will contain only the user's name, that is, no group informations is recorded. In that case, the "roles=" section of the log wll show "N/A". In the following example, it is assumed that the source-identity tuple was used and the "roles=" section shows a long list of the groups that the user "Administrator" belongs to.

Action Display the log information.

Sample Output

```
<14>1 2015-01-19T15:03:40.482+08:00 user1 RT_FLOW - RT_FLOW_SESSION_CREATE
[user@host2636 192.0.2.123 source-address="198.51.100.13“ source-port="635"
destination-address="198.51.100.10" destination-port="51" service-name="junos-ftp"
 nat-source-address="203.0.113.10" nat-source-port="12349" nat-destination-address
 ="198.51.100.13" nat-destination-port="3522" nat-rule-name="None"
dst-nat-rule-name="None" protocol-id="6" policy-name="appfw-policy1"
source-zone-name="trust" destination-zone-name="untrust" session-id-22="12245"
username="MyCompany/Administrator " roles="administrators, Users, Enterprise
Admins, Schema Admins, ad, Domain Users, Group Policy Creator Owners, example-team,
 Domain Admins" packet-incoming-interface="ge-0/0/0.1" application="UNKNOWN"
nested-application="UNKNOWN" encrypted="UNKNOWN"] session created 192.0.2.1/21
junos-ftp 10.1.1.12/32898->10.3.1.10/21 junos-ftp 10.1.1.1/547798->10.1.2.10/21
None None 6 appfw-policy1 trust untrust 20000025 MyCompany/Administrator
(administrators, Users, Enterprise Admins, Schema Admins, ad, Domain Users, Group
 Policy Creator Ownersexample-team, Domain Admins) ge-0/0/0.0 UNKNOWN UNKNOWN
UNKNOWN
```

```
<14>1 2015-01-19T15:03:59.427+08:00 user1 RT_FLOW - RT_FLOW_SESSION_CLOSE
[user@host2636 192.0.2.123 reason="idle Timeout” source-address=”198.51.100.13“
source-port=”635”" destination-address=”198.51.100.10” destination-port=”51"
service-name="junos-ftp" nat-source-address="203.0.113.10" nat-source-port="12349"
  nat-destination-address ="198.51.100.13" "nat-destination-port="3522"
src-nat-rule-name="None" dst-nat-rule-name="None" protocol-id="6"
policy-name="appfw-policy1" source-zone-name="trust"
destination-zone-name="untrust"session-id-32="20000025" packets-from-client="3"
bytes-from-client="180"
packets-from-server="0" bytes-from-server="0" elapsed-time="19"
application="INCONCLUSIVE" nested-application="INCONCLUSIVE" username=" J
“MyCompany /Administrator” roles="administrators, Users, Enterprise Admins,
Schema Admins, ad, Domain Users, Group Policy Creator Owners, example-team,
Domain Admins" packet-incoming-interface="ge-0/0/0.1" encrypted="UNKNOWN"]
session closed idle Timeout: 111.1.1.10/1234>10.1.1.11/21 junos-ftp
10.1.1.12/32898->10.3.1.10/21 1 None None 6 appfw-policy1 trust untrust 20000025
 3(180) 0(0) 19
INCONCLUSIVE INCONCLUSIVE MyCompany/Administrator (administrators, Users,
Enterprise Admins, Schema Admins, ad, Domain Users, Group Policy Creator Owners,
 example-team, Domain Admins) ge-0/0/0.1 UNKNOWN
```

Related Documentation

- Understanding How to Include User Identity Information in the Session Log File Based on the Source Zone on page 177
- source-identity-log (Security) on page 397
- Overview of Integrated User Firewall on page 141
- Example: Configuring Integrated User Firewall on page 153

CHAPTER 19

Configuring Integrated User Firewall Device Identity Authentication for Access Control

Understanding Access Control to Network Resources Based on Device Identity Information

You can use the integrated user firewall device identity authentication feature to control access to network resources based on the attributes, or characteristics, of the device used. After you configure device identity authentication feature, you can configure security polices that allow or deny traffic from the identified device based on the policy action.

Why Use Device Identity Information to Control Access to Your Network

For various reasons, you might want to control access to your network resources based on the identity of the user's device rather than on the identity of the user. For example, you might not know the identity of the user. You might allow your users to use their own devices (BYOD) to access network resources and you do not want to use captive portal

authenticate. Some companies might have older switches that do not support 802.1, or they might not have a network access Control (NAC) system. The integrated user firewall device identity authentication feature was designed to offer a solution to these and other similar situations by enabling you to control network access based on attributes of the user's device.

Background

Fundamentally, the Juniper Networks SRX Series Services Gateway device receives or obtains the device identity information from the authentication source in the same manner that it obtains the user identity information, depending on the authentication source. If Microsoft Windows Active Directory is the authentication source, the SRX Series device retrieves the device information from the Active Directory domain controller. In the case of third-party Network Access Control (NAC) systems, the SRX Series device receives the information from the authentication source through the RESTful Web services API that the SRX Series device exposes to it for this purpose. After the SRX Series device obtains the device identity information, it creates an entry for it in the device identity authentication table.

The purpose of obtaining the device information and entering it into the device identity authentication table is to control user access to network resources based on the device's identity. For this to occur, you must also configure security policies that identify the device, based on the specified device identity profile, and specify the action to be taken on traffic that issues from that device. That process is covered elsewhere.

In broad terms, the process in which the device identity information is obtained and stored in the device identity information table entails the following actions on the part of the SRX Series device:

- Getting the device identity information.

  Depending on the authentication source, the SRX Series device uses one of the following two methods to obtain the device identity information:

  - Active Directory—For Active Directory, the SRX Series device can extract the device information from the domain controller's event log and then connect to the Active Directory LDAP server to obtain the names of the groups that the device belongs to. The SRX Series device uses the information that it obtained from the event log to locate the device's information in the LDAP directory.
  - Third-party NAC systems—These authentication systems use the POST service of the RESTful Web services API, called Web API. The SRX Series device implements the API and exposes to the authentication systems to allow them to send the device identity information to the SRX Series device.

    The API has a formal XML structure and restrictions that the authentication source must adhere to in sending this information to the SRX Series device.

- Creating an entry for the device in the device identity authentication table.

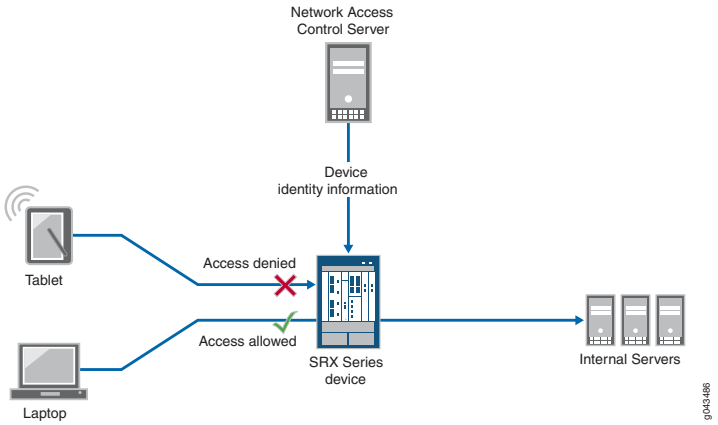
  After the SRX Series device obtains the device identity information, it creates an entry for it in the device identity authentication table. The device identify authentication table is separate from the Active Directory authentication table or any of the other

local authentication tables used for third party authentication sources. Too, unlike local user authentication tables which are particular to an authentication source or feature, the device identity authentication table holds device identity information for all authentication sources. However, only one authentication source, such as Active Directory, can be active at a time. The SRX Series device allows only authentication source to be used at a time to constrain the demand on the system to process information.

NOTE: The device identity authentication feature supports various types of authentication systems, such as Active Directory or a third-party authentication source. That is, the device identity authentication feature provides a generic solution that stores device identity information in the same table regardless of the authentication source.

Figure 18 on page 187 shows the communication between the SRX Series and a third-party NAC authentication source that is used for device identity authentication. The SRX Series device receives the device identity information from the NAC system and stores it in its local device identity authentication table. A security policy that specifies a device identity profile is applicable to one or more devices. If a device matches the device identity profile and other parts of the security policy, the security policy is applied to traffic issuing from that device.

Figure 18: Using a Third-Party Network Access Control (NAC) System for Device Identity Authentication



Related Documentation

- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192
- Understanding How the SRX Series Obtains the Authenticated Device Identity Information From Windows Active Directory for Network Access Control on page 198
- Understanding the Device Identity XML Solution for Third-Party NAC Authentication Systems on page 199

- Example: Configuring the SRX Series Device Identity Feature in an Active Directory Environment on page 206

Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature

The *device identity profile*, referred to in the CLI as the ***end-user-profile***, is a key component of the integrated user firewall device identity authentication feature. It identifies the device and specifies its attributes. The device identity authentication feature allows you to control access to your network resources based on the identity of the device used and not the identity of the user of that device. This feature supports Microsoft Windows Active Directory and third-party network access control (NAC) systems as authentication sources.

This topic focuses on device identity and the device identity profile.

- Device Identity on page 188
- Device Identity Profile Contents on page 188
- Predefined Device Identity Attributes on page 190
- Characteristics of Device Identity Profiles, and Attributes and Target Scaling on page 190

Device Identity

The device identity essentially consists of the IP address of the device, its name, its domain, and the groups that the device belongs to.

For example, the following output shows information about the device, which is referred to from the device identity profile.

This example shows that the device identity authentication table contains entries for two devices. For each entry, it shows the IP address of the device, the name assigned to the device, and the groups that the device belongs to. Note that both devices belong to the group grp4.

```
Source IP       Device ID      Device-Groups
192.0.2.1       lab-computer1   grp1, grp3, grp4
198.51.100.1    dev-computer2   grp5, grp6, grp4
```

Device Identity Profile Contents

The device identity profile is a collection of attributes that are characteristics of a specific group of devices, or of a specific device, depending on the attributes configured in the profile. The Packet Forwarding Engine of the SRX Series device maps the IP address of a device to the device identity profile.

A device identity profile specifies the name of the device and information that includes the IP address of the device, groups to which the device belongs, and attributes of the device which are collectively referred to as the host attributes.

When traffic from a device arrives at the SRX Series, the SRX Series obtains the IP address of the device from the first packet of the traffic and uses it to search the device identity authentication table for a matching device identity entry. Then it matches that device identity profile with a security policy whose **source-end-user-profile** field specifies the device identity profile name. If a match is found, the security policy is applied to traffic issuing from the device.

The same device identity profile can also apply to other devices sharing the same attributes. However, for the same security policy to apply, the device and its traffic must match all other fields in the security policy.

A device identity profile must contain the domain name. It might contain more than one set of attributes, but it must contain at least one. Consider the following two sets of attributes that belong to the profile called marketing-main-alice.

The profile contains the following set of attributes:

- alice-T430, as the name of the device.
- MARKETING and WEST-COAST, as the groups that the device belongs to.
- example.net as the name of the domain that it belongs to.

The profile also the following attributes that characterize the device:

- laptop, as the category of the device (device-category)
- Lenovo, as the device vendor (device-vendor)
- ThinkPad T430, as the type of device (device-type)

In cases such as the marketing-main-alice profile that includes the name given to the device, the profile applies exclusively to that device.

However, now suppose that another profile called marketing-west-coast-T430 was configured and that it contains the same attributes as the marketing-main-alice profile with one exception: the name given to the device in the marketing-main-alice profile was not included as an attribute in the marketing-west-coast-T430 profile. In this case, the attributes contained in the profile now make up a group profile. Application of the profile is widened to include all Lenovo ThinkPad T430 devices (which are laptops) that fit the rest of the characteristics, or attributes, defined in the profile.

Devices are covered by the profile if all other attributes match: devices that belong to either the MARKETING or WEST-COAST groups, which the marketing-west-coast-T430 profile specifies as its groups, or to both groups, match the profile.

As mentioned previously, a device identity profile can contain more than one group. A device can also belong to more than one group.

To illustrate further, note that the group device identity profile called marketing-west-coast-T430 also applies to the device called alice-T430 because that device belongs to both the MARKETING and the WEST-COAST groups and it matches all other attributes defined in the profile. Of course, the marketing-main-alice device identity profile still applies to the device called alice-T430. Therefore, the device called

alice-T430 belongs to at least two groups, and it is covered by at least two device identity profiles.

Suppose that another profile called marketing-human-resources was defined with all of the attributes of the marketing-west-coast-T430 device identity profile but with these differences: the new device identity profile includes a group called HUMAN-RESOURCES and it does not include the group called WEST-COAST. However, it does contain the MARKETING group.

Because the device called alice-T430 belongs to the MARKETING group, which remains as a group in marketing-human-resources profile, the alice-T430 device also matches the marketing-human-resources device identity profile. Now the alice-T430 device matches three profiles. If the names of any of these profiles is specified in a security policy's source-end-user-profile and the alice-T430 device matches all of the other fields in the security profile, then that profile's action is applied to traffic from that device.

The previous examples of device identity profiles illustrate the following points:

- A profile can be defined to identify only one device or it can be defined to apply to many devices.
- A device identity profile can contain more than one group to which a given device belongs.
- A device can match more than one device identity profile by matching the characteristics, or attributes, including at least one of the groups, configured for the profile.

The flexible use of device identity profiles will become evident when you configure security policies that are designed to include the source-end-user-profile field, in particular when you want the policy's action to be applied to a number of devices.

Predefined Device Identity Attributes

The SRX Series device provides the following predefined device identity policy attributes that you can configure to characterize a device:

- device-identity
- device-category
- device-vendor
- device-type
- device-os
- device-os-version

You specify values for these attributes in a device identity profile.

Characteristics of Device Identity Profiles, and Attributes and Target Scaling

This section describes how the SRX Series treats device identity attributes and profiles. It also gives SRX Series device-independent and device-dependent scaling numbers for these entities.

The following attribute and profile characteristics apply to their use on all supported SRX Series devices.

- The maximum length of the following entities is 64 bytes: device identity profile names (referred to in the CLI as **end-user-profile**) attribute names, attribute-values.
- You can not overlap values in a range if you configure more than one digital value range for the same attribute.
- When the SRX Series device matches a device identity profile to a security policy, all of the attributes in the profile are taken into account. Here is how they are treated:
  - If the device identity profile contains multiple values for an attribute, the values of that attribute are treated individually. It is said that they are ORed.

    For the security policy to be applied to the device, the following conditions must be met. The device must match:

    - One of the values for each attribute that has multiple values.
    - The rest of the attribute values specified in the device identity profile.
    - The security policy field values.
  - All individual attributes that have a single value are treated separately and considered together as a collection of values—that is, the AND operation is applied to them. The SRX Series device uses its standard policy-matching criteria in handling these attributes.

Table 15 on page 191 shows the platform-independent scaling values used in the device identity authentication feature.

Table 15: Platform-Independent Scaling

Item	Maximum
Values per attribute	20
Attributes per profile	100
Device identity profile specification per security policy (source-end-user-profile)	1

Table 16 on page 191 shows the platform-dependent scaling values used in the device identity authentication feature..

Table 16: Platform-Dependent Scaling

Platform	Maximum Number of Profiles	Maximum Total Number of Attribute Values
SRX5000 Series	4000	32000
SRX Series 1500	1000	8000

Table 16: Platform-Dependent Scaling *(continued)*

Platform	Maximum Number of Profiles	Maximum Total Number of Attribute Values
SRX Series 550M	500	4000
SRX Series 300 and SRX Series 320	100	1000
SRX Series 340 and SRX Series 345	100	1000
SRX Series 4100-4XE	1000	8000
SRX Series 4200-8XE	2000	16000
vSRX	500	4000

The following changes to device identity profiles and their use in security policies do not cause the SRX Series device to perform a session scan:

- Updates to a profile which is referenced in a security policy.
- Updates to the profile configuration.
- Updates to attributes that are made through the RESTful web services API, which is is used by NAC systems, or Active Directory LDAP.

Understanding the Device Identity Authentication Table and Its Entries

The SRX Series device contains a number of local authentication tables used for user authentication for various purposes. For example, the SRX Series device contains a local Active Directory authentication table for user authentication when Microsoft Windows Active Directory is used as the authentication source.

When you configure the SRX Series device to use the integrated user firewall device identity authentication feature for authentication based on the device identity and its attributes, the SRX Series device creates a new table called the device identity authentication table.

To gain a complete view of the device identity authentication feature, it helps to understand this table, its contents, and its relationship to other entities.

This topic covers the device identity authentication table and its device entries, and how the table contents change based on several factors.

- The Device Identity Authentication Table on page 193
- Why the Device Identity Authentication Table Content Changes on page 193
- [xref target has no title]

The Device Identity Authentication Table

Unlike other local authentication tables, the device identity authentication table does not contain information about a user but rather about the user's device. Moreover, unlike user authentication tables, it does not contain information about devices authenticated by one authentication source. Rather, it serves as a repository for device identity information for all devices regardless of their authentication source. For example, it might contain entries for devices authenticated by Active Directory or third-party NAC authentication sources.

A device identity authentication table entry contains the following parts:

- The IP address of the device.
- The name of the domain that the device belongs to.
- The groups with which the device is associated.
- The device identity.

  The device identity is actually that of a device identity profile (referred to in the CLI as **end-user-profile**). This type of profile contains a group of attributes that characterize a specific individual device or a specific group of devices, for example, a specific type of laptop.

Why the Device Identity Authentication Table Content Changes

The device identity entries in the device identity authentication table are changed when certain events occur: when the user authentication entry with which the device identity entry is associated expires, when security policy changes occur in regard to referencing a group that the device belongs to, when the device is added to or removed from groups, or when groups that it belongs to are deleted and that change is made to the Windows Active Directory LDAP server.

- When the User Identity Entry with Which a Device Identity Entry Is Associated Expires

  When the SRX Series device generates an entry for a device in the device identity authentication table, it associates that entry with a user identity entry in a local authentication table for the specific authentication source that authenticated the user of the device, such as Active Directory. That is, it ties the device identity entry in the device identity authentication table to the entry for the user of the device in the user authentication table.

  When the user authentication entry with which the device identity entry is associated expires and is deleted from the user authentication table, the device identity entry is deleted silently from the device identity authentication table. That is, no message is issued to inform you of this event.

- When Security Policy Changes Occur in Regard to Referencing a Group to Which the Device Belongs

  To control access to network resources based on device identity, you create a device identity profile that you can refer to in a security policy. In addition to other attributes, a device identity profile contains the names of groups. When a device identity profile

is referenced by a security policy, the groups that it contains are referred to as *interested groups*.

A group qualifies as an *interested group* if it is referenced by a security policy—that is, if it is included in a device identity profile which is specified in the **source-end-user-devic**e field of a security policy. If a group is included in a device identity profile that is not currently used in a security policy, it is not included in the list of interested groups. A group can move in and out of the list of groups referenced by security policies.

- When a Device Is Added to Or Removed From a Group or a Group Is Deleted

  To keep the device identity entries in the local device identity authentication table current, the SRX Series monitors the Active Directory event log for changes. In addition to determining whether a device had logged out or in to the network, it can determine changes to any groups that the device might belong to. When changes occur to the groups that a device belongs to—that is, when a device is added to or removed from a group or the group is deleted—the SRX Series device modifies the contents of the affected device entries in its own device identity authentication table to reflect the changes made in the Microsoft Windows Active Directory LDAP server.

The SRX Series device identity authentication table is updated according to changes to groups with which the device is associated in the LDAP server, as illustrated in Table 17 on page 194.

Table 17: Group Changes for Devices in the Active Directory LDAP and the SRX Series Response

Changes Made to LDAP	SRX Series LDAP Message and UserID Daemon Action
Group information for a device has changed. The device has been added to or removed from a group, or a group that the device belongs to has been deleted.	The Active Directory LDAP module sends notification of the change to the SRX Series UserID daemon, directing it to revise information in its local device identity authentication table. The SRX Series device processes these messages every 2 minutes.
The device entry in LDAP is deleted.	The Active Directory LDAP module sends notification of the change to the UserID daemon, directing it to revise information in its local device identity authentication table. The SRX Series device processes these messages every 2 minutes.

The SRX Series UserID daemon is informed of the changes. Whether or not a group that a device belongs to is specified in a security policy has bearing on what information is stored in device identity authentication table entries for the affected device. Table 18 on page 195 shows the activity that occurs when a group is added to or deleted from the Active Directory LDAP.

Table 18: Changes to Device Identity Entries Based on Security Policy Specifications

Device Identity Profile Changes	Device-Group Mapping Behavior	SRX Series UserID Daemon Response
A new group that was added to the Active Directory LDAP is added to the SRX Series device identity profile.	The SRX Series gets the list of devices that belong to the new group and its subgroups from the Active Directory LDAP server. It adds the list to its local LDAP directory.	The UserID daemon determines whether the device identity authentication table includes entries for the set of affected devices. If so, it updates the group information for these entries. For example, here is the entry for device1 before it was updated to include the new group and after the group was added: • device1, g1 • device1, g1, g2
A group is deleted from the Active Directory LDAP. The SRX Series deletes the group from the device identity profile.	The SRX Series gets the list of devices that belong to the deleted group from its local LDAP database. It deletes the device-group mapping from the local LDAP directory.	The UserID daemon checks the device identity authentication table for entries that belong to the group. It removes the group from affected entries. For example, here is the entry for device1 before the group was deleted and after the group was deleted: • device1, g1, g2 • device1, g1

Table 19 on page 195 elaborates on the contents of device authentication entries for several devices which are affected by deletion of a group.

Table 19: Changes to Device Identity Authentication Table Resulting From LDAP and Security Policy Changes

Changes to Device identity Authentication Table Entries		
IP Address	**Device Information**	**Group**
Original Entries		
192.0.2.10	device1	group1, group2
192.0.2.11	device2	group3, group4
192.0.2.12	device3	group2
Same Entries After group2 Is Deleted		
192.0.2.10	device1	group1
192.0.2.11	device2	group3, group4
192.0.2.12	device3	*This entry no longer contains groups.*

Security Policy Matching and Device Identity Profiles

The SRX Series device follows the standard rules for matching traffic against security policies. The following behavior pertains to the use of a device identity profile in a security policy for determining a match:

- Use of a device identity profile in a security policy is optional.
  - If no device identity profile is specified in the source-end-user-profile field, **any** profile is assumed.
  - You cannot use the keyword **any** in the **source-end-user-profile** field of a security policy.

    If you use the source-end-user-profile field in a security policy, you must reference a specific profile. The device from which the access attempt is issued must match the profile's attributes.
- Only one device identity profile can be specified in a single security policy.
- A security policy rematch is triggered when the **source-end-user-profile** field value of the security policy is changed. No rematch is triggered when an attribute value of a profile is changed.

Security Policy Matching and Device Identity Profiles

The SRX Series device follows the standard rules for matching traffic against security policies. The following behavior pertains to the use of a device identity profile in a security policy for determining a match:

- Use of a device identity profile in a security policy is optional.
  - If no device identity profile is specified in the source-end-user-profile field, **any** profile is assumed.
  - You cannot use the keyword **any** in the **source-end-user-profile** field of a security policy.

    If you use the source-end-user-profile field in a security policy, you must reference a specific profile. The device from which the access attempt is issued must match the profile's attributes.
- Only one device identity profile can be specified in a single security policy.
- A security policy rematch is triggered when the **source-end-user-profile** field value of the security policy is changed. No rematch is triggered when an attribute value of a profile is changed.

Related Documentation

- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding Access Control to Network Resources Based on Device Identity Information on page 185

- Example: Configuring the SRX Series Device Identity Feature in an Active Directory Environment on page 206

Understanding Security Policy Matching with Device Identity Profiles

pre-defined attributes:

- device-category
- device-vendor
- device-type
- device-os
- device-os-version

Policy Match Criteria

- Use of a device identity profile in a security policy is optional.

  If no device identity profile is specified, “any” profile is assumed.

  You can not use the keyword “any” in the source-end-user-profile field of a security policy. It is a reserved keyword.
- Only one device identity policy can be specified in a single security policy.
- A security policy rematch is triggered when the source-end-user-profile field value of a security policy is changed. No rematch is triggered when an attribute value of a profile is changed.

Related Documentation

-

Understanding How the SRX Series Obtains the Authenticated Device Identity Information From Windows Active Directory for Network Access Control

You can use the integrated user firewall device identity authentication feature to control access to your network resources based on the identity and attributes of the device used rather than the user identity. Information about a device is stored in the device identity authentication table. You can specify the name of a device identity profile that contains the device attributes in the source-end-user-profile field of a security policy. If all conditions are met, the security policy's actions are applied to traffic issuing from that device.

For you to be able to use device identity profiles in security policies, the SRX Series device must obtain the device identity information for authenticated devices. The SRX Series device creates the device identity authentication table to use to store device identity entries. It searches the table for a device match when traffic arrives from a device. This topic considers the process followed when Active Directory is used as the authentication source.

An Active Directory domain controller authenticates users when they log in to the domain, and it writes a record of that event to the Windows event log. It also writes a record to the event log when a user logs out of the domain. The domain controller event log provides the SRX Series device with information about authenticated devices that are currently active in the domain and when those devices are logged out from it.

The SRX Series UserID daemon takes the following actions:

1. It reads the Active Directory domain controller event logs to obtain the IP addresses of devices logged into the domain and authenticated by Windows.

   The UserID daemon in the SRX Series Routing Engine implements a Windows Management Instrumentation (WMI) client with Microsoft Distributed COM/Microsoft RPC stacks and an authentication mechanism to communicate with a Windows Active Directory domain controller in an Active Directory domain. Using event log information retrieved from the Active Directory controller, the process obtains the IP addresses of active Active Directory devices. The process monitors Active Directory event log changes using the same WMI DCOM interface to adjust its device identity information in its local authentication table to reflect any changes made to the Active Directory server.

2. It uses the device IP addresses that it obtained from the event log to obtain information about the groups that a device belongs to. To obtain this group information, the SRX Series device connects to the LDAP service in the Active Directory controller using the LDAP protocol for this purpose.

As a result of this process, the SRX Series is able to generate entries for the devices in the device identity authentication table. After it generates an entry for a device in the device identity authentication table, the SRX Series device associates that entry with the appropriate user entry in its local Active Directory authentication table. You can then reference the device identity profile entries in security policies to control access to resources.

Behavior and Constraints

- The SRX Series process of reading the event log consumes domain controller CPU resources which may lead to high CPU usage in the domain controller. For this reason, the Active Directory domain controller should have a high-performance configuration of at least 4 cores and 8 gigabytes of memory.
- The domain controller event log records a maximum length of 16 bytes of the device ID, including a null terminator. Therefore, the maximum length of the device ID that the SRX Series device obtains from the domain controller is 15 bytes.
- If the domain controller clears the event log or if the data written to the event log is missing or delayed, the device identity mapping information might be inaccurate. If the SRX Series device is unable to read the event log or if it contains null data, the SRX Series reports an error condition in its own log.
- If the device identity information table reaches capacity, it cannot add new device identity entries. In that case, traffic from the device is dropped.

Related Documentation

- Example: Configuring the SRX Series Device Identity Feature in an Active Directory Environment on page 206
- Understanding Access Control to Network Resources Based on Device Identity Information on page 185
- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192

Understanding the Device Identity XML Solution for Third-Party NAC Authentication Systems

The SRX Series integrated user firewall device identity authentication feature enables you to control access to network resources based on the identity of a device. You can use one of the following device identity solutions:

- Microsoft Active Directory as the authentication source.

  If your environment is set up to use Microsoft Active Directory, the SRX Series device obtains the device IP address and groups from the Active Directory domain controller and LDAP service.

- Network access control (NAC) authentication system.

  If your network environment is configured for a NAC solution and you decide to take this approach, the NAC system sends the device identity information to the SRX Series device. The RESTful Web services API enables you to send the device information to the SRX Series device in a formal XML structure.

WARNING: If you take this approach, you must verify that your NAC solution works with the SRX Series device.

The SRX Series XML Web API Implementation

The RESTful Web services API enables you to send the device identity information to the SRX Series device in a formal XML structure. It allows your NAC solution to integrate with the SRX Series and efficiently send the device information to it. You must adhere to the formal structure and restrictions in sending information to the SRX Series using the API.

Ensuring the Integrity of Data Sent from the NAC Service to the SRX Series

The following requirements ensure that the data sent from the NAC service is not compromised:

- The API implementation is restricted to processing only HTTP/HTTPS POST requests. Any other type of request that it receives generates an error message.
- The API daemon analyzes and processes HTTP/HTTPS requests from only the following dedicated URL:

  `/api/userfw/v1/post-entry`
- The HTTP/HTTPS content that your NAC solution posts to the SRX Series device must be consistently formatted correctly. The correct XML format indicates a lack of compromise, and it ensures that user identity information is not lost.

Data Size Restrictions and Other Constraints

The following data size restrictions and limitations apply to the data posted to the SRX Series device:

- The NAC authentication system must control the size of the data that it posts. Otherwise, the Web API daemon is unable to process it. The Web API daemon can process a maximum of 2 megabytes of data.
- The following limitations apply to XML data for role and device posture information. The Web API daemon discards XML data sent to it that exceeds these amounts (that is, the overflow data):
  - The SRX Series device can process a maximum of 209 roles.
  - The SRX Series device supports only one type of posture with six possible posture tokens, or values. Identity information for an individual user can have only one posture token.

Example: Configuring a Device Identity Profile to Control Network Access

This example shows how to configure the integrated user firewall device identity feature to control access to network resources based on the identity of the device and not its user. It covers how to configure a security policy that references it, including the zones and their interfaces that are used in the policy.

Requirements

This example uses the following hardware and software components:

- An SRX Series XXX???? device running Junos SRX 15.1X49-D70.
- Windows Active Directory with a Domain Controller and LDAP Server

  The Active Directory domain controller has a high performance configuration of four multiple cores and 8 gigabytes of memory.

  NOTE: The SRX Series obtains the IP address of a device by reading the domain controller event log. The process that reads the event log consumes domain controller CPU resources which may lead to high CPU usage. For this reason, the Active Directory domain controller should have a high performance configuration of at least four multiple cores and 8 gigabytes of memory.

- A server on the internal corporate network

Before you begin to configure this feature:

- Verify the version of Junos OS software that is installed on your SRX Series device. The integrated user firewall device identity feature is supported initially in Junos SRX 15.1X49-D70.
- Ensure that the SRX Series device is configured for access to an Active Directory domain controller. Ensure that it is configured for access to the Active Directory LDAP server that contains the groups that the devices belong to. For details, see TOPICS ON THIS.

Overview

Starting with Junos 15.1X49-D70, the SRX Series provides support for controlling access to network resources based on the identity of a device. You can configure a device identity profile that specifies attributes, or characteristics, of a device and refer to the profile from

a security policy. The security policy's action is applied to traffic issuing from any device that matches the profile attributes, in addition to the security policy parameters.

NOTE: The device identity profile is also referred to as the end-user-profile. The security policy parameter that allows you to specify its name is called *source-end-user-profile*.

This example supports Active Directory as the authentication source. The SRX Series device obtains the device identity information—its IP address—by reading the Active Directory domain controller's event log. The SRX Series device uses the device IP address to obtain from the Active Directory LDAP server the groups that the device belongs to.

This example shows how to configure a device identity profile, also referred to as an end-user-profile, and the security policy that refers to the profile. It also shows how to configure the zones to be used in the security policy and their interfaces.

Topology

Configuration

To configure the device identity feature, perform these tasks:

- Configuring Zones and Their Interfaces on page 203
- Results on page 204

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/3.1 family inet address 192.0.2.18/24
set interfaces ge-0/0/3.2 family inet address 192.0.2.14/24
set security zones security-zone marketing-zone interfaces ge-0/0/3.1 host-inbound-traffic
    system-services all
set security zones security-zone marketing-zone interfaces ge-0/0/3.1 host-inbound-traffic
    protocols all
set security zones security-zone servers-zone interfaces ge-0/0/3.2 host-inbound-traffic
    system-services all
set security zones security-zone servers-zone interfaces ge-0/0/3.2 host-inbound-traffic
    protocols all
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast domain-name ad-domain1
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute location string mountain-view
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-category string laptop
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-vendor string Lenovo
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-type string ThinkPad
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-os string Windows
```

```
set services user-identification device-information end-user-profile profile-name
  marketing-west-coast attribute device-os-version string 10.1
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match source-address any destination-address any
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match set application any
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match source-end-user-profile marketing-west-coast
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access application any
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access then permit
```

Configuring Zones and Their Interfaces

Step-by-Step Procedure

1. Configure interfaces, and their IP addresses, to be used for the marketing-zone and the servers-zone.

   ```
   [edit interfaces]
   user@host# set ge-0/0/3.1 family inet address 192.0.2.18/24
   user@host# set ge-0/0/3.2 family inet address 192.0.2.14/24
   ```

2. Configure the marketing-zone and the servers-zone and assign interfaces to them.

   ```
   [edit security zones]
   user@host#set security-zone marketing-zone interfaces ge-0/0/3.1
     host-inbound-traffic system-services all
   user@host#set security-zone marketing-zone interfaces ge-0/0/3.1
     host-inbound-traffic protocols all
   user@host#set security-zone servers-zone interfaces ge-0/0/3.2
     host-inbound-traffic system-services all
   user@host#set security-zone servers-zone interfaces ge-0/0/3.2
     host-inbound-traffic protocols all
   ```

3. Configure attributes for the device identity profile.

   ```
   [edit services user-identification]
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast domain-name ad-domain1
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute location string mountain-view
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute device-category string laptop
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute device-vendor string Lenovo
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute device-type string ThinkPad
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute device-os string Windows
   user@host# set device-information end-user-profile profile-name
     marketing-west-coast attribute device-os-version string 10.1
   ```

4. Configure a security policy that references the device identity profile.

   ```
   [edit security policies]
   user@host# set from-zone marketing-zone to-zone servers-zone policy
     mark-server-access match source-address any destination-address any
   user@host# set from-zone marketing-zone to-zone servers-zone policy
     mark-server-access match set application any
   ```

```
user@host#set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access match source-end-user-profile marketing-west-coast
user@host# set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access application any
user@host# set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access then permit
```

Results

show interfaces

```
ge-0/0/3 {
  unit 1 {
    family inet {
      address 192.0.2.18/24;
    }
  }
  unit 2 {
    family inet {
      address 192.0.2.14/24;
    }
  }
}
```

show security zones

```
security-zone marketing-zone {
  interfaces {
    ge-0/0/3.1 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
security-zone servers-zone {
  interfaces {
    ge-0/0/3.2 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
```

show services user-identification device-information end-user-profile

```
domain-name ad-domain1
attribute location {
   string mountain-view;
}
attribute device-identity {
   string marketing-clas;
}
attribute device-category {
   string laptop;
}
attribute device-vendor {
   string Lenovo;
}
attribute device-type {
   string ThinkPad;
}
attribute device-os {
   string Windows;
}
attribute device-os-version {
   string 10.1;
}
```

show security policies

```
from-zone marketing-zone to-zone servers-zone {
   policy mark-server-access {
      match {
         source-address any;
         destination-address any;
         application any;
         source-end-user-profile {
            marketing-west-coast;
         }
      }
      then {
         permit;
      }
   }
}
```

Verification

Verifying [item]

Purpose

Action show services user-identification device-information end-user-profile profile-name marketing-west-coast

```
domain-name ad-domain1;
attribute location {
    string mountain-view;
}
attribute device-category {
    string laptop;
}
attribute device-vendor {
    string [ Lenovo 10.0 ];
}
attribute device-type {
    string ThinkPad;
}
attribute device-os {
    string Windows;
}
```

Meaning

Troubleshooting

To troubleshoot [item], perform these tasks:

- Troubleshooting [item] on page 206

Troubleshooting [item]

Problem

Solution

Related Documentation

-

Example: Configuring the SRX Series Device Identity Feature in an Active Directory Environment

This example shows how to configure the integrated user firewall device identity authentication feature to control access to network resources based on the identity of an authenticated device, not its user. This example uses Microsoft Active Directory as the authentication source. It covers how to configure a device identity profile that characterizes a device, or set of devices, and how to reference that profile in a security policy. If a device matches the device identity profile and the security policy parameters, the security policy's action is applied to traffic issuing from that device.

For various reasons, you might want to use the identity, or attributes, of a device for resource access control. For example, you might not know the identity of the user. You might allow your users to use their own devices (BYOD) to access network resources and you do not want to use a captive portal to authenticate them. Also some companies might have older switches that do not support 802.1, or they might not have a network access control (NAC) system. The device identity authentication feature was designed

to offer a solution to these and other similar situations by enabling you to control network access based on attributes of the user's device. You can control access for a group of devices that fit the specification or an individual device.

Requirements

This example uses the following hardware and software components:

- An SRX Series Services Gateway device running Junos OS Release 15.1X49-D70.
- Microsoft Active Directory with a domain controller and the Lightweight Directory Access Protocol (LDAP) server

  The Active Directory domain controller has a high-performance configuration of 4 cores and 8 gigabytes of memory.

  NOTE: The SRX Series obtains the IP address of a device by reading the domain controller event log. The process that reads the event log consumes domain controller CPU resources, which might lead to high CPU usage. For this reason, the Active Directory domain controller should have a high-performance configuration of at least 4 cores and 8 gigabytes of memory.

- A server on the internal corporate network.

Overview

Starting with Junos OS Release 15.1X49-D70, the SRX Series provides support for controlling access to network resources based on the identity of a device authenticated by Active Directory or a third-party network access control (NAC) system. This example uses Active Directory as the authentication source.

NOTE: You must configure the authentication source for this feature to work.

This example covers the following configuration parts:

- Zones and their interfaces

  You must configure the zones to which the source and destination entities specified in the security policy belong. If you do not configure them, the security policy that references the device identity profile will be invalid.

- A device identity profile

You configure the device identity profile apart from the security policy; you refer to it from a security policy. A device identity profile specifies device attributes, or characteristics, that can be matched by one or more devices.

NOTE: The device identity profile is referred to as end-user-profile in the CLI.

- A security policy

  You configure a security policy whose action is applied to traffic issuing from any device that matches the device identity profile attributes and the rest of the security policy's parameters.

NOTE: You specify the name of the device identity profile in the security policy's *source-end-user-profile* field.

- Authentication source

  You configure the authentication source to be used to authenticate the device. This example uses Active Directory as the device identity authentication source.

  If Active Directory is the authentication source, the SRX Series obtains identity information for an authenticated device by reading the Active Directory domain's event log. The SRX Series then queries the LDAP interface of Active Directory to identify the groups that the device belongs to, using the device's IP address for the query.

  For this purpose, the SRX Series implements a Windows Management Instrumentation (WMI) client with Microsoft Distributed COM/Microsoft RPC stacks and an authentication mechanism to communicate with the Windows Active Directory controller in the Active Directory domain. It is the SRX Series wmic daemon that extracts device information from the event log of the Active Directory domain.

  The wmic daemon also monitors the Active Directory event log for changes by using the same WMI DCOM interface. When changes occur, the SRX Series adjusts its local device identity authentication table to reflect those changes.

Topology

In this example, users who belong to the marketing-zone zone want to access resources on the internal corporate servers. Access control is based on the identity of the device. Therefore, it is the device that is either granted or denied access to the server resources. Access is not controlled based on user identification.

Two SRX Series zones are established: one that includes the network devices (marketing-zone) and one that includes the internal servers (servers-zone). The SRX Series device interface ge-0/0/3.1, whose IP address is 192.0.2.18/24, is assigned to the marketing-zone zone. The SRX Series device interface ge-0/0/3.2, whose IP address is 192.0.2.14/24, is assigned to the servers-zone zone.

This examples covers the following activity:

1. The SRX Series device connects to the Active Directory domain controller using the WMI DCOM interface to obtain information about devices authenticated by Active Directory.

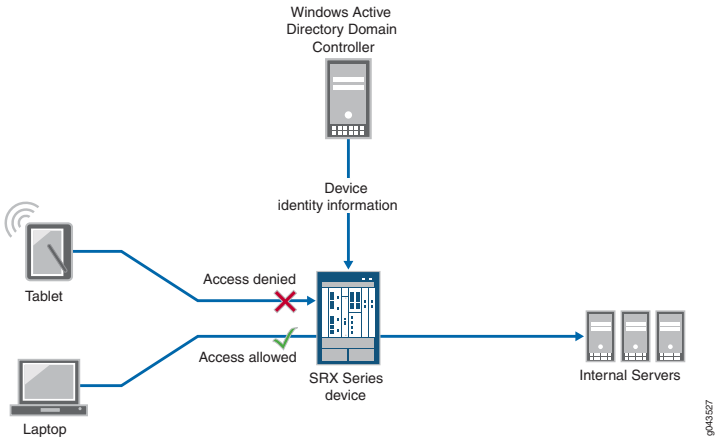
   When a user logs in to the network and is authenticated, information about the user's device is written to the event log.

2. The SRX Series extracts the device information from the event log of the Active Directory domain controller.
3. The SRX Series uses the extracted information to obtain a list of the groups that the device belongs to from the Active Directory LDAP server.
4. The SRX Series creates a local device identity authentication table and stores the device identity information that it obtained from the domain controller and LDAP server in the table.
5. When traffic from a device arrives at the SRX Series device, the SRX Series checks the device identity authentication table for a matching entry for the device that issued the traffic.
6. If the SRX Series finds a matching entry for the device that is requesting access, it checks the security policy table for a security policy whose **source-end-user-profile** field specifies a device identity profile with attributes that match those of the device that is requesting access.
7. The matching security policy is applied to traffic issuing from the device.

   This example is configured for the laptop device whose security policy allows it access to the internal corporate server.

Figure 19 on page 210 show the topology for this example.

Figure 19: Topology for the Device Identity Feature with Active Directory as the Authentication Source



Configuration

To configure the device identity feature in an Active Directory environment, perform these tasks:

- Configuring the Integrated User Firewall Device Identity Authentication Feature in an Active Directory Environment on page 211
- Results on page 213

CLI Quick Configuration

To quickly configure this example, copy the following commands to a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/3.1 family inet address 192.0.2.18/24
set interfaces ge-0/0/3.2 family inet address 192.0.2.14/24
set security zones security-zone marketing-zone interfaces ge-0/0/3.1 host-inbound-traffic
    system-services all
set security zones security-zone marketing-zone interfaces ge-0/0/3.1 host-inbound-traffic
    protocols all
set security zones security-zone servers-zone interfaces ge-0/0/3.2 host-inbound-traffic
    system-services all
set security zones security-zone servers-zone interfaces ge-0/0/3.2 host-inbound-traffic
    protocols all
set services user-identification device-information authentication-source active-directory
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast domain-name example.net
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-category string laptop
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-vendor string Lenovo
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-type string ThinkPad
set services user-identification device-information end-user-profile profile-name
    marketing-west-coast attribute device-os string Windows
```

```
set services user-identification device-information end-user-profile profile-name
  marketing-west-coast attribute device-os-version string 10.1
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match source-address any destination-address any
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match application any
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match source-end-user-profile marketing-west-coast
set security policies from-zone marketing-zone to-zone servers-zone policy
  mark-server-access then permit
set services user-identification active-directory-access domain example.net user1
  password pswd
set services user-identification active-directory-access domain example.net
  domain-controller dc-example address 203.0.113.0
set services user-identification active-directory-access domain example.net
  ip-user-mapping discovery-method wmi event-log-scanning-interval 30
set services user-identification active-directory-access domain example.net
  ip-user-mapping discovery-method wmi initial-event-log-timespan 1
set services user-identification active-directory-access domain example.net
  user-group-mapping ldap authentication-algorithm simple
set services user-identification active-directory-access domain example.net
  user-group-mapping ldap address 198.51.100.9 port 389
set services user-identification active-directory-access domain example.net
  user-group-mapping ldap base dc=example, dc=net
set services user-identification active-directory-access authentication-entry-timeout
  100
set services user-identification active-directory-access wmi-timeout 60
```

Configuring the Integrated User Firewall Device Identity Authentication Feature in an Active Directory Environment

Step-by-Step Procedure

This procedure includes the configuration statements required to configure the SRX Series device to support the device identity authentication feature in an Active Directory environment.

1. Configure the interfaces to be used for the marketing-zone and the servers-zone.

   ```
   [edit interfaces]
   user@host# set ge-0/0/3.1 family inet address 192.0.2.18/24
   user@host# set ge-0/0/3.2 family inet address 192.0.2.14/24
   ```

2. Configure the marketing-zone and the servers-zone and assign interfaces to them.

   ```
   [edit security zones]
   user@host# set security-zone marketing-zone interfaces ge-0/0/3.1
     host-inbound-traffic system-services all
   user@host# set security-zone marketing-zone interfaces ge-0/0/3.1
     host-inbound-traffic protocols all
   user@host# set security-zone servers-zone interfaces ge-0/0/3.2
     host-inbound-traffic system-services all
   user@host# set security-zone servers-zone interfaces ge-0/0/3.2
     host-inbound-traffic protocols all
   ```

3. Configure the authentication source to specify Microsoft Active Directory. You must specify the authentication source for the device identity feature to work. This is a required value.

```
[edit services user-identification]
user@host# set device-information authentication-source active-directory
```

4. Configure attributes for the device identity profile, which is also referred to as **end-user-profile**.

```
[edit services user-identification]
user@host# set device-information end-user-profile profile-name
  marketing-west-coast domain-name example.net
user@host# set device-information end-user-profile profile-name
  marketing-west-coast attribute device-category string laptop
user@host# set device-information end-user-profile profile-name
  marketing-west-coast attribute device-vendor string Lenovo
user@host# set device-information end-user-profile profile-name
  marketing-west-coast attribute device-type string ThinkPad
user@host# set device-information end-user-profile profile-name
  marketing-west-coast attribute device-os string Windows
user@host# set device-information end-user-profile profile-name
  marketing-west-coast attribute device-os-version string 10.1
```

5. Configure a security policy, called mark-server-access, that references the device identity profile called marketing-west-coast. The security policy allows any device that belongs to the marketing-zone zone (and that matches the device identity profile's attributes) access to the target server's resources.

```
[edit security policies]
user@host# set from-zone marketing-zone to-zone servers-zone policy
  mark-server-access match source-address any destination-address any
user@host# set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access match source-end-user-profile marketing-west-coast
user@host# set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access match application any
user@host# set security policies from-zone marketing-zone to-zone servers-zone
  policy mark-server-access then permit
```

6. Configure the SRX Series device to communicate with Active Directory and to use the LDAP service.

   To get the group information necessary to implement the device identity authentication feature, the SRX Series device uses the Lightweight Directory Access Protocol (LDAP). The SRX Series acts as an LDAP client communicating with an LDAP server. Typically, the Active Directory domain controller acts as the LDAP server. The LDAP module in the SRX Series, by default, queries the Active Directory in the domain controller.

```
[edit services user-identification]
user@host# set active-directory-access domain example.net user user1 password
  pswd
user@host# set active-directory-access domain example.net domain-controller
  dc-example address 203.0.113.0
user@host# set active-directory-access domain example.net ip-user-mapping
  discovery-method wmi event-log-scanning-interval 30
user@host# set active-directory-access domain example.net ip-user-mapping
  discovery-method wmi initial-event-log-timespan 1
user@host set active-directory-access domain example.net user-group-mapping
  ldap address 198.51.100.9 port 389
```



```
user@host# set active-directory-access domain example.net user-group-mapping
  ldap base dc=example,dc=net
user@host# set active-directory-access domain example.net user-group-mapping
  ldap authentication-algorithm simple
user@host# set active-directory-access authentication-entry-timeout 100
user@host# set active-directory-access wmi-timeout 60
```

Results

show interfaces

```
ge-0/0/3 {
  unit 1 {
    family inet {
      address 192.0.2.18/24;
    }
  }
  unit 2 {
    family inet {
      address 192.0.2.14/24;
    }
  }
}
```

show security zones

```
security-zone marketing-zone {
  interfaces {
    ge-0/0/3.1 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
security-zone servers-zone {
  interfaces {
    ge-0/0/3.2 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
```

show services user-identification device-information end-user-profile

```
domain-name example.net
attribute device-category {
  string laptop;
}
attribute device-vendor {
  string Lenovo;
}
attribute device-type {
  string ThinkPad;
}
attribute device-os {
  string Windows;
}
attribute device-os-version {
  string 10.1;
}
```

show services user-identification device-information authentication-source

```
active-directory;
```

show security policies

```
from-zone marketing-zone to-zone servers-zone {
  policy mark-server-access {
    match {
      source-address any;
      destination-address any;
      application any;
      source-end-user-profile {
        marketing-west-coast;
      }
    }
    then {
      permit;
    }
  }
}
```

show services user-identification active-directory-access

```
domain example-net {
  user {
    user1;
    password $ABC123; ## SECRET-DATA
  }
  ip-user-mapping {
    discovery-method {
      wmi {
        event-log-scanning-interval 30;
        initial-event-log-timespan 1;
      }
    }
  }
  user-group-mapping {
    ldap {
      base dc=example,DC=net;
```

```
        address 198.51.100.9 {
          port 389;
        }
      }
    }
  }
```

show services user-identification active-directory-access domain example-net

```
user {
  user1;
  password $ABC123 ## SECRET-DATA
}
domain-controller dc-example {
  address 203.0.113.0;
}
```

Verification

Verifying the Device Identity Authentication Table Contents

Purpose Verifying that the device identity authentication table contains the expected entries and their groups.

Action In this case, the device identity authentication table contains three entries. The following command displays extensive information for all three entries.

Enter **show services user-identification device-information table all extensive** to display the table's contents.

Sample Output

```
Domain: example.net
Total entries: 3
  Source IP: 192.0.2.19
    Device ID: example-dev1
    Device-Groups: device_group1, device_group2, device_group3, device_group4,
device_group5
    device-category: laptop
    device-vendor: Lenovo
    device-type: ThinkPad
    device-os: Windows
    device-os-version: 10.1
    Location1: us1
    Referred by: mark-server-access
  Source IP: 192.0.2.22
    Device ID: example-dev2
    Device-Groups: device_group06, device_group7, device_group8, device_group9,
device_group10
    device-category: laptop
    device-vendor: Lenovo
    device-type: ThinkPad
    device-os: Windows
    device-os-version: 10.1
```

```
    Location1: us1
    Referred by: mark-server-access
Source IP: 192.0.2.19
  Device ID: example-dev3
    Device-Groups: device_group1, device_group2, device_group3, device_group4,
device_group5
    device-category: Laptop
    device-vendor: Lenovo
    device-type: Thinkpad-t430
    device-os: Windows
    device-os-version: 7.1
    Location1: us1
    Referred by: mark-server-access
```

Meaning The table should contain entries with information for all authenticated devices and the groups that they belong to.

Verify the Domain Configuration on the SRX Series Device

Purpose Ensure that the SRX Series device is configured with the correct domain information.

Action Enter **show services user-identification active-directory-access domain example-net**.

```
user {
  user1;
  password $ABC123 ; ## SECRET-DATA
}
domain-controller dc-example {
  address 203.0.113.0;
}
```

Meaning The output should reflect the correct information configured for the domain.

Related Documentation

- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192
- Understanding Access Control to Network Resources Based on Device Identity Information on page 185

PART 5

Configuring Integrated ClearPass Authentication and Enforcement

CHAPTER 20

Understanding Integrated ClearPass Authentication and Enforcement

Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature

This topic introduces the SRX Series integrated ClearPass authentication and enforcement feature in which the SRX Series device and Aruba ClearPass collaborate to protect your network resources by enforcing security at the user identity level and controlling user access to the Internet. The ClearPass Policy Manager (CPPM) can authenticate users across wired, wireless, and VPN infrastructures. The integrated ClearPass feature allows the CPPM and the SRX Series device to collaborate in multiple environments in which they are deployed together.

Why You Need to Protect Your Environment With the SRX Series Integrated ClearPass Authentication and Enforcement Feature

The proliferation of mobile devices and cloud services and securing them has become a fundamental strategic part of enterprise cybersecurity. Use of company smartphones poses one of the biggest IT security risks to businesses. The integrated ClearPass feature protects against malicious intrusions introduced through use of mobile devices and multiple concurrently connected devices.

In a work environment that supports mobile devices, knowing the identity of the user whose device is associated with an attack or threat provides IT administrators with improved advantage in identifying the source of the attack and stemming future potential attacks that follow the same strategy.

Attackers can gain access to nearby company-owned mobile devices and install malware on them that they can then use to capture data at any time. Whether reconnaissance or malicious, attacks against network resources are commonplace in today's computing

environment. Attackers can launch information-gathering ventures, stop business activity, and steal sensitive corporate data.

Today's network environments are more open to attacks of various kinds because they support *anywhere, anytime, any device* access, to a greater or lesser degree, and they allow a user to use multiple concurrently network-connected devices.

The SRX Series integrated ClearPass authentication and enforcement feature can protect you against attacks and intrusions by allowing you to configure security policies that identify users by their usernames or by the groups that they belong to. It also identifies threats and attacks perpetrated against your network environment and provides this information to the CPPM. As administrator of the CPPM, you can better align your security enforcement to protect against possible future attacks of the same kind. If a user is logged in to the network with more than one device, you can keep track of their activity based on their identity, not only by their devices, and you can more easily control their network access and any egregious activity on their behalf, whether intended or not.

How the SRX Series Integrated ClearPass Authentication and Enforcement Feature Can Protect Your Network Environment

The SRX Series integrated ClearPass authentication and enforcement feature gives you granular control at the user level, not the device's IP address, over user access to protected resources and the Internet. As administrator of the SRX Series device, you can now specify in the source-identity parameter of *identity-aware* security policies a username or a role (group) name that the CPPM posts to the SRX Series device. You are no longer restricted to relying solely on the IP address of the device as a means of identifying the user. Honing in on the user of the device, rather than only the device, enhances your control over security enforcement.

In addition to providing the SRX Series device with authenticated user information, the CPPM can map a device type to a role and assign users to that role. It can then send that role mapping to the SRX Series device. This capability allows you to control through security policies a user's access to resources when they are using a *specific type of device*.

For example, suppose that the administrator of the CPPM configured a role called marketing-company-device and mapped to that role both company devices and members of the Marketing department. As administrator of the SRX Series device, you could specify that role in a security policy as if it were a group. The security policy would then apply to all users mapped to the role, inherently controlling their network activity when they use that type of device type.

The SRX Series integrated ClearPass feature delivers the protection of the SCREENS, IDP and UTM features to defend your network against a wide range of attack strategies. In addition to protecting the company's network resources, the SRX Series device can make available to the CPPM log records generated by these protective security features in response to attack or attack threats. Knowing about threats and specific attacks that have already occurred can help IT departments to identify noncompliant systems and exposed areas of the network. With this information, they can harden their security by enforcing device compliance and strengthening protection of their resources.

SRX Series security policies protect the company's resources and enforce access control at a fine-grain level, taking advantage of the user authentication and identity information sent to the device from the CPPM. The CPPM acts as the authentication source. It uses its own internal RADIUS server to authenticate users. It can also rely on an external authentication source to perform the authentication for it, such as an external RADIUS server or Active Directory.

The CPPM authentication is triggered by requests from NAS devices such as switches and access controllers. The CPPM uses the XML portion of the RESTful Web services that the SRX Series device exposes to it to send in POST request messages to the SRX Series device authenticated user identity and device posture information.

The SRX Series device and Aruba ClearPass simplify the complex and complicated security tasks required to safeguard company resources and enforce Internet access policy for mobile devices. This security is essential in a network environment that supports the mobile experience and that gives the user latitude to use a wide range of devices, including their own systems, smartphones, and tablets.

Related Documentation

- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223
- Example: Configuring the SRX Series Integrated ClearPass Feature to Allow the Device to Receive User Authentication Data from ClearPass on page 226
- Example: Configuring the SRX Series Integrated ClearPass Feature to Allow the Device to Receive User Authentication Data from ClearPass on page 226
- Example: Enforcing SRX Series Security Policies Using Aruba ClearPass as the Authentication Source on page 246

CHAPTER 21

Configuring Communication with ClearPass Using the WebAPI Feature

Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API

The integrated ClearPass authentication and enforcement feature enables the SRX Series device and Aruba ClearPass to collaborate in protecting your company's resources by enforcing security at the user identity level in environments in which they are deployed together. The ClearPass Policy Manager (CPPM) can authenticate users across wired, wireless, and VPN infrastructures and post that information to the SRX Series device, which, in turn, uses it to authenticate users requesting access to your protected resources and to the internet. The SRX Series device can provide the CPPM with threat and attack logs associated users' devices so that you can better harden your security at the ClearPass end.

Web API

The SRX Series device exposes to the CPPM its Web API daemon (webapi) interface that enables the CPPM to integrate with it and efficiently send authenticated user identity information to the SRX Series device. The SRX Series Web API daemon acts as an HTTP server in that it implements part of the RESTful Web services that supports concurrent HTTP and HTTPS requests. In this relationship, the CPPM is the client. The Web API

daemon is restricted to processing only HTTP/HTTPS requests. Any other type of request it receives generates an error message.

WARNING: **If you are deploying the integrated ClearPass Web API function and Web management at the same time, you must ensure that they use different HTTP or HTTPS service ports.**

However, for security considerations, we recommend that you use HTTPS instead of HTTP. HTTP is supported primarily for debugging purposes.

The Web API daemon runs on the master Routing Engine in a chassis cluster environment. After an Chassis Cluster switchover, the daemon will start automatically on the new master Routing Engine. It has no effect on the Packet Forwarding Engine.

ClearPass Authentication Table

After the SRX Series device receives information posted to it from the CPPM, the SRX Series device extracts the user authentication and identity information, analyzes it, and distributes it to the appropriate processes for handling. The SRX Series device creates a ClearPass authentication table on the Packet Forwarding Engine side to hold this user information. When the SRX Series device receives the information sent to it from ClearPass, the SRX Series device generates entries in the ClearPass authentication table for the authenticated users. When the SRX Series device receives an access request from a user, it can check its ClearPass authentication table to verify that the user is authenticated, and then apply the security policy that matches the traffic from the user.

Using HTTPS or HTTP for the Connection Protocol Between ClearPass and the SRX Series Device

When you configure the SRX Series Web API, you specify a certificate key if you are using HTTPS as the connection protocol. To ensure security, the HTTPS default certificate key size is 2048 bytes. If you do not specify a certificate size, the default size is assumed. There are three methods that you can use to specify a certificate:

- Default certificate
- Certificate generated by PKI
- Custom certificate and certificate key

  The SRX Series Web API supports only the Privacy-Enhanced Mail (PEM) format for the certificate and certificate key configuration.

If you enable the Web API on the default ports—HTTP (8080) or HTTPS (8443)—you must enable host inbound traffic on the ports. If you enable it on any other TCP port, you must enable host inbound traffic specifying the parameter **any-service**. For example:

```
user@host# set security zones security-zone trust host-inbound-traffic system-services
    any-service
```

Ensuring the Integrity of Data Sent from ClearPass to the SRX Series Device

The following requirements ensure that the data sent from the CPPM is not compromised:

- The Web API implementation is restricted to processing only HTTP/HTTPS POST requests. Any other type of request that it receives generates an error message.
- The Web API daemon analyzes and processes HTTP/HTTPS requests from only the following dedicated URL:

  `/api/userfw/v1/post-entry`
- The HTTP/HTTPS content that the CPPM posts to the SRX Series device must be consistently formatted correctly. The correct XML format indicates a lack of compromise, and it ensures that user identity information is not lost.

Data Size Restrictions and Other Constraints

The following data size restrictions and limitations apply to the CPPM:

- The CPPM must control the size of the data that it posts. Otherwise the Web API daemon is unable to process it. Presently the Web API can process a maximum of 2 megabytes of data.
- The following limitations apply to XML data for role and device posture information. The Web API daemon discards XML data sent to it that exceeds these amounts (that is, the overflow data):
  - The SRX Series device can process a maximum of 209 roles.
  - The SRX Series device supports only one type of posture with six possible posture tokens, or values. Identity information for an individual user can have only one posture token.

NOTE: The CPPM checks the health and posture of a device and it can send that information to the SRX Series device as part of the user information that it posts. You cannot define posture on the SRX Series device. Also, the SRX Series device does not check posture information that it receives.

Posture States and the Posture Group

User, role, and posture token fields are distinct in the context of the CPPM. Each set of user identity information contains user and role (group) identity and a posture token. Because the SRX Series device supports only user and role (group) fields, the posture token value is mapped to a role by adding the prefix **posture–**. You can then use that role in a security policy as a group and that policy will be applied to all traffic that matches the policy.

The predefined posture identity states are:

- posture-healthy (HEALTHY)
- posture-checkup (CHECKUP)
- posture-transition (TRANSITION)
- posture-quarantine (QUARANTINE)
- posture-infected (INFECTED)
- posture-unknown (UNKNOWN)

Related Documentation

- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Example: Enforcing SRX Series Security Policies Using Aruba ClearPass as the Authentication Source on page 246
- Understanding the Integrated ClearPass Authentication and Enforcement User Query Function on page 265

Example: Configuring the SRX Series Integrated ClearPass Feature to Allow the Device to Receive User Authentication Data from ClearPass

The SRX Series device and the ClearPass Policy Manager (CPPM) collaborate to control access to your protected resources and to the Internet. To carry this out, the SRX Series device must authenticate users in conjunction with applying security policies that match their requests. For the integrated ClearPass authentication and enforcement feature, the SRX Series device relies on ClearPass as its authentication source.

The Web API function, which this example covers, exposes to the CPPM an API that enables it to initiate a secure connection with the SRX Series device. The CPPM uses this connection to post user authentication information to the SRX Series device. In their relationship, the SRX Series device acts as an HTTPS server for the CPPM client.

Requirements

This section defines the software and hardware requirements for the topology for this example. See Figure 21 on page 231 for the topology design.

The hardware and software components are:

- Aruba ClearPass Policy Manager (CPPM). The CPPM is configured to use its local authentication source to authenticate users.

  NOTE: It is assumed that the CPPM is configured to provide the SRX Series device with user authentication and identity information, including the username, a list of the names of any groups that the user belongs to, the IP addresses of the devices used, and the device posture token.

- SRX Series device running Junos OS that includes the integrated ClearPass feature.
- A server farm composed of six servers, all in the servers-zone:
  - marketing-server-protected (203.0.113.23)
  - human-resources-server (203.0.113.25)
  - accounting-server (203.0.113.72)
  - public-server (192.0.2.96)
  - corporate-server (203.0.113.71)
  - sales-server (203.0.113.81)
- AC 7010 Aruba Cloud Services Controller running ArubaOS.
- Aruba AP wireless access controller running ArubaOS.

  The Aruba AP is connected to the AC7010.

  Wireless users connect to the CPPM through the Aruba AP.
- Juniper Networks EX4300 switch used as the wired 802.1 access device.

  Wired users connect to the CPPM using the EX4300 switch.
- Six end-user systems:
  - Three wired network-connected PCs running Microsoft OS
  - Two BYOD devices that access the network through the Aruba AP access device
  - One wireless laptop running Microsoft OS

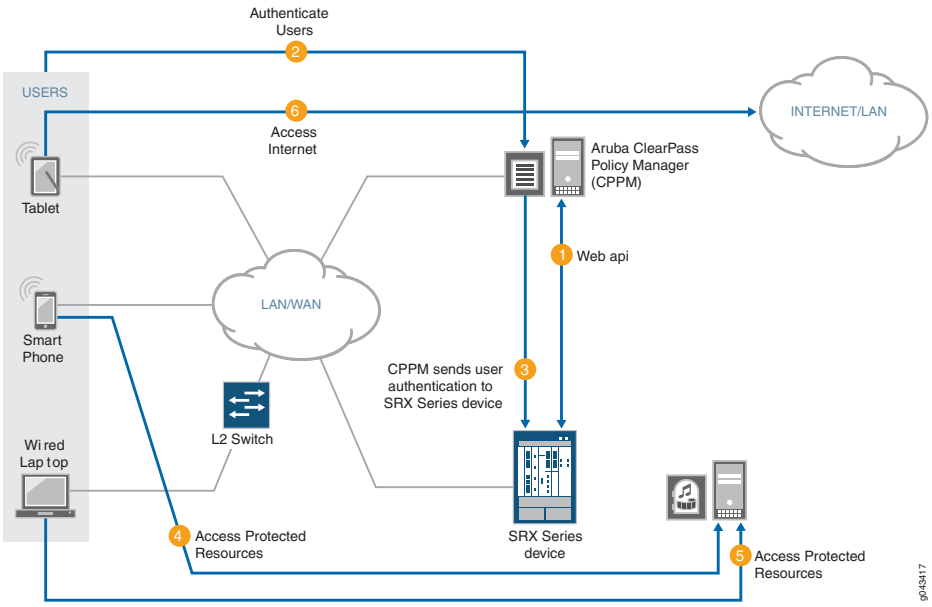
Overview

You can configure identity-aware security policies on the SRX Series device to control a user's access to resources based on username or group name, not the IP address of the device. For this feature, the SRX Series device relies on the CPPM for user authentication.

The SRX Series device exposes to ClearPass its Web API (webapi) to allow the CPPM to integrate with it. The CCPM posts user authentication information efficiently to the SRX Series device across the connection. You must configure the Web API function to allow the CPPM to initiate and establish a secure connection. There is no separate Routing Engine process required on the SRX Series device to establish a connection between the SRX Series device and the CPPM.

Figure 20 on page 228 illustrates the communication cycle between the SRX Series device and the CPPM, including user authentication.

Figure 20: ClearPass and SRX Series Device Communication and User Authentication Process



As depicted, the following activity takes place:

1. The CPPM initiates a secure connection with the SRX Series device using Web API.
2. Three users join the network and are authenticated by the CPPM.
   - A tablet user joins the network across the corporate WAN.
   - A smartphone user joins the network across the corporate WAN.
   - A wireless laptop user joins the network from a wired laptop connected to a Layer 2 switch that is connected to the corporate LAN.
3. The CPPM sends the user authentication and identity information for the users who are logged in to the network to the SRX Series device in POST request messages using the Web API.

   When traffic from a user arrives at the SRX Series device, the SRX Series device:
   - Identifies a security policy that the traffic matches.
   - Locates an authentication entry for the user in the ClearPass authentication table.

- Applies the security policy to the traffic after authenticating the user.

4. Traffic from the smartphone user who is requesting access to an internal, protected resource arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the protected resource.
5. Traffic from the wired laptop user who is requesting access to a protected resource arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the resource.
6. Traffic from the tablet user who is requesting access to the Internet arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the Internet.

The Web API daemon is not enabled by default for security reasons. When you start up the Web API daemon, by default it opens either the HTTP (8080) or the HTTPS (8443) service port. You must ensure that one of these ports is configured, depending on which version of the HTTP protocol you want to use. We recommend that you use HTTPS for security reasons. Opening these ports makes the system more vulnerable to service attacks. To protect against service attacks that might use these ports, the Web API daemon will start up only after you enable it.

The Web API is a RESTful Web services implementation. However, it does not fully support the RESTful Web services. Rather, it acts as an HTTP or HTTPS server that responds to requests from the ClearPass client.

NOTE: **The Web API connection is initialized by the CPPM using the HTTP service port (8080) or HTTPS service port (8443). For ClearPass to be able to post messages, you must enable and configure the Web API daemon.**

To mitigate abuse and protect against data tampering, the Web API daemon:

- Requires ClearPass client authentication by HTTP or HTTPS basic user account authentication.
- Allows data to be posted to it only from the IP address configured as the client source. That is, it allows HTTP or HTTPS POST requests only from the ClearPass client IP address, which in this example is 192.0.2.199.
- Requires that posted content conforms to the established XML data format. When it processes the data, the Web API daemon ensures that the correct data format was used.

NOTE: **Note that if you deploy Web management and the SRX Series device together, they must run on different HTTP or HTTPS service ports.**

See "Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API" on page 223 for further information on how this feature protects against data tampering.

The SRX Series UserID daemon processes the user authentication and identity information and synchronizes it to the ClearPass authentication table on the Packet Forwarding Engine. The SRX Series device creates the ClearPass authentication table to be used for information received only from the CPPM. The ClearPass authentication table does not contain user authentication information from other authentication sources. The SRX Series device checks the ClearPass authentication table to authenticate users attempting to access protected network resources on the Internet using wired or wireless devices and local network resources.

For the CPPM to connect to the SRX Series device and post authentication information, it must be certified using HTTPS authentication. The Web API daemon supports three methods that can be used to refer to an HTTPS certificate: a default certificate, a PKI local certificate, and a customized certificate implemented through the certificate and certificate-key configuration statements. These certificate methods are mutually exclusive.

This example uses HTTPS for the connection between the CPPM and the SRX Series device. To ensure security, the integrated ClearPass feature default certificate key size is 2084 bits.

Whether you use any method—the default certificate, a PKI-generated certificate, or a custom certificate—for security reasons, you must ensure that the certificate size is 2084 bits or greater.

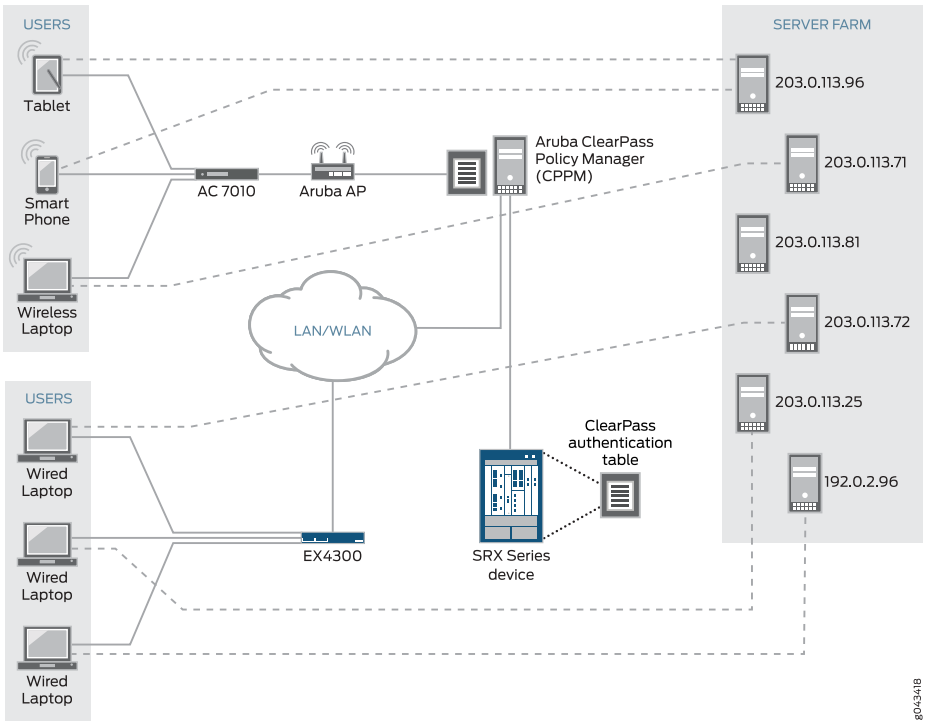
The following example shows how to generate a certificate and key using PKI:

```
user@host>request security pki generate-key-pair certificate-id aruba size 2048
user@host>request security pki local-certificate generate-self-signed certificate-id aruba
    domain-name mycompany.net email jxchan@mycompany.net ip-address 192.51.100.21
    subject "CN=John Doe,OU=Sales ,O=mycompany.net ,L=MyCity ,ST=CA,C=US"
```

Topology

Figure 21 on page 231 shows the topology used for the integrated ClearPass deployment examples.

Figure 21: Integrated ClearPass Authentication and Enforcement Deployment Topology



Configuration

This section covers how to enable and configure the SRX Series Web API.

NOTE: You must enable the Web API. It is not enabled by default.

- Configuring the SRX Series Web API Daemon on page 232
- Configuring the ClearPass Authentication Table Entry Timeout and Priority on page 234

CLI Quick Configuration

To quickly configure this example, copy the following statements, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the statements into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set system services webapi user sunny password i4%rgd
set system services webapi client 192.0.2.199
set system services webapi https port 8443
set system services webapi https default-certificate
set system services webapi debug-level alert
set interfaces ge-0/0/3.4 vlan-id 340 family inet address 192.51.100.21
set security zones security-zone trust interfaces ge-0/0/3.4 host-inbound-traffic
    system-services webapi-ssl
set security user-identification authentication-source aruba-clearpass priority 110
```

```
set security user-identification authentication-source local-authentication-table priority
  120
set security user-identification authentication-source active-directory-authentication-table
  priority 125
set security user-identification authentication-source firewall-authentication priority 150
set security user-identification authentication-source unified-access-control priority 200
```

Configuring the SRX Series Web API Daemon

Step-by-Step Procedure

Configuring the Web API allows the CPPM to initialize a connection to the SRX Series device. No separate connection configuration is required.

It is assumed that the CPPM is configured to provide the SRX Series device with authenticated user identity information, including the username, the names of any groups that the user belongs to, the IP addresses of the devices used, and a posture token.

Note that the CPPM might have configured role mappings that map users or user groups to device types. If the CPPM forwards the role mapping information to the SRX Series device, the SRX Series device treats the role mappings as groups. The SRX Series device does not distinguish them from other groups.

Step-by-Step Procedure

To configure the Web API daemon:

1. Configure the Web API daemon (webapi) username and password for the account.

   This information is used for the HTTPS certification request.

   ```
   [edit system services]
   user@host# set webapi user sunny password i4%rgd
   ```

2. Configure the Web API client address—that is, the IP address of the ClearPass webserver's data port.

   The SRX Series device accepts information from this address only.

   NOTE: The ClearPass webserver data port whose address is configured here is the same one that is used for the user query function, if you configure that function.

   ```
   [edit system services]
   user@host# set webapi client 192.0.2.199
   ```

3. Configure the Web API daemon HTTPS service port.

   If you enable the Web API service on the default TCP port 8080 or 8443, you must enable host inbound traffic on that port.

   In this example, the secure version of the Web API service is used (webapi-ssl), so you must configure the HTTPS service port, 8443.

   ```
   [edit system services]
   user@host# set webapi https port 8443
   ```

4. Configure the Web API daemon to use the HTTPS default certificate.

```
[edit system services]
user@host# set webapi https default-certificate
```

5. Configure the trace level for the Web API daemon.

   The supported trace levels are notice, warn, error, crit, alert, and emerg. The default value is error.

   ```
   [edit system services]
   user@host# webapi debug-level alert
   ```

6. Configure the interface to use for host inbound traffic from the CPPM.

   ```
   user@host# set interfaces ge-0/0/3.4 vlan-id 340 family inet address 192.51.100.21
   ```

7. Enable the Web API service over HTTPS host inbound traffic on TCP port 8443.

   ```
   [edit security zones]
   user@host# set security-zone trust interfaces ge-0/0/3.4 host-inbound-traffic
     system-services webapi-ssl
   ```

Results From configuration mode, confirm your Web API configuration by entering the **show system services webapi** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user {
  sunny;
  password "$ABC123"; ## SECRET-DATA
}
client {
  192.0.2.199;
}
https {
  port 8443;
  default-certificate;
}
debug-level {
  alert;
}
```

From configuration mode, confirm the configuration for the interface used for host inbound traffic from the CPPM by entering the **show interfaces ge-0/0/3.4** command. If the output does not display the intended configuration, repeat the verification process in this example to correct it.

```
vlan-id 340;
family inet {
    address 192.51.100.21/32;
}
```

From configuration mode, confirm your security zone configuration that allows host-inbound traffic from the CPPM using the secure Web API service (web-api-ssl) by entering the **show security zones security-zone trust** command. If the output does not display the intended configuration, repeat the verification process in this example to correct it.

```
interfaces {
  ge-0/0/3.4 {
    host-inbound-traffic {
      system-services {
        webapi-ssl;
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Configuring the ClearPass Authentication Table Entry Timeout and Priority

Step-by-Step Procedure

This procedure configures the following information:

- The timeout parameter that determines when to age out idle authentication entries in the ClearPass authentication table.
- The ClearPass authentication table as the first authentication table in the lookup order for the SRX Series device to search for user authentication entries. If no entry is found in the ClearPass authentication table and there are other authentication tables configured, the SRX Series device will search them, based on the order that you set.

1. Set the timeout value that is used to expire idle authentication entries in the ClearPass authentication table to 20 minutes.

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba-clearpass authentication-entry-timeout
     20
   ```

   The first time that you configure the SRX Series device to integrate with an authentication source, you must specify a timeout value to identify when to expire idle entries in the ClearPass authentication table. If you do not specify a timeout value, the default value is assumed.

   - default = 30 minutes
   - range = If set, the timeout value should be within the range [10,1440 minutes]. A value of 0 means that the entry will never expire.

2. Set the authentication table priority order to direct the SRX Series device to search for user authentication entries in the ClearPass authentication table first. Specify the order in which other authentication tables are searched if an entry for the user is not found in the ClearPass authentication table.

   NOTE: You need to set this value if the ClearPass authentication table is *not* the only authentication table on the Packet Forwarding Engine.

   ```
   [edit security user-identification]
   user@host# set authentication-source aruba-clearpass priority 110
   user@host# set authentication-source local-authentication-table priority 120
   ```

```
user@host# set authentication-source active-directory-authentication-table priority
    125
user@host# set authentication-source firewall-authentication priority 150
user@host# set authentication-source unified-access-control priority 200
```

The default priority value for the ClearPass authentication table is 110. You must change the local authentication table entry from 100 to 120 to direct the SRX Series device to check the ClearPass authentication table first if there are other authentication tables on the Packet Forwarding Engine. Table 20 on page 235 shows the new authentication table search priority.

Table 20: SRX Series Device Authentication Tables Search Priority Assignment

SRX Series Authentication Tables	Set Value
ClearPass authentication table	110
Local authentication table	120
Active Directory authentication table	125
Firewall authentication table	150
UAC authentication table	200

Results From configuration mode, confirm that the timeout value set for aging out ClearPass authentication table entries is correct. Enter the **show services user-identification** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
authentication-source aruba-clearpass {
    authentication-entry-timeout 20;
}
```

Related Documentation

- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Understanding Enforcement of ClearPass User and Group Authentication on the SRX Series Devices on page 237
- Understanding the Integrated ClearPass Authentication and Enforcement User Query Function on page 265

CHAPTER 22

Configuring Integrated ClearPass Authentication and Enforcement

Understanding Enforcement of ClearPass User and Group Authentication on the SRX Series Devices

This topic describes how the SRX Series device enforces user and group authentication when a user attempts to access a resource. It also explains how the SRX Series device handles information in the ClearPass authentication table user entries when a security policy that references a group in a user entry is removed. Understanding that process will help you troubleshoot issues related to group identity and give you insight into changes in the ClearPass authentication table user entries.

Understanding How the SRX Series Device Manages the ClearPass Authentication Table

The integrated ClearPass authentication and enforcement feature enables the SRX Series device and the Aruba ClearPass Policy Manager (CPPM) to collaborate in protecting your company's resources. It enables the SRX Series device to apply firewall security policies to user traffic and to control user access to protected resources based on user or group identity. To ensure the identity of the user, the SRX Series device relies on authenticated user information that it receives from the CPPM.

It is useful to understand how the SRX Series device gets authenticated user identity information from the CPPM, generates entries in its ClearPass authentication table, and

manages those entries in relation to security policies and user events. Understanding these processes will help you to quickly identify and resolve related problems.

This topic focuses on:

- How the SRX Series device obtains user identity information from the CPPM and manages it, and how you can use this information in security policies.
- How security policies that reference a group as the source (source-identity) have bearing on the groups listed in user entries in the ClearPass authentication table. Groups that are referenced by security policies are referred to as *interested groups*.

User Authentication Entries in the ClearPass Authentication Table

In their collaboration, ClearPass acts as the authentication source for the SRX Series device. The CPPM sends to the SRX Series device identity information about users that it has authenticated. The UserID daemon process in the SRX Series device receives this information, processes it, and synchronizes it to the Packet Forwarding Engine side in the independent ClearPass authentication table that is generated for this purpose.

As administrator of the SRX Series device, you can use the authenticated user identity information in security policies to control access to your protected resources and the Internet.

The collection of user identity information that the SRX Series device obtains from the CPPM and uses to create entries in its global Routing Engine authentication table that is synchronized to its individual ClearPass authentication table is referred to as a mapping, or, more commonly, an IP-user mapping because the username and the related group list are mapped to the IP address of the user's device.

NOTE: For each user authentication entry in the ClearPass authentication table, a group list identifies the groups that a user belongs to in addition to other information such as the posture token, which indicates state of the device, such as whether it is healthy.

You can use a username or a group name in security policies to identity a user and not rely directly on the IP address of the device used, because the IP address of the device is tied to the username and its groups in the ClearPass authentication table entry.

NOTE: For each user entry, the number of groups, or roles, in the entry cannot exceed 200. After the capacity is reached, additional roles are discarded and the following syslog message is sent:

```
userid_get_and_check_adauth_num: src_ip ip-address user domain:user
  dropped.record numrecord-number has arrived max num of db
```

The CPPM posts user information to the SRX Series device in the following format. The SRX Series device does not use all of this information.

```
<userfw-entries>
  <userfw-entry>
    <source>Aruba ClearPass</source>
    <timestamp>2016-01-29T0310Z</timestamp>
    <operation>logon</operation>
    <IP>192.0.2.123</IP>
    <domain>my-company-domain</domain>
    <user>user1</user>
    <role-list>
      <role>human-resources-grp</role>
      <role>[User Authenticated],/role>
    </role-list>
    <posture>HEALTHY</posture>
    <device_category>Computer</device_category>
  </userfw-entry>
</userfw-entries>
```

Here is the format for a ClearPass authentication table entry for a user, followed by an example entry and a description of its components.

IP-address, domain, user, user-group-list

In the following example, the user belongs to two groups, the human-resources-grp group and the posture-healthy group. The SRX Series device converts the posture information from the CPPM to a group name. You might configure a security policy that allows all users access to the marketing server if their devices belong to the posture-healthy group (role).

192.0.2.11 , my-company-domain, lin, human-resources-grp, posture-healthy

- IP address

  This is the IP address of the device used.

- The name of the domain that the user belongs to.

  In this example, the domain name is “my-company-domain.” The default domain name GLOBAL is used if a domain name is not provided.

- The username

  The username is the user’s login name used to connect to the network, which, in this example, is lin.

  This name is constant regardless of the device used.

  When you configure a security policy whose source-identity tuple identifies the source of the traffic by username or group name, not by the IP address of the device used, it is as if the security policy were device independent; it applies to the user’s activity regardless of the device used.

- One or more groups that a user belongs to

  It is here where the concept of *interested groups* and their relationship to security policies comes into play. An interested group is a group that is referenced in a security policy. The concept of interested groups is covered later in this topic.

Note that if a user is connected to the network using multiple devices, there might be more than one IP-user mapping for that user. Each mapping would have its own set of

values—that is, domain name and group-list—in conjunction with the username and IP address.

For example, the following three IP address-to-username mappings might exist for the user abe who is connected to the network using three separate devices:

203.0.113.5 abe, marketing-grp, posture-healthy
192.0.2.34 abe, marketing-grp, posture-transition
203.0.133.19 abe, marketing-grp, posture-unknown

Assume that the SRX Series device receives a logout message for 110.208.132.23, abe. The following partial user authentication entry shows that the user abe is now logged in to the network using only two devices:

192.0.2.34 abe, marketing-grp, posture-transition
203.0.133.19 abe, marketing-grp, posture-unknown

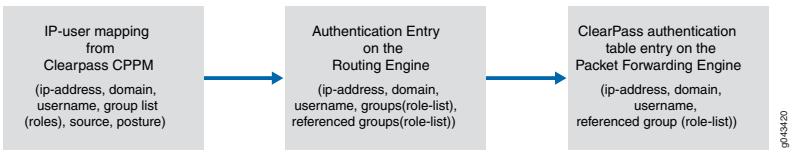
Communication Between ClearPass and the SRX Series Device

Here is a summary of how the SRX Series device and ClearPass communicate:

- A user joins the company network via a wired or wireless LAN.
- The CPPM authenticates the user.
- The CPPM initiates a secure connection with the SRX Series device using the integrated Web API.
- The SRX Series UserID daemon gets the full IP-user mapping from the CPPM. For each authenticated user, the UserID daemon generates an entry in the Routing Engine authentication table.

  The Routing Engine authentication table is common in that it holds authentication entries based on information from other authentication sources in addition to ClearPass. For example, it might also hold entries for users authenticated by Microsoft Active Directory.
- The UserID daemon synchronizes the user authentication information from the Routing Engine authentication table to the ClearPass authentication table on the Packet Forwarding Engine. The ClearPass authentication table is dedicated to holding only ClearPass authentication information. See Figure 22 on page 240.

Figure 22: User Information from the CPPM to the SRX Series Device Routing Engine Synchronized to the ClearPass Authentication Table



The SRX Series device uses the authenticated user identity information in the following process. When a user attempts to access an internal, protected resource or the Internet, the SRX Series device:

- Checks the traffic generated by the user for a matching security policy. The source traffic must match all of the tuples specified in the security policy. The match includes the source-identity field, which specifies a username or a group name.

  To identify a match, the SRX Series device compares the username or the group name with the source-identity specification that is configured in a security policy, along with all other security policy values.

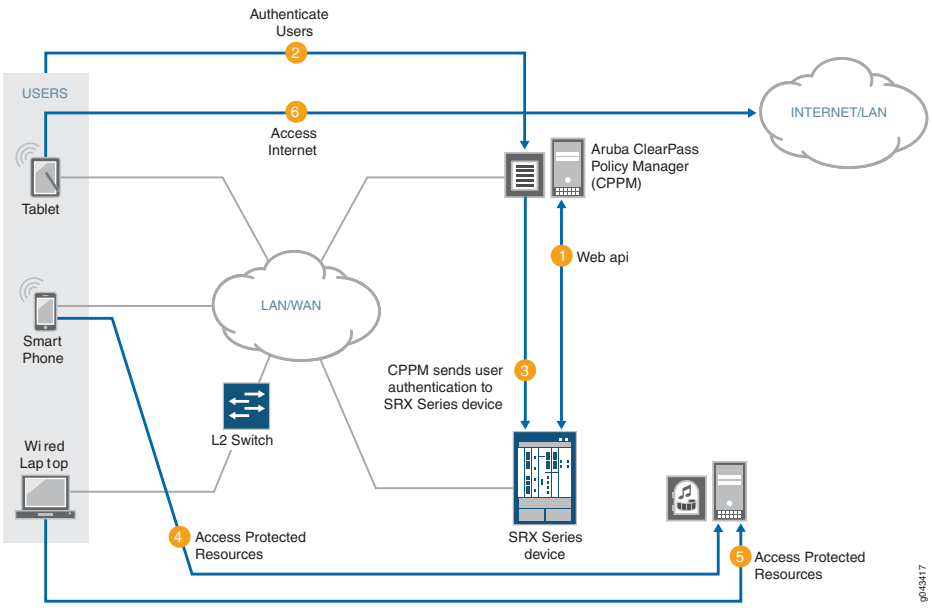
- Checks the ClearPass authentication table for an authentication entry for the user, if a security policy match was found.

  If it does not find an entry in the ClearPass authentication table, the SRX Series device checks other local authentication tables, in the order that you specified, until a match is found. However, it does not check other local authentication tables if the user query function is configured. See "Understanding the Integrated ClearPass Authentication and Enforcement User Query Function" on page 265.

NOTE: The SRX Series device can query the CPPM for individual user information, under certain circumstances, when it has not already received that information from the CPPM. This feature is referred to as user query.

Figure 23 on page 241 illustrates the connection and communication between the SRX Series device and the CPPM. It also shows the paths entailed in authenticating users and allowing them access to the Internet and internal, protected resources.

Figure 23: ClearPass and SRX Series Device Communication and User Authentication Process



As Figure 23 on page 241 depicts, the following activity takes place:

1. The CPPM initiates a secure connection with the SRX Series device using the Web API.
2. Three users join the network and are authenticated by the CPPM.
   - A tablet user joins the network across the corporate WAN.
   - A smartphone user joins the network across the corporate WAN.
   - A wireless laptop user joins the network from a wired laptop connected to a Layer 2 switch that is connected to the corporate LAN.
3. The CPPM sends the user authentication and identity information for the users who are logged in to the network to the SRX Series device in POST request messages using the Web API.

   When traffic from a user arrives at the SRX Series device, the SRX Series device:
   - Identifies a security policy that the traffic matches.
   - Locates an authentication entry for the user in the ClearPass authentication table.
   - Applies the security policy to the traffic after authenticating the user.
4. Traffic from the smartphone user who is requesting access to an internal, protected resource arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the protected resource.
5. Traffic from the wired laptop user who is requesting access to a protected resource arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the resource.
6. Traffic from the tablet user who is requesting access to the Internet arrives at the SRX Series device. Because all of the conditions identified in Step 3 are met and the security policy permits it, the SRX Series device allows the user connection to the Internet.

Understanding Domains and Interested Groups

How the user identity group information is managed on the SRX Series device is dominated by two concepts:

- Domain group

  The SRX Series device follows the usual course in regard to how it handles usernames in domain namespaces. It makes use of the namespace to distinguish names that are the same–such as **admin**—but that are from different sources and are in different domains. Because they belong to different domains, the names are not in conflict.

  Any group that is part of an IP-user mapping will always belong to a domain, whether that domain is a specific domain or the GLOBAL domain. If a domain name is not specified in the IP-user mapping, then the GLOBAL domain is assumed.

Table 21 on page 243 illustrates how the domain for a group is determined, based on the IP-user mapping information obtained from the CPPM.

Table 21: Assigning a Domain to a Group

Does the IP-User Mapping Contain a Domain Name?	What Domain Is Applied to the Group?
No For example: **IP, , user1, group-list** The second comma serves as a placeholder for the domain name and the GLOBAL domain is applied.	Groups included in group-list belong to the GLOBAL domain.
Yes For example: **IP, domain1, user1, group-list** NOTE: In this example, the IP-user mapping specifies the domain name as domain1.	The domain name, domain1, is included in the IP-user mapping from the CPPM, and it is used. It is retained in the entry for the authenticated user in the ClearPass authentication table on the Packet Forwarding Engine.

- Interested group

  A group qualifies as an *interested group* if it is referenced by a security policy—that is, if it is specified in a policy's source-identity field. On the Routing Engine authentication table, each user entry contains a group referenced by a policy list that identifies the names of the groups for which a security policy exists. If a group included in a user entry is not currently used in a security policy, it is not included in this list. A group can move in and out of the groups referenced by a policy list.

  - Interested group lists

    An interested group list, or a list of groups referenced by policies, is a subset of overall groups. It is the intersection of the group list in a user authentication entry and the source-identity list for security policies. That is, any group included in a ClearPass authentication table user entry qualifies as an interested group. The Routing Engine synchronizes to the user entry in the ClearPass authentication table on the Packet Forwarding Engine only those groups that are referenced by security policies.

    Here is how it works:

    - The UserID daemon gets the full IP-user role (group) mapping from the CPPM.
    - For each group, the UserID daemon identifies whether it is an interested group by determining if there is a security policy that references it. Any qualifying groups are included in the groups referenced by a policy list on the Routing Engine. The UserID daemon synchronizes to the user entry in the ClearPass authentication table on the Packet Forwarding Engine interested groups along with the rest of the user authentication and identity information.

The interested groups list for a user entry on the Routing Engine can change, based on the following events:

- A new security policy is configured that references a group included in the user entry on the Routing Engine but that is not already in the entry's referenced groups list.
- A currently configured security policy that references a group in its source-identity is deleted.

Consider the following example:

- Assume that the CPPM posted the following information for two users to the SRX Series device:

  192.51.100.1, abe, group1, group2, group3, group4, healthy
  192.0.2.21, john, group1, group5, healthy

- After the SRX Series device maps the posture, defining it as a group, the two user entries in the SRX Series device Routing Engine authentication table appear as follows:

  192.51.100.1, abe, group1, group2, group3, group4, posture-healthy
  192.0.2.21, john, group1, group5, posture-healthy

- Assume that several security policies include source-identity fields that reference one of the following: group1, group3, posture-healthy.

  The intersection of the preceding sets—the original group list and the list of security policies that refer to the groups—results in the following interested groups list:

  - For the user john, the groups referenced by policy list includes group1 and posture-healthy.
  - For the user abe, the groups referenced by policy list includes group1, group3, and posture-healthy.

Now suppose that the security policy whose source-identity field specified group1 was deleted. The groups referenced by policy lists for the user authentication entries for the two users—john and abe—would be changed, producing the following results:

- For the user john, the list would include only posture-healthy.
- For the user abe, the list would include group3 and posture-healthy.

Table 22 on page 245 shows how a security policy that references a group affects the ClearPass authentication table. It also shows the effect on the ClearPass authentication table when a group is *not* referenced by a security policy, and therefore is not an interested group.

Table 22: Interested Groups: Effect on the ClearPass Authentication Table

Security Policies Configuration and Modification	Resulting Effect on ClearPass Authentication Table Packet Forwarding Engine Entries
Case 1: The SRX Series device gets the IP-user mapping for a user from the CPPM. None of the groups in the user mapping are referenced by security policies.	
IP-user mapping from the CPPM: 203.0.113.9, ,user1, g1, g2, g3, g4	The user authentication entry written to the ClearPass authentication table in the Packet Forwarding Engine for this user does not contain any groups. 203.0.113.9 , ,user1
Case 2: The SRX Series device gets the IP-user mapping for a user from the CPPM. It checks the groups list against the security policies list and finds that two of the groups are referenced by security policies.	
IP-user mapping on the Routing Engine: 192.0.2.1, domain1, user2, g1, g2, g3, g4	The user authentication entry written to the ClearPass authentication table on the Packet Forwarding Engine for this user includes the following groups that are included in the groups referenced by the policy list on the Routing Engine: 192.0.2.1, domain1, user2, g2, g4

When a User Has Already Been Authenticated By Another Source

It can happen that the SRX Series device Routing Engine authentication table and the individual Microsoft Active Directory authentication table on the Packet Forwarding Engine, for example, contain an entry for a user who was authenticated by Active Directory. As usual, the CPPM sends the IP-user mapping for the user to the SRX Series device. The SRX Series device must resolve the problem because its Routing Engine authentication table is common to both Active Directory and ClearPass.

Here is how the SRX Series device handles the situation:

- On the Routing Engine authentication table:
  - The SRX Series device overwrites the Active Directory authentication entry for the user in its common Routing Engine authentication table with the newly generated one from the IP-user mapping for the user from the CPPM.

    There is now no IP address or username conflict.
- On the Packet Forwarding Engine:

- The SRX Series device deletes the existing Active Directory authentication entry for the user from the Active Directory authentication table.

  This will delete active sessions associated with the IP address.

- The SRX Series device generates a new entry for the CPPM-authenticated user in the Packet Forwarding Engine ClearPass authentication table.

  Traffic associated with the IP-user mapping entry will initiate new sessions based on user authentication in the ClearPass authentication table.

Related Documentation

- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Example: Enforcing SRX Series Security Policies Using Aruba ClearPass as the Authentication Source on page 246
- Example: Configuring the Integrated ClearPass Authentication and Enforcement User Query Function on page 268

Example: Enforcing SRX Series Security Policies Using Aruba ClearPass as the Authentication Source

This example covers how to configure security to protect your resources and control access to the internet using the SRX Series device integrated ClearPass authentication and enforcement feature, which relies on the Aruba ClearPass Policy Manager as its authentication source. The SRX Series integrated ClearPass feature allows you to configure security policies that control access to company resources and the Internet by identifying users by username, group name, or the name of a role that ties together a group of users and a device type.

Today's network environments are more open to attacks of various kinds because they support *anywhere, anytime, any device* access, to a greater or lesser degree, and they allow a user to use multiple concurrently network-connected devices. Because it allows you identify the user by username, the integrated ClearPass authentication and enforcement feature narrows the security gap that these capabilities introduce.

For details on how user authentication and identity information is conveyed from the CPPM to the SRX Series device, see the following topics:

- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223
- Understanding the Integrated ClearPass Authentication and Enforcement User Query Function on page 265

The example covers the following processes:

- How to control access at the user level based on username or group name, not device IP address.

  You can use the source-identity parameter in a security policy to specify the name of a user or the name of a group of users whose authentication is provided by the CPPM.

The policy is applied to traffic generated by the users when they attempt to access a protected resource or the Internet regardless of the device used. The access control is tied to the user's name, and not directly to the IP address of the user's device.

NOTE: **You can configure different security policies for a single user that specify different actions, differentiated by the zones and the destination addresses specified or a group that the user belongs to.**

- How to display and interpret the contents of the ClearPass authentication table.

  The SRX Series device creates the ClearPass authentication table to contain user authentication and identity information that it receives from the CPPM. The device refers to the table to authenticate a user who requests access to a resource.

  The ClearPass authentication table contents are dynamic. They are modified to reflect user activity in response to various events and also in regard to security policies that reference groups.

  For example, when a user logs out of the network or in to the network, the ClearPass authentication table is modified, as is the case when a user is removed from a group or a referenced security policy that specifies a group that the user belongs to is deleted. In the latter case, the user entry no longer shows the user as belonging to that group.

  In this example, the ClearPass authentication table contents are displayed to depict changes made because of two events. The content for the users is displayed:

  - Before and after a specific user logs out of the network
  - Before and after a referenced security policy is deleted

    The entry for the user who belonged to the group referenced by the security policy is displayed before and after the policy is deleted.

Requirements

This section defines the software and hardware requirements for the topology for this example. See Figure 24 on page 251 for the topology design.

The hardware and software components are:

- Aruba ClearPass. The ClearPass Policy Manager (CPPM) is configured to use its local authentication source to authenticate users.

  NOTE: It is assumed that the CPPM is configured to provide the SRX Series device with user authentication and identity information, including the username, a list of the names of any groups that the user belongs to, the IP addresses of the devices used, and the device posture token.

- SRX Series device running Junos OS that includes the integrated ClearPass feature.
- A server farm composed of six servers, all in the servers-zone:
  - marketing-server-protected (203.0.113.23)
  - human-resources-server (203.0.113.25)
  - accounting-server (203.0.113.72)
  - public-server (203.0.113.62)
  - corporate-server (203.0.113.71)
  - sales-server (203.0.113.81)
- AC 7010 Aruba Cloud Services Controller running ArubaOS.
- Aruba AP wireless access controller running ArubaOS.

  The Aruba AP is connected to the AC7010.

  Wireless users connect to the CPPM through the Aruba AP.
- Juniper Networks EX4300 switch used as the wired 802.1 access device.

  Wired users connect to the CPPM using the EX4300 switch.
- Six end-user systems:
  - Three wired network-connected PCs running Microsoft OS
  - Two BYOD devices that access the network through the Aruba AP access device
  - One wireless laptop running Microsoft OS

Overview

In its capacity as the authentication source for the integrated ClearPass feature, the CPPM posts to the SRX Series device user authentication and identity information. When it receives this information, the SRX Series UserID daemon processes it and generates

entries for the authenticated users in the Routing Engine authentication table and then synchronizes that information to the ClearPass authentication table on the Packet Forwarding Engine side.

The SRX Series device requires the user authentication and identity information to verify that a user is authenticated when the user makes an access request and the traffic generated from the user's device arrives at the SRX Series device. If a security policy exists that specifies in the source-identity parameter the username or the name of a group that the user belongs to, the SRX Series device searches the contents of its ClearPass authentication table for an entry for that user.

If it does not find an entry for the user in its ClearPass authentication table, the SRX Series device can search its other authentication tables, if you have configured a search order that includes them. See "Example: Configuring the SRX Series Integrated ClearPass Feature to Allow the Device to Receive User Authentication Data from ClearPass" on page 226 for information about the authentication table search order.

The integrated ClearPass feature allows you to create identity-aware security policies configured to match traffic issued by users based on their username or the name of a group that they belong to.

NOTE: **You configure role mappings on the CPPM, not on the SRX Series device.**

For example, a device type role mapping might tie user identities to company-owned computers. You could specify this role as a group in a security policy configured to apply to all users who are mapped to the rule. In this case, the conditions set by CPPM for the rule—use of company-owned computer—would apply to all users mapped to the rule. The SRX Series device does not consider the conditions, but rather accepts the rule from the CPPM.

The following configurations included in this example cover security policies that are applicable based on the type of device used as defined by the CPPM through rule mappings. It is assumed that the CPPM posted to the SRX Series device the following mapped rules that are used as groups in security policies:

- marketing-access-for-pcs-limited-group

  Maps jxchan to the device type PC.

  The policy that specifies marketing-access-for-pcs-limited-group in its source-identity field allows jxchan, and other users who are mapped to it, access to the marketing-server-protected server using their PC, whether it is company owned or not.

- accounting-grp-and-company-device

  Maps users who belong to accounting groups using company devices. The CPPM sends the role accounting-grp-and-company-device to the SRX Series device. The mapping is done on the CPPM by role mapping rules.

  The policy that specifies accounting-grp-and-company-device in its source identity field allows users who are mapped to the rule to access protected resources on the

accounting-server. The group accounting-grp is mapped to the rule. Therefore the mapped rule applies to the members of accounting-grp.

The user viki2 belongs to accounting-grp. If all conditions apply—that is, if viki2 is using a company-owned device and the policy permits access—she is allowed access to the resources on accounting-server. But, recall that the SRX Series device does not analyze the rule. Rather it applies it to all users who are mapped to it by the CPPM.

- guest-device-byod

  Maps the guest group to the device type byod—that is, any user-owned device brought to the network.

  The policy that specifies guest-device-byod in its source identity field denies users who are mapped to the rule access to all servers in the server zone if they are using smartphones or other user-owned devices. The username guest2 is mapped to this rule by the CPPM.

For all cases, if the users are allowed or denied access according to the security policy conditions, you can assume that the following conditions exist:

- The CPPM posted the correct authentication information for the users and groups to the SRX Series device.
- The SRX Series device processed the authenticated user information correctly and generated entries for the users and groups in its ClearPass authentication table.

Table 23 on page 250 summarizes the users, their groups, and the zones to which they belong. All users belong to the default GLOBAL domain.

Table 23: Authenticated User Information for Security Policy Example

User	Group	Zone
Abe (abew1)	• marketing-access-limited-grp	marketing-zone
John (jxchan)	• posture-healthy • marketing-access-for-pcs-limited-group • marketing-general • sales-limited • corporate-limited	marketing-zone
Lin (lchen1)	• posture-healthy • human-resources-grp • accounting-limited • corporate-limited	human-resources-zone
Viki (viki2)	• posture-healthy • accounting-grp • accounting-grp-and-company-device • corporate-limited	accounting-zone

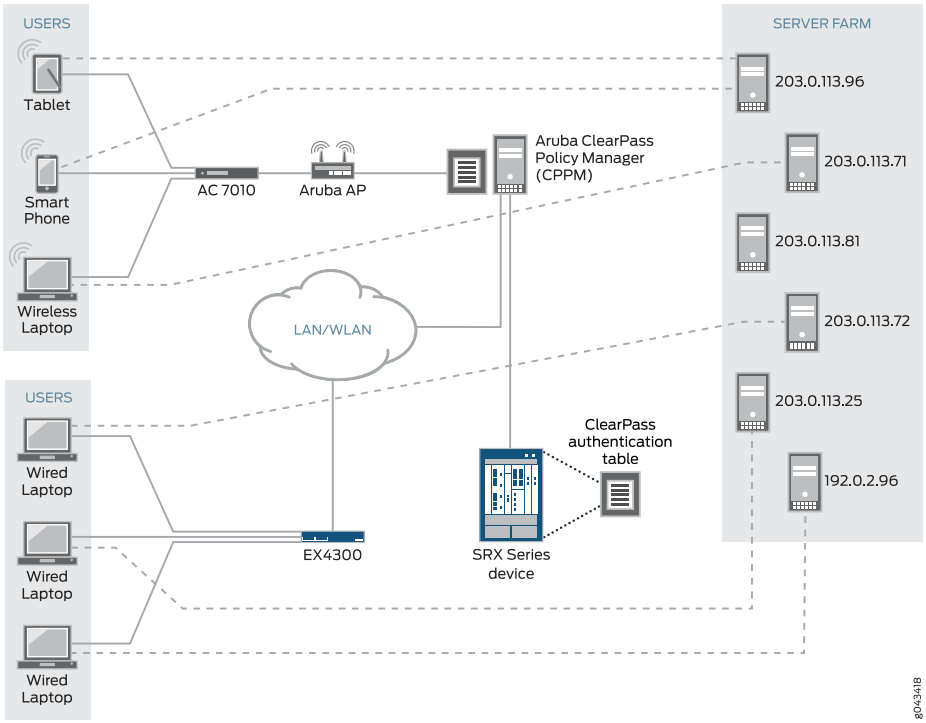
Table 23: Authenticated User Information for Security Policy Example *(continued)*

User	Group	Zone
guest1	• posture-healthy • guest	public-zone
guest2	• posture-healthy • guest-device-byod	public-zone

Topology

Figure 24 on page 251 shows the topology for this example.

Figure 24: Topology for the Integrated ClearPass Authentication Enforcement Through Security Policies Example



Configuration

This section covers how to configure the SRX Series device to include security policies that match traffic issued by users authenticated by the CPPM.

- Configuring Interfaces, Zones, and an Address Book on page 254
- Configuring Identity-Aware Security Policies to Control User Access to Company Resources on page 257
- Results on page 259

CLI Quick Configuration

To quickly configure this example, copy the following statements, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the statements into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/3 vlan-tagging
set interfaces ge-0/0/3.0 vlan-id 300 family inet address 203.0.113.45/24
set interfaces ge-0/0/3.1 vlan-id 310 family inet address 192.0.2.18/24
set interfaces ge-0/0/3.2 vlan-id 320 family inet address 192.0.2.14/24
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4.0 vlan-id 400 family inet address 192.0.2.16/24
set interfaces ge-0/0/4.1 vlan-id 410 family inet address 192.0.2.19/24
set security zones security-zone marketing-zone interfaces ge-0/0/3.0
    host-inbound-traffic system-services all
set security zones security-zone marketing-zone interfaces ge-0/0/3.0
    host-inbound-traffic protocols all
set security zones security-zone accounting-zone interfaces ge-0/0/3.1
    host-inbound-traffic system-services all
set security zones security-zone accounting-zone interfaces ge-0/0/3.1
    host-inbound-traffic protocols all
set security zones security-zone human-resources-zone interfaces ge-0/0/3.2
    host-inbound-traffic system-services all
set security zones security-zone human-resources-zone interfaces ge-0/0/3.2
    host-inbound-traffic protocols all
set security zones security-zone public-zone interfaces ge-0/0/4.0 host-inbound-traffic
    system-services all
set security zones security-zone public-zone interfaces ge-0/0/4.0 host-inbound-traffic
    protocols all
set security zones security-zone servers-zone interfaces ge-0/0/4.1 host-inbound-traffic
    system-services all
set security zones security-zone servers-zone interfaces ge-0/0/4.1 host-inbound-traffic
    protocols all
set security address-book servers-zone-addresses address marketing-server-protected
    203.0.113.23
set security address-book servers-zone-addresses address human-resources-server
    203.0.113.25
set security address-book servers-zone-addresses address accounting-server 203.0.113.72
set security address-book servers-zone-addresses address corporate-server 203.0.113.71
set security address-book servers-zone-addresses address public-server 203.0.113.91
set security address-book servers-zone-addresses attach zone servers-zone
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match source-address any destination address any
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match application any
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match source-identity “global\marketing-access-for-pcs-limited-group”
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p1
    then permit
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p2
    match source-address any destination address marketing-zone-protected
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p2
    match application any
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p2
    match source-identity “global\abew1”
```

```
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p2
    then permit
set security policies from-zone accounting-zone to-zone servers-zone policy acct-cp-device
    match source-address any destination-address accounting-server
set security policies from-zone accounting-zone to-zone servers-zone policy acct-cp-device
    match application any
set security policies from-zone accounting-zone to-zone servers-zone policy acct-cp-device
    match source-identity “global\accounting-grp-and-company-device”
set security policies from-zone accounting-zone to-zone servers-zone policy acct-cp-device
    then permit
set security policies from-zone human-resources-zone to-zone servers-zone policy
    human-resources-p1 match source-address any destination-address corporate-server
set security policies from-zone human-resources-zone to-zone servers-zone policy
    human-resources-p1 match application any
set security policies from-zone human-resources-zone to-zone servers-zone policy
    human-resources-p1 match source-identity “global\corporate-limited”
set security policies from-zone human-resources-zone to servers-zone policy
    human-resources-p1 then permit
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p0
    match source-address any destination-address corporate-server
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p0
    match application any
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p0
    match source-identity “global\marketing-access-limited-grp”
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p0
    then permit
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p3
    match source-address any destination-address human-resources-server
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p3
    match application any
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p3
    match source-identity “global\sales-limited-group”
set security policies from-zone marketing-zone to-zone servers-zone policy marketing-p3
    then permit
set security policies from-zone public-zone to-zone servers-zone policy guest-allow-access
    match source-address any destination address public-server
set security policies from-zone public-zone to-zone servers-zone policy guest-allow-access
    match application any
set security policies from-zone public-zone to-zone servers-zone policy guest-allow-access
    match source-identity “global\guest”
set security policies from-zone public-zone to-zone servers-zone policy guest-allow-access
    then permit
set security policies from-zone public-zone to-zone servers-zone policy guest-deny-access
    match source-address any destination-address any
set security policies from-zone public-zone to-zone servers-zone policy guest-deny-access
    match application any
set security policies from-zone public-zone to-zone servers-zone policy guest-deny-access
    match source-identity “global\guest-device-byod”
set security policies from-zone public-zone to-zone servers-zone policy guest-deny-access
    then deny
```

Configuring Interfaces, Zones, and an Address Book

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

Configure the following interfaces and assign them to zones:

- ge-0/0/3.0 > marketing-zone
- ge-0/0/3.1 > human-resources-zone
- ge-0/0/3.2> accounting-zone
- ge-0/0/4.0 > public-zone
- ge-0/0/4.1 > servers-zone

Because this example uses logical interfaces, you must configure VLAN tagging.

1. Configure interfaces for the SRX Series device:

```
[edit interfaces]
set ge-0/0/3 vlan-tagging
set ge-0/0/3.0 vlan-id 300 family inet address 203.0.113.45/24
set ge-0/0/3.1 vlan-id 310 family inet address 192.0.2.18/24
set ge-0/0/3.2 vlan-id 320 family inet address 192.0.2.14/24
set ge-0/0/4 vlan-tagging
set ge-0/0/4.0 vlan-id 400 family inet address 192.0.2.16/24
set ge-0/0/4.1 vlan-id 410 family inet address 192.0.2.19/24
```

2. Configure zones.

```
[edit security zones]
user@host#set security-zone marketing-zone interfaces ge-0/0/3.0
  host-inbound-traffic system-services all
user@host#set security-zone marketing-zone interfaces ge-0/0/3.0
  host-inbound-traffic protocols all
user@host#set security-zone accounting-zone interfaces ge-0/0/3.1
  host-inbound-traffic system-services all
user@host#set security-zone accounting-zone interfaces ge-0/0/3.1
  host-inbound-traffic protocols all
user@host#set security-zone human-resources-zone interfaces ge-0/0/3.2
  host-inbound-traffic system-services all
user@host#set security-zone human-resources-zone interfaces ge-0/0/3.2
  host-inbound-traffic protocols all
user@host#set security-zone public-zone interfaces ge-0/0/4.0 host-inbound-traffic
  system-services all
user@host#set security-zone public-zone interfaces ge-0/0/4.0 host-inbound-traffic
  protocols all
user@host#set security-zone servers-zone interfaces ge-0/0/4.1 host-inbound-traffic
  system-services all
user@host#set security-zone servers-zone interfaces ge-0/0/4.1 host-inbound-traffic
  protocols all
```

3. Configure an address book containing the IP addresses of the servers to use as destination addresses in security policies.

```
[edit security address-book servers-zone-addresses]
user@host# set address marketing-server-protected 203.0.113.23
user@host# set address human-resources-server 203.0.113.25
user@host# set address accounting-server 203.0.113.72
user@host# set address corporate-server 203.0.113.71
user@host# set address public-server 203.0.113.91
```

4. Attach the servers-zone-addresses address book to servers-zone.

```
[edit security address-book]
user@host# set servers-zone-addresses attach zone servers-zone
```

Results From configuration mode, confirm your configuration for interfaces by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
ge-0/0/3 {
    unit 0 {
        vlan-id 300;
        family inet {
            address 203.0.113.45/24;
        }
    }
    unit 1 {
        vlan-id 310;
        family inet {
            address 192.0.2.18/24;
        }
    }
    unit 2 {
        vlan-id 320;
        family inet {
            address 192.0.2.14/24;
        }
    }
}
ge-0/0/4 {
    vlan-tagging;
    unit 0 {
        vlan-id 400;
        family inet {
            address 192.0.2.16/24;
        }
    }
    unit 1 {
        vlan-id 410;
        family inet {
            address 192.0.2.19/24;
        }
    }
}
```

From configuration mode, confirm your configuration for zones by entering the **show security zones** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
security-zone human-resources-zone {
  interfaces {
    ge-0/0/3.2 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
security-zone accounting-zone {
  interfaces {
    ge-0/0/3.1 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
security-zone marketing-zone {
  interfaces {
    ge-0/0/3.0 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
security-zone servers-zone {
  interfaces {
    ge-0/0/4.1 {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
    }
  }
}
```

```
security-zone public-zone {
    interfaces {
        ge-0/0/4.0 {
            host-inbound-traffic {
                system-services {
                    all;
                }
                protocols {
                    all;
                }
            }
        }
    }
}
```

From configuration mode, confirm your configuration for the address book by entering the **show security address-book** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
servers-zone-addresses {
    address marketing-zone-protected 203.0.113.23 /32;
    address human-resources-server 203.0.113.25 /32;
    address accounting-server 203.0.113.72/32;
    address corporate-server 203.0.113.71/32;
    address public-server 203.0.113.91/32;
    attach {
        zone servers-zone;
    }
}
```

Configuring Identity-Aware Security Policies to Control User Access to Company Resources

Step-by-Step Procedure

This task entails configuring security policies that apply to a user's access to resources based on username or group name, and not the IP address of the device used.

Note that all users belong to the default GLOBAL domain.

1. Configure a security policy that specifies marketing-access-for-pcs-limited-group as the source-identity. It allows the user jxchan, who belongs to this group, access to any of the servers in the servers-zones when he is using a PC, whether it is a personal device or a company-owned device. The username jxchan is mapped by the CPPM to the rule marketing-access-for-pcs-limited-group.

```
[edit security policies]
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match source-address any destination address any
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match application any
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p1
    match source-identity "global\marketing-access-for-pcs-limited-group"
user@hoset from-zone marketing-zone to-zone servers-zone policy marketing-p1
    then permit
```

2. Configure a security policy that allows the user abew1 access to the marketing-zone-protected server (IP address 203.0.113.23) in the servers-zone regardless of the device that he uses.

```
[edit security policies]
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p2
  match source-address any destination address marketing-zone-protected
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p2
  match application any
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p2
  match source-identity “global\abew1”
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p2
  then permit
```

3. Configure a security policy that allows the user viki2 access to the accounting-server (IP address 203.0.113.72) in the servers-zone when she is using a company-owned device. The user viki2 belongs to accounting-grp which is mapped to the company-owned-device rule (accounting-grp-and-company-device) by the CPPM.

```
[edit security policies]
user@host# set from-zone accounting-zone to-zone servers-zone policy
  acct-cp-device match source-address any destination-address accounting-server
user@host# set from-zone accounting-zone to-zone servers-zone policy
  acct-cp-device match application any
user@host# set from-zone accounting-zone to-zone servers-zone policy
  acct-cp-device match source-identity
  “global\accounting-grp-and-company-device”
user@host# set from-zone accounting-zone to-zone servers-zone policy
  acct-cp-device then permit
```

4. Configure a security policy that allows users who belong to the corporate-limited group limited access to the corporate-server server (IP address 203.0.113.71) in the servers-zone when they are initiating a request from the human-resources zone.

   If the source-address were specified as “any”, the policy would apply to other users who also belong to the corporate-limited group.

```
[edit security policies]
user@host# set from-zone human-resources-zone to-zone servers-zone policy
  human-resources-p1 match source-address any destination-address
  corporate-server
user@host# set from-zone human-resources-zone to-zone servers-zone policy
  human-resources-p1 match application any
user@host# set from-zone human-resources-zone to-zone servers-zone policy
  human-resources-p1 match source-identity “global\corporate-limited”
user@host# set from-zone human-resources-zone to servers-zone policy
  human-resources-p1 then permit
```

5. Configure a security policy that allows the user abew1 access to the corporate-server (IP address 203.0.113.71) server in the servers-zone. The user abew1 belongs to marketing-access-limited-grp to which the security policy applies.

```
[edit security policies]
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p0
  match source-address any destination-address corporate-server
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p0
  match application any
```

```
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p0
  match source-identity “global\marketing-access-limited-grp”
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p0
  then permit
```

6. Configure a security policy that allows users who belong to the sales-limited-group access to the human-resources-server (IP address 203.0.113.81) server when they initiate a request from the marketing-zone. The user jxchan belongs to sales-limited-group.

```
[edit security policies]
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p3
  match source-address any destination-address human-resources-server
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p3
  match application any
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p3
  match source-identity “global\sales-limited-group”
user@host# set from-zone marketing-zone to-zone servers-zone policy marketing-p3
  then permit
```

7. Configure a security policy that allows users who belong to the guest group access to the public-server (IP address 203.0.113.91) in the servers-zone.

```
[edit security policies]
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-allow-access match source-address any destination address public-server
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-allow-access match application any
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-allow-access match source-identity “global\guest”
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-allow-access then permit
```

8. Configure a security policy that denies users who belong to the guest-device-byod group access to any servers in the servers-zone when they use their own devices.

```
[edit security policies]
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-deny-access match source-address any destination-address any
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-deny-access match application any
user@host# user@host# set from-zone public-zone to-zone servers-zone policy
  guest-deny-access match source-identity “global\guest-device-byod”
user@host# set from-zone public-zone to-zone servers-zone policy
  guest-deny-access then deny
```

Results

From configuration mode, confirm your security policies configuration for integrated ClearPass by entering the **show security policies** command.

If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
from-zone marketing-zone to-zone servers-zone {
  policy marketing-p1 {
    match {
```

```
            source-address any;
            destination-address any;
            application any;
            source-identity "global\marketing-access-for-pcs-limited-group";
          }
          then {
            permit;
          }
        }
        policy marketing-p2 {
          match {
            source-address any;
            destination-address marketing-zone-protected;
            application any;
            source-identity "global\abew1";
          }
          then {
            permit;
          }
        }
        policy marketing-p0 {
          match {
            source-address any;
            destination-address corporate-server;
            application any;
            source-identity "global\marketing-access-limited-grp";
          }
          then {
            permit;
          }
        }
        policy marketing-p3 {
          match {
            source-address any;
            destination-address human-resources-server;
            application any;
            source-identity "global\sales-limited-group";
          }
          then {
            permit;
          }
        }
      }
      from-zone accounting-zone to-zone servers-zone {
        policy acct-cp-device {
          match {
            source-address any;
            destination-address accounting-server;
            application any;
            source-identity "global\accounting-grp-and-company-device";
          }
          then {
            permit;
          }
        }
      }
```

```
from-zone human-resources-zone to-zone servers-zone {
  policy human-resources-p1 {
    match {
      source-address any;
      destination-address corporate-server;
      application any;
      source-identity "global\corporate-limited";
    }
    then {
      permit;
    }
  }
}
from-zone public-zone to-zone servers-zone {
  policy guest-allow-access {
    match {
      source-address any;
      destination-address public-server;
      application any;
      source-identity “global\guest”;
    }
    then {
      permit;
    }
  }
  policy guest-deny-access {
    match {
      source-address any;
      destination-address any;
      application any;
      source-identity “global\guest-device-byod”;
    }
    then {
      deny;
    }
  }
}
```

Verification

This section verifies the ClearPass authentication table contents after certain events occur that cause some of its user authentication entries to be modified. It also shows how to ensure that the ClearPass authentication table has been deleted successfully after you issue the delete command. It includes the following parts:

- Displaying the ClearPass Authentication Table Contents Before and After an Authenticated User Logs Out of the Network on page 262
- Displaying the Authentication Table Contents Before and After a Referenced Security Policy Is Deleted on page 262

Displaying the ClearPass Authentication Table Contents Before and After an Authenticated User Logs Out of the Network

Purpose Display the ClearPass authentication table contents when a specific, authenticated user is logged in to the network and after the user logs out.

Action Enter the **show services user-identification authentication-table authentication-source *authentication-source*** command for the ClearPass authentication table, which is referred to as aruba-clearpass. Notice that the ClearPass authentication table includes an entry for the user viki2.

```
show services user-identification authentication-table authentication-source
aruba-clearpass
Domain: GLOBAL
Total entries: 6
Source IP       Username       groups(Ref by policy)          state
203.0.113.21        viki2          accounting-grp-and-company-dev Valid
203.0.113.89        abew1          marketing-access-limited-grp   Valid
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1         corporate-limited              Valid
203.0.113.54        guest1                                        Valid
203.0.113.55        guest2                                        Valid
```

Enter the same command again after viki2 logs out of the network. Notice that the ClearPass authentication table no longer contains an entry for viki2.

```
Domain: GLOBAL
Total entries: 6
Source IP       Username       groups(Ref by policy)          state
203.0.113.89        abew1          marketing-access-limited-grp   Valid
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1         corporate-limited              Valid
203.0.113.54        guest1                                        Valid
203.0.113.55        guest2                                        Valid
```

Displaying the Authentication Table Contents Before and After a Referenced Security Policy Is Deleted

Purpose Display the ClearPass authentication table contents for a specific user—lchen1—who belongs to a group that is referenced by a security policy. Delete that security policy, then display the entry for that user again.

Action Enter the **show service user-identification authentication-table authentication-source user *user-name*** command to display the ClearPass authentication table entry for a specific user, lchen1. Notice that it includes the group corporate-limited.

show service user-identification authentication-table authentication-source user lchen1

```
Domain: GLOBAL
Source IP       Username       groups(Ref by policy)          state
203.0.113.53        lchen1         corporate-limited              Valid
```

The human-resources-p1 security policy source-identity field refers to the group corporate-limited. As shown above in the ClearPassauthentication entry for him, the user lchen1 belongs to that group. Here is the configuration for the human-resources-p1 referenced security policy:

```
from-zone human-resources-zone to-zone servers-zone {
    policy human-resources-p1 {
        match {
            source-address any;
            destination-address corporate-server;
            application any;
            source-identity "global\corporate-limited";
        }
        then {
            permit;
        }
    }
}
```

After you delete the human-resources-p1 security policy, whose source-identity parameter refers to the group called corporate-limited, enter the same command again. Notice that the authentication entry for lchen1 does not contain the corporate-limited group.

show service user-identification authentication-table authentication-source aruba-clearpass user lchen1

```
Domain: GLOBAL
Source IP       Username       groups(Ref by policy)          state
203.0.113.53        lchen1                                        Valid
```

Take a different approach in verifying the ClearPass authentication table state after the modification. Display the entire table to verify that the group—corporate-limited—is not included in any of the user entries. Note that if more than one user belonged to the corporate-limited group, authentication entries for all of the affected users would not show that group name.

From operational mode, enter the **show services user-identification authentication-table authentication-source aruba-clearpass** command.

```
show services user-identification authentication-table authentication-source
aruba-clearpass
Domain: GLOBAL
Total entries: 6
Source IP       Username       groups(Ref by policy)          state
203.0.113.21        viki2          accounting-grp-and-company-dev Valid
203.0.113.89        abew1          marketing-access-limited-grp   Valid
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1                                        Valid
203.0.113.54        guest1                                        Valid
203.0.113.55        guest2                                        Valid
```

Related Documentation

- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Understanding Enforcement of ClearPass User and Group Authentication on the SRX Series Devices on page 237
- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223

CHAPTER 23

Configuring the Integrated ClearPass Authentication and Enforcement User Query Function

Understanding the Integrated ClearPass Authentication and Enforcement User Query Function

This topic focuses on how you can obtain user authentication and identity information for an individual user when that information is not posted directly to the SRX Series device by the ClearPass Policy Manager (CPPM).

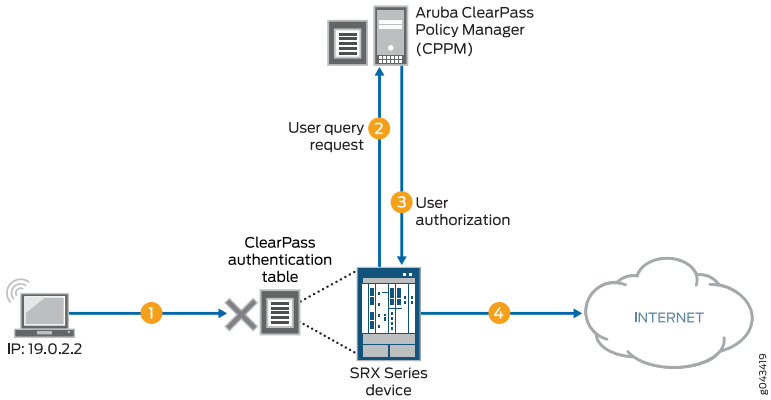
The SRX Series integrated ClearPass authentication and enforcement feature allows the SRX Series device and Aruba ClearPass to control access to protected resources and the Internet from wireless and wired devices. For this to occur, ClearPass sends user authentication and identity information to the SRX Series device. The SRX Series device stores the information in its ClearPass authentication table. To send this information, usually the CPPM uses the Web API (webapi) services implementation, which allows it to make HTTP or HTTPS POST requests to the SRX Series device.

It can happen that the CPPM does not send user authentication information for a user, for various reasons. When traffic from that user arrives at the SRX Series device, the device cannot authenticate the user. If you configure the SRX Series device to enable the user query function, it can query the ClearPass webserver for authentication information for an individual user. The SRX Series device bases the query on the IP address of the user's device, which it obtains from the user's access request traffic.

If the user query function is configured, the query process is triggered automatically when the SRX Series device does not find an entry for the user in its ClearPass authentication table when it receives traffic from that user requesting access to a resource or the Internet. The SRX Series device does not search its other authentication tables. Rather, it sends a query to the CPPM requesting authentication information for the user.
Figure 25 on page 266 depicts the user query process. In this example:

1. A user attempts to access a resource. The SRX Series device receives the traffic requesting access. The SRX Series device searches for an entry for the user in its ClearPass authentication table, but none is found.
2. The SRX Series device requests authentication for the user from the CPPM.
3. The CPPM authenticates the user and returns the user authentication and identity information to the SRX Series device.
4. The SRX Series device creates an entry for the user in its ClearPass authentication table, and grants the user access to the Internet.

Figure 25: The SRX Series ClearPass Integration User Query Function



You can control when the SRX Series device sends its requests automatically by configuring the following two mechanisms:

- The **delay-query-time** parameter

  To determine the value to set for the **delay-query-time** parameter, it helps to understand the events and duration involved in how user identity information is transferred to the SRX Series device from ClearPass, and how the **delay-query-time** parameter influences the query process.

  A delay is incurred from when the CPPM initially posts user identity information to the SRX Series device using the Web API to when the SRX Series device can update its local ClearPass authentication table with that information. The user identity information must first pass through the ClearPass device's control plane and the control plane of the SRX Series device. In other words, this process can delay when the SRX Series device can enter the user identity information in its ClearPass authentication table.

  While this process is taking place, traffic might arrive at the SRX Series device that is generated by an access request from a user whose authentication and identity information is in transit from ClearPass to the SRX Series device.

  Rather than allow the SRX Series device to respond automatically by sending a user query *immediately*, you can set a **delay-query-time** parameter, specified in seconds, that allows the SRX Series device to wait for a period of time before sending the query.

After the delay timeout expires, the SRX Series device sends the query to the CPPM and creates a pending entry in the Routing Engine authentication table. During this period, the traffic matches the default policy and is dropped or allowed, depending on the policy configuration.

NOTE: If there are many query requests in the queue, the SRX Series device can maintain multiple concurrent connections to ClearPass to increase throughput. However, to ensure that ClearPass is not stressed by these connections, the number of concurrent connections is constrained to no more than 20 (<=20). You cannot change this value.

- A default policy, which is applied to a packet if the SRX Series device does not find an entry for the user associated with the traffic in its ClearPass authentication table.

  The system default policy is configured to drop packets. You can override this action by configuring a policy that specifies a different action to apply to this traffic.

Table 24 on page 267 shows the effect on the user query function in regard to whether or not Active Directory is enabled.

Table 24: Relationship Between User Query Function and Active Directory Authentication as Processed by the CLI

Active Directory Is Configured	ClearPass User Query Function Is Enabled	CLI Check Result
No	No	Pass
No	Yes	Pass
Yes	No	Pass
Yes	Yes	Fail

To avoid the failure condition reflected in the bottom row of the table, you must disable either Active Directory or the user query function. If both are configured, the system displays the following error message:

```
The priority of CP auth source is higher than AD auth source, and the CP user-query
 will shadow all AD features. Therefore, please choose either disabling CP
user-query or not configuring AD.
```

In its response to the user query request, the ClearPass web server returns information for the user's device whose IP address was specified in the request. This response includes a time stamp, which is expressed in UTC (Coordinated Universal Time) as defined by ISO 8601.

Here are some examples:

- 2016-12-30T09:30:10.678123Z
- 2016-12-30T09:30:10Z

- 2016-06-06T00:31:52-07:00

Table 25 on page 268 shows the components that comprise a timestamp format.

Table 25: Time Stamp Components as Defined by ISO 8601

Format Component	Meaning
YYYY	two-digit month
DD	two-digit day of month
hh	two-digits of hour (00 through 23)
mm	two-digits of minute
ss	two-digits of second
s	one or more digits representing a decimal fraction of a second
TZD	time zone designator: Z or +hh:mm or -hh:mm

Related Documentation

- Example: Configuring the Integrated ClearPass Authentication and Enforcement User Query Function on page 268
- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223
- Example: Configuring the SRX Series Integrated ClearPass Feature to Allow the Device to Receive User Authentication Data from ClearPass on page 226

Example: Configuring the Integrated ClearPass Authentication and Enforcement User Query Function

This example covers how to configure the SRX Series device to enable it to query Aruba ClearPass automatically for user authentication and identity information for an individual user when that information is not available.

NOTE: The user query function is supplementary to the Web API method of obtaining user authentication and identity information, and it is optional.

- Requirements on page 269
- Overview on page 269
- Configuration on page 272
- Verification on page 275

Requirements

This section defines the software and hardware requirements for the overall topology that includes user query requirements. See Figure 27 on page 272 for the topology. For details on the user query process, see Figure 26 on page 270.

The hardware and software components are:

- Aruba ClearPass (CPPM). The CPPM is configured to use its local authentication source to authenticate users.

  NOTE: **It is assumed that the CPPM is configured to provide the SRX Series device with user authentication and identity information, including the username, a list of the names of any groups that the user belongs to, the IP addresses of the devices used, and the device posture token.**

- SRX Series device running Junos OS that includes the integrated ClearPass feature.
- A server farm composed of six servers, all in the servers-zone:
  - marketing-server-protected (203.0.113.23)
  - human-resources-server (203.0.113.25)
  - accounting-server (203.0.113.72)
  - public-server (203.0.113.91)
  - corporate-server (203.0.113.71)
  - sales-server (203.0.113.81)
- AC 7010 Aruba Cloud Services Controller running ArubaOS.
- Aruba AP wireless access controller running ArubaOS.

  The Aruba AP is connected to the AC7010.

  Wireless users connect to the CPPM through the Aruba AP.
- Juniper Networks EX4300 switch used as the wired 802.1 access device.

  Wired users connect to the CPPM using the EX4300 switch.
- Six end-user systems:
  - Three wired network-connected PCs running Microsoft OS
  - Two BYOD devices that access the network through the Aruba AP access device
  - One wireless laptop running Microsoft OS

Overview

You can configure the user query function to enable the SRX Series device to obtain authenticated user identity information from the CPPM for an individual user when the

SRX Series device's ClearPass authentication table does not contain an entry for that user. The SRX Series device bases the query on the IP address of the user's device that generated the traffic issuing from the access request.

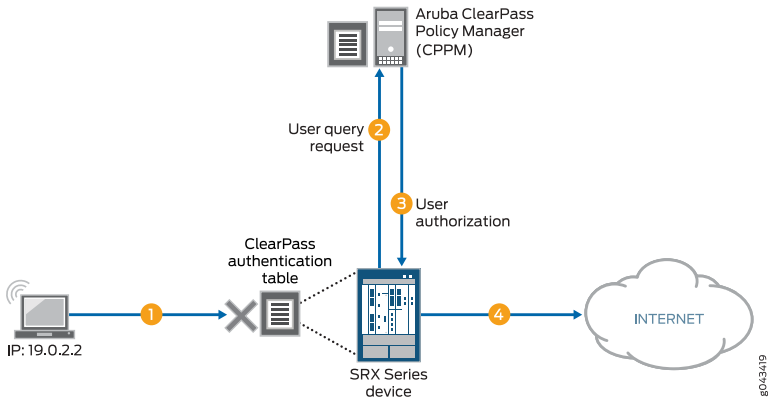
There are a number of reasons why the SRX Series device might not already have authentication information from the CPPM for a particular user. For example, it can happen that a user has not already been authenticated by the CPPM. This condition could occur if a user joined the network through an access layer that is not on a managed switch or WLAN.

The user query function provides a means for the SRX Series device to obtain user authentication and identity information from the CPPM for a user for whom the CPPM did not post that information to the SRX Series device using the Web API. When the SRX Series device receives an access request from a user for which there is not an entry in its ClearPass authentication table, it will automatically query the CPPM for it if this function is configured.

Figure 26 on page 270 shows the user query flow process, which encompasses the following steps:

1. A user attempts to access a resource. The SRX Series device receives the traffic requesting access. The SRX Series device searches for an entry for the user in its ClearPass authentication table, but none is found.
2. The SRX Series device requests authentication for the user from the CPPM.
3. The CPPM authenticates the user and returns the user authentication and identity information to the SRX Series device.
4. The SRX Series device creates an entry for the user in its ClearPass authentication table, and grants the user access to the Internet.

Figure 26: User Query Function Process



For details on the parameters that you can use to control when the SRX Series device issues the query, see "Understanding the Integrated ClearPass Authentication and Enforcement User Query Function" on page 265.

NOTE: You can also manually query the CPPM for authentication information for an individual user when this feature is configured.

The ClearPass endpoint API requires use of OAuth (RFC 6749) to authenticate and authorize access to it. For the SRX Series device to be able to query the CPPM for individual user authentication and authorization information, it must acquire an access token. For this purpose, the SRX Series device uses the Client Credentials access token grant type, which is one of the two types that ClearPass supports.

As administrator of the ClearPass Policy Manager (CPPM), you must create an API client on the CPPM with the grant_type set to “client_credentials”. You can then configure the SRX Series device to use that information to obtain an access token. Here is an example of the message format for doing this:

```
curl https://{$Server}/api/oauth – – insecure – – data
“grant_type=client_credentials&client_id=Client2&client_secret=
  m2Tvcklsi9je0kH9UTwuXQwlutKLC2obaDL54/fC2DzC"
```

A successful request from the SRX Series device to obtain an access token results in a response that is similar to the following example:

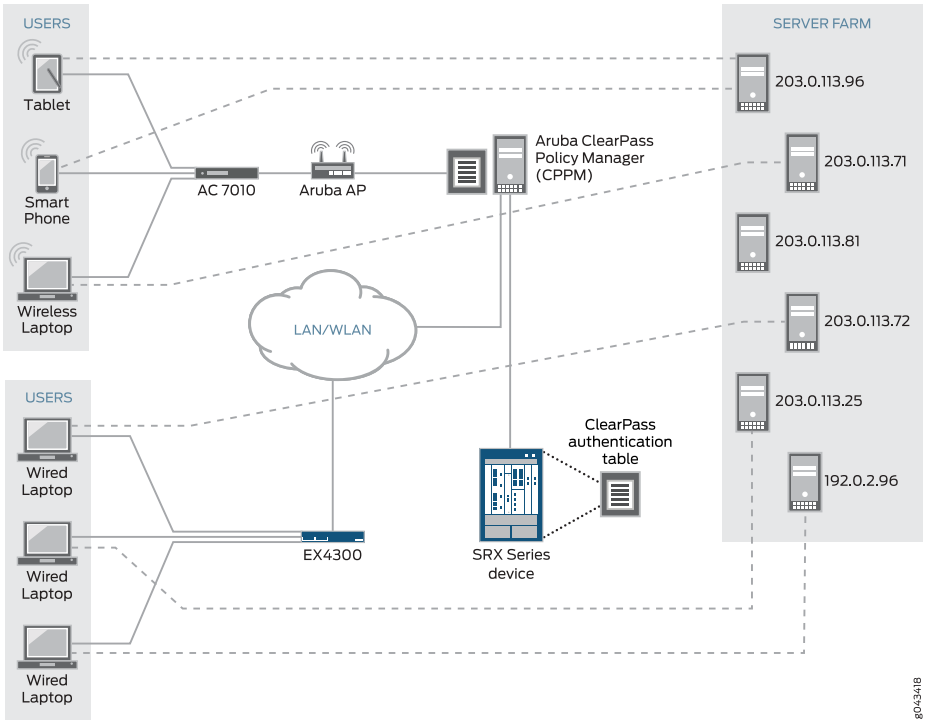
```
{
“access_token”:”ae79d980adf83ecb8e0eaca6516a50a784e81a4e”,
“expires_in”:2880,
“token_type”:”Bearer”,
“scope”=nu;
}
```

Before the access token expires, the SRX Series device can obtain a new token using the same message.

Topology

Figure 27 on page 272 shows the overall topology for this deployment, which encompasses the user query environment.

Figure 27: Topology for the Overall Deployment that Includes User Query



Configuration

To enable and configure the user query function, perform these tasks:

- Configure the User Query Function (Optional) on page 273
- Manually Issuing a Query to the CPPM for Individual User Authentication Information (Optional) on page 275

CLI Quick Configuration

To quickly configure this example, copy the following statements, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the statements into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set services user-identification authentication-source aruba-clearpass user-query
    web-server cp-webserver address 192.0.2.199
set services user-identification authentication-source aruba_clearpass user-query
    ca-certificate RADUISServerCertificate.crt
set services user-identification authentication-source aruba-clearpass user-query client-id
    client-1
set services user-identification authentication-source aruba-clearpass user-query
    client-secret 7cTr13#
set services user-identification authentication-source aruba-clearpass user-query token-api
    “api/aouth”
set services user-identification authentication-source aruba-clearpass user-query IP
    addres“api/vi/insight/endpoint/ip/$IP$”
```

Configure the User Query Function (Optional)

Step-by-Step Procedure

Configure the user query function to allow the SRX Series device to connect automatically to the ClearPass client to make requests for authentication information for individual users.

The user query function supplements input from the CPPM sent using the Web API. The Web API daemon does not need to be enabled for the user query function to work. For the user query function, the SRX Series device is the HTTP client. By it sends HTTPS requests to the CPPM on port 443.

To enable the SRX Series device to make individual user queries automatically:

1. Configure Aruba ClearPass as the authentication source for user query requests, and configure the ClearPass webserver name and its IP address. The SRX Series device requires this information to contact the ClearPass webserver.

   NOTE: You must specify aruba-clearpass as the authentication source.

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba-clearpass user-query web-server
     cp-webserver address 192.0.2.199
   ```

   NOTE: You can configure only one ClearPass webserver.

   Optionally, configure the port number and connection method, or accept the following values for these parameters. This example assumes the default values.

   - connect-method (default is HTTPS)
   - port (by default, the SRX Series device sends HTTPS requests to the CPPM on port 443

   However, if you were to explicitly configure the connection method and port, you would use these statements:

   ```
   set services user-identification authentication-source aruba-clearpass user-query
     web-server cp-webserver connect method <https/http>
   set services user-identification authentication-source aruba-clearpass user-query
     web-server cp-webserver port port-number
   ```

2. (Optional) Configure the ClearPass CA certificate file for the SRX Series device to use to verify the ClearPass webserver. (The default certificate is assumed if none is configured.)

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba_clearpass user-query ca-certificate
     RADUISServerCertificate.crt
   ```

The ca-certificate enables the SRX Series device to verify the authenticity of the ClearPass webserver and that it is trusted.

Before you configure the certificate, as administrator of the ClearPass device you must take the following actions:

- Export the ClearPass webserver's certificate from CPPM and import the certificate to the SRX Series device.
- Configure the ca-certificate as the path, including its CA filename, as located on the SRX Series device. In this example, the following path is used:

  `/var/tmp/RADUISServerCertificate.crt`

3. Configure the client ID and the secret that the SRX Series device requires to obtain an access token required for user queries.

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba-clearpass user-query client-id client-1
   user@host# set authentication-source aruba-clearpass user-query client-secret
     7cTr13#
   ```

   The client ID and the client secret are required values. They must be consistent with the client configuration on the CPPM.

   TIP: When you configure the client on the CPPM, copy the client ID and secret to use in the SRX Series device configuration.

4. Configure the token API that is used in generating the URL for acquiring an access token.

   NOTE: You must specify the token API. It does not have a default value.

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba-clearpass user-query
     token-api "api/aouth"
   ```

   In this example, the token API is api/oauth. It is combined with the following information to generate the complete URL for acquiring an access token https://192.0.2.199/api/oauth

   - The connection method is HTTPS.
   - In this example, the IP address of the ClearPass webserver is 192.0.2.199.

5. Configure the query API to use for querying individual user authentication and identity information.

   ```
   [edit services user-identification]
   user@host# set authentication-source aruba-clearpass user-query query-api
     'api/vi/insight/endpoint/ip/$IP$'
   ```

In this example, the query-api is **api/vi/insight/endpoint/ip/IP**. It is combined with the URL **https://192.0.2.199/api/oauth** resulting in **https://192.0.2.199/api/oauth/api/vi/insight/endpoint/ip/IP**.

The $IP variable is replaced with the IP address of the end-user's device for the user whose authentication information the SRX Series is requesting.

6. Configure the amount of time in seconds to delay before the SRX Series device sends the individual user query.

```
[edit services user-identification]
user@host# set authentication-source aruba-clearpass user-query delay-query-time
  10
```

Manually Issuing a Query to the CPPM for Individual User Authentication Information (Optional)

Step-by-Step Procedure

- Configure the following statement to manually request authentication information for the user whose device's IP address is 203.0.113.46.

```
root@device>request service user-identification authentication-source
  aruba-clearpass user-query address 203.0.113.46
```

Verification

Use the following procedures to verify that the user query function is behaving as expected:

- Verifying That the ClearPass Webserver Is Online on page 275
- Enabling Trace and Checking the Output on page 275
- Determining If the User Query Function Is Executing Normally on page 276
- Determining If a Problem Exists by Relying on User Query Counters on page 276

Verifying That the ClearPass Webserver Is Online

Purpose Ensure that the ClearPass webserver is online, which is the first mean of verifying that the user query request can complete successfully.

Action Enter the **show service user-identification authentication-source authentication-source user-query status** command to verify that ClearPass is online.

```
show service user-identification authentication-source aruba-clearpass user-query
status

Authentication source: aruba-clearpass
Web server Address: 192.0.2.199
Status: Online
Current connections: 0
```

Enabling Trace and Checking the Output

Purpose Display in the trace log any error messages generated by the user query function.

Action Set the trace log file name and enable trace using the following commands:

```
set system services webapi debug-log trace-log-1
set services user-identification authentication-source aruba-clearpass traceoptions flag user-query
```

Determining If the User Query Function Is Executing Normally

Purpose Determine if there is a problem with user query function behavior.

Action Check syslog messages to determine if the user query request failed.

If it failed, the following error message is reported:

```
LOG1: sending user query for IP <ip-address> to ClearPass web server failed.
:reason
```

The reason might be “server unconnected” or “socket error”.

Determining If a Problem Exists by Relying on User Query Counters

Purpose Display the user query counters to home in on the problem, if one exists, by entering the **show service user-identification authentication-source** *authentication-source* **user-query counters** command.

NOTE: The timestamp returned by ClearPass in response to the user query request can be specified in any of the ISO 8601 formats, including the format that includes a time zone.

Action

```
show service user-identification authentication-source aruba-clearpass user-query
counters

Authentication source: aruba-clearpass

    Web server Address: Address: ip-address
    Access token: token-string
    RE quest sent number: counter
    Routing received number: counter
    Time of last response: timestamp
```

Related Documentation

- Understanding the Integrated ClearPass Authentication and Enforcement User Query Function on page 265
- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223

CHAPTER 24

Configuring the Integrated ClearPass Authentication Threat and Attack Function

Understanding How the Integrated ClearPass Feature Detects Threats and Attacks and Notifies the CPPM

The integrated ClearPass authentication and enforcement feature allows you to integrate your SRX Series device with the ClearPass Policy Manager (CPPM) to obtain authenticated user identity information. It also allows the SRX Series device to send attack and threat logs to the CPPM. This topic focuses on sending attack and threat logs to the CPPM.

When the SRX Series device features detect threat and attack events, the event is recorded in the SRX Series device event log. The SRX Series device uses syslog to forward the logs to the CPPM. The CPPM can evaluate the logs and take action based on matching conditions. As administrator of ClearPass, you can use the information from the SRX Series device and define appropriate actions on the CPPM to harden your security.

Junos OS on the SRX Series device generates over 100 different types of log entries issued by more than 10 of its modules. Among the SRX Series device features that generate threat and attack logs are SCREENS, IDP, and UTM. To avoid overburdening the SRX Series device and the log server, the integrated ClearPass feature allows you to configure the SRX Series device to send to the CPPM only attack and threat log entries that were written to the event log in response to activity detected by the SCREENS, IDP, and UTM security features.

You can set the following conditions to control the log transmission:

- A log stream filter to ensure that only threat and attack logs are sent.
- A rate limiter to control the transmission volume. The SRX Series device log transmission will not exceed the rate-limiting conditions that you set.

For the CPPM to analyze the log information that the SRX Series sends to it, the content must be formatted in a standard, structured manner. The SRX Series log transmission follows the syslog protocol, which has a message format that allows vendor-specific extensions to be provided in a structured way.

Here is an example of an attack log generated by IDP:

```
<14>1 2014-07-24T1358.362+08:00 bjsolar RT_IDP - IDP_ATTACK_LOG_EVENT
[junos@2636.1.1.1.2.86 epoch-time="1421996988" message-type="SIG"
source-address="192.0.2.66" source-port="32796" destination-address="192.0.2.76"
 destination-port="21" protocol-name="TCP" service-name="SERVICE_IDP"
application-name="NONE" rule-name="1" rulebase-name="IPS" policy-name="idpengine"
 export-id="4641"repeat-count="0" action="NONE" threat-severity="MEDIUM"
attack-name="FTPROOT" nat-source-address="0.0.0.0" nat-source-port="0"
nat-destination-address="0.0.0.0" nat-destination-port="0" elapsed-time="0"
inbound-bytes="0" outbound-bytes="0" inbound-packets="0" outbound-packets="0"
source-zone-name="untrust" source-interface-name="ge-0/0/1.0"
destination-zone-name="trust" destination-interface-name="ge-0/0/7.0"
packet-log-id="0" alert="no" username="N/A" roles="N/A" message="-"]
```

Table 26 on page 278 uses the content of this example IDP attack log to identify the parts of an attack log entry. See "SRX Series Threat and Attack Logs Sent to Aruba ClearPass" on page 279 for further details on types of attack and threat logs.

Table 26: Attack Log Fields Using Example Log

Log Entry Component	Meaning	Format	Example
Priority	pri = LOG_USER + severity. Version is always 1	pri *version*	<14>1
Time and Time Zone	When the log was recorded and in what time zone.	*y-m-dThs.ms+time zone* • y = year • m=month • d = day • T+hours	2014-07-24T1358.362+08:00
Device/Host Name	Name of the device from which the event log was sent. This value is configured by the user.	string, *hostname*	bjsolar
Service Name	SRX Series feature that issued the event log.	string *service*	SERVICE_IDP
Application Name	Application that generated the log entry.	string *application-name*	NONE
PID	Process ID. The process ID is not meaningful in this context, so *pid* is replaced by "-". The value "-" is a placeholder for process ID.	*pid*	-

Table 26: Attack Log Fields Using Example Log *(continued)*

Log Entry Component	Meaning	Format	Example
Errmsg Tag	Log ID name, error message tag.	string, *log-name and tag*	IDP_ATTACK_LOG_EVENT
Errmsg Tag Square Bracket	Log content enclosed in square brackets.	[ ]	-
OID	Product ID provided by the chassis daemon (chassisd).	junos@oid	junos@2636.1.1.1.2.86
Epoch Time	The time when the log was generated after the epoch.	*number*	1421996988

Related Documentation

- SRX Series Threat and Attack Logs Sent to Aruba ClearPass on page 279
- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223
- Example: Configuring the Integrated ClearPass Authentication and Enforcement User Query Function on page 268

SRX Series Threat and Attack Logs Sent to Aruba ClearPass

The SRX Series integrated ClearPass authentication and enforcement feature collaborates with Aruba ClearPass in protecting a company's resources against potential and actual attacks through use of attack and threat event logs. These logs that are generated by the SRX Series SCREENS, IDP, and UTM components clearly identify the types of attacks and threats that threaten a company's network security.

The SRX Series device filters from the overall log entries the logs that report on threat and attack events, and it forwards these log entries to the ClearPass Policy Manager (CPPM) to be used in assessing and enforcing the company's security policy. The SRX Series device transmits the logs in volumes determined by the rate-limiting conditions that you set.

Table 27 on page 279 identifies the types of threat and attack log entries and the events that they represent.

Table 27: Threat and Attack Log Entries Generated by SRX Series Components

Log Type	Description
RT_SCREEN_ICMP	ICMP attack
RT_SCREEN_ICMP_LS	
RT_SCREEN_IP	IP attack
RT_SCREEN_IP_LS	

Table 27: Threat and Attack Log Entries Generated by SRX Series Components *(continued)*

Log Type	Description
RT_SCREEN_TCP	TCP attack
RT_SCREEN_TCP_LS	
RT_SCREEN_TCP_DST_IP	TCP destination IP attack
RT_SCREEN_TCP_DST_IP_LS	
RT_SCREEN_TCP_SRC_IP	TCP source IP attack
RT_SCREEN_TCP_SRC_IP_LS	
RT_SCREEN_UDP	UDP attack
RT_SCREEN_UDP_LS	
AV_VIRUS_DETECTED_MT	Virus infection A virus was detected by the antivirus scanner.
AV_VIRUS_DETECTED_MT_LS	
ANTISPAM_SPAM_DETECTED_MT	spam The identified e-mail was detected to be spam.
ANTISPAM_SPAM_DETECTED_MT_LS	
IDP_APPDDOS_APP_ATTACK_EVENT	Application-level distributed denial of service (AppDDoS) attack The AppDDoS attack occurred when the number of client transactions exceeded the user-configured connection, context, and time binding thresholds.
IDP_APPDDOS_APP_ATTACK_EVENT_LS	
IDP_APPDDOS_APP_STATE_EVENT	AppDDoS attack The AppDDoS state transition occurred when the number of application transactions exceeded the user-configured connection or context thresholds.
IDP_APPDDOS_APP_STATE_EVENT_LS	
IDP_ATTACK_LOG_EVENT	Attack discovered by IDP IDP generated a log entry for an attack.
IDP_ATTACK_LOG_EVENT_LS	

Example: Configuring Integrated ClearPass to Filter and Rate-limit Threat and Attack Logs

The SRX Series device can dynamically send to the ClearPass Policy Manager (CPPM) information about threats and attacks identified by its security modules that protect network resources. It detects attack and attack threats that pertain to the activity of specific devices and their users, and it generates corresponding logs. To control this

transmission, you must configure the type of logs to be sent and the rate at which they are sent. You can then use this information in setting policy rules on the CPPM to harden your network security.

This example shows how to configure the SRX Series integrated ClearPass authentication and enforcement feature to filter and transmit only threat and attack logs to the CPPM and to control the volume and rate at which the SRX Series device transmits them.

Requirements

The topology for this example uses the following hardware and software components:

- Aruba CPPM implemented in a virtual machine (VM) on a server. The CPPM is configured to use its local authentication source to authenticate users.
- SRX Series device running Junos OS that includes the integrated ClearPass feature. The SRX Series device is connected to the Juniper Networks EX4300 switch and to the Internet. The SRX Series device communicates with ClearPass over a secure connection.
- Juniper Networks EX4300 switch used as the wired 802.1 access device. The EX4300 Layer 2 switch connects the endpoint users to the network. The SRX Series device is connected to the switch.
- Wired, network-connected PC running Microsoft OS. The system is directly connected to the EX4300 switch.

  Threat and attack logs are written for activity from these devices triggered by events that the security features catch and protect against.

Overview

The SRX Series integrated ClearPass authentication and enforcement feature participates with Aruba ClearPass in protecting your company's resources against actual and potential attacks. The SRX Series device informs the CPPM about threats to your network resources and attacks against them through logs that it sends. You can then use this information to assess configuration of your security policy on the CPPM. Based on this information, you can harden your security in regard to individual users or devices.

To control the behavior of this feature, you must configure the SRX Series device to filter for attack and threat log entries and set rate-limiting conditions.

You can tune the behavior of this function in the following ways:

- Set a filter to direct the SRX Series device to send only threat and attack logs to the CPPM. This filter allows you to ensure that the SRX Series device and the log server do not need to handle irrelevant logs.
- Establish rate limit conditions to control the volume of logs that are sent.

You set the rate-limit parameter to control the volume and rate that logs are sent. For example, you can set the rate-limit parameter to 1000 to specify that a maximum of 1000 logs are sent to ClearPass in 1 second. In this case, if there is an attempt to send 1015 logs, the number of logs over the limit—15 logs, in this case—would be dropped. The logs are not queued or buffered.

You can configure a maximum of three log streams with each individual log defined by its destination, log format, filter, and rate limit. Log messages are sent to all configured log streams. Each stream is individually rate-limited.

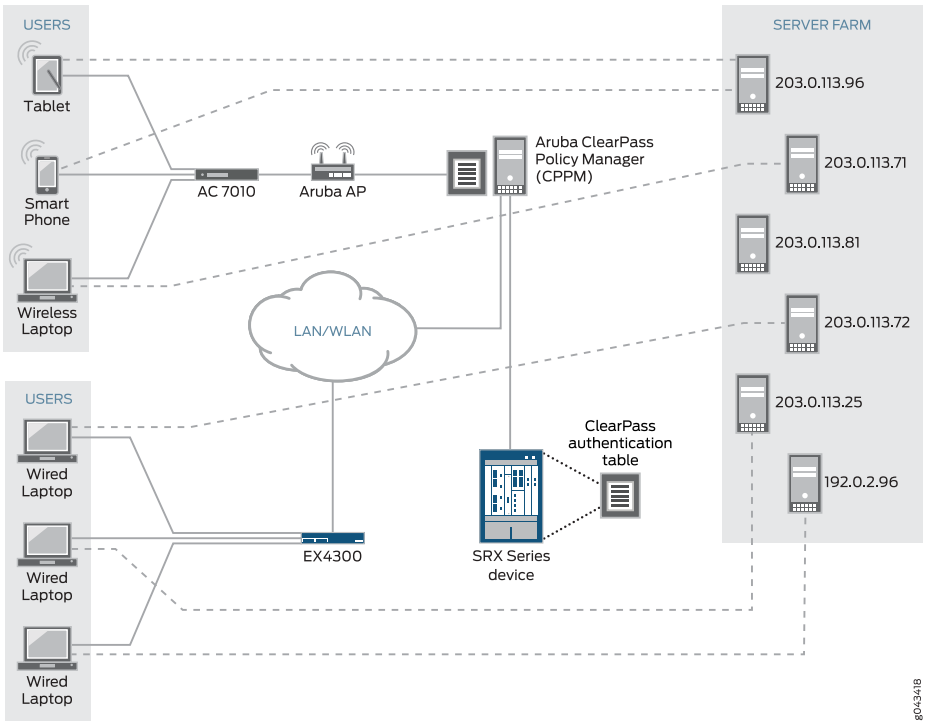
NOTE: To support rate-limiting on high-end platforms, log messages are sent out from the device's local SPU at a divided rate. In the configuration process, the Routing Engine assigns a divided rate to each SPU. The divided rate is equal to the configured rate divided by the number of SPUs on the device:

divided-rate = configured-rate/number-of-SPUs

Topology

Figure 28 on page 282shows the topology for this example.

Figure 28: Integrated ClearPass Authentication and Enforcement Deployment Topology



Configuration

This example covers how to configure a filter to select threat and attack logs to be sent to ClearPass. It also covers how to set a rate limiter to control the volume of logs sent during a given period. It includes these parts:

CLI Quick Configuration

To quickly configure this example, copy the following statements, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the statements into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security log stream threat-attack-logs host 203.0.113.47
set security log mode stream
set security log source-interface ge-0/0/1.0
set security log stream to_clearpass format sd-syslog
set security log stream to_clearpass filter threat-attack
set security log stream to_clearpass rate-limit 1000
```

Configuring Integrated ClearPass Authentication and Enforcement to Filter for Threat and Attack Logs Sent to the CPPM

Step-by-Step Procedure

1. Specify a name for the log stream and the IP address of its destination.

   ```
   [edit security]
   user@host# set security log stream threat-attack-logs host 203.0.113.47
   ```

2. Set the log mode to stream.

   ```
   [edit security]
   user@host# set log mode stream
   ```

3. Set the host source interface number.

   ```
   [edit security]
   user@host#set log source-interface ge-0/0/1.0
   ```

4. Set the log stream to use the structured syslog format for sending logs to ClearPass through syslog.

   ```
   [ edit security]
   user@host# set log stream to_clearpass format sd-syslog
   ```

5. Specify the type of events to be logged.

   ```
   [edit security]
   user@host# set log stream to_clearpass filter threat-attack
   ```

NOTE: This configuration is mutually exclusive in relation to the current category set for the filter.

6. Set rate limiting for this stream. The range is from 1 through 65,535.

   This example specifies that up to 1000 logs per second can be sent to ClearPass. When the maximum is reached, any additional logs are dropped.

   ```
   [ edit security]
   user@host# set log stream to_clearpass rate-limit 1000
   ```

Results

From configuration mode, confirm your configuration for interfaces by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
mode stream;
source-interface ge-0/0/1.0;
stream threat-attack-logs {
  host {
    203.0.113.47;
  }
}
stream to_clearpass {
  format sd-syslog;
  filter threat-attack;
  rate-limit {
    1000;
  }
}
```

Related Documentation

- Understanding How the Integrated ClearPass Feature Detects Threats and Attacks and Notifies the CPPM on page 277
- SRX Series Threat and Attack Logs Sent to Aruba ClearPass on page 279
- Understanding the SRX Series Integrated ClearPass Authentication and Enforcement Feature on page 219
- Understanding How ClearPass Initiates a Session and Communicates User Authentication Information to the SRX Series Device Using the Web API on page 223

PART 6

Configuration Statements and Operational Commands

CHAPTER 25

Configuration Statements

actions (Services SSL Proxy)

Syntax

```
actions {
  crl {
    disable;
    if-not-present (allow | drop);
    ignore-hold-instruction-code;
  }
  disable-session-resumption;
  ignore-server-auth-failure;
  logs {
    all;
    errors;
    info;
    sessions-allowed;
    sessions-dropped;
    sessions-ignored;
    sessions-whitelisted;
    warning;
  }
  renegotiation {
    (allow | allow-secure | drop);
  }
}
```

Hierarchy Level [edit services ssl proxy profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. The **crl** statement is supported from 15.1X49-D30.

Description Specify the logging and traffic related actions.

Options

- **crl**—Specify the certificate revocation actions.
  - **disable**—Disable CRL verification.
  - **if-not-present**—Specify actions for sessions.
    - **allow**—Allow sessions when CRL information is not available.
    - **drop**—Drop sessions when CRL information is not available.
  - **ignore-hold-instruction-code**—Ignore the unconfirmed (on hold) revocation status, and accept a certificate.
- **disable-session-resumption**—Disable session resumption.
- **ignore-server-auth-failure**—Ignore server authentication failure.
- **log**—Specify the logging actions.
  - **all**—Log all events.
  - **errors**—Log all error events.
  - **info**—Log all information events.

- **sessions-allowed**—Log SSL session allowed events after an error.
- **sessions-dropped**—Log only SSL session dropped events.
- **sessions-ignored**—Log session ignored events.
- **sessions-whitelisted**—Log SSL session whitelisted events.
- **warning**—Log all warning events.

- **renegotiation**—Specify the renegotiation options.
  - **allow**—Allow secure and nonsecure renegotiation.
  - **allow-secure**—Allow secure negotiation only.
  - **drop**—Drop session on renegotiation request.

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- SSL Proxy Overview on page 13
- Configuring SSL Proxy on page 23
- Enabling Debugging and Tracing for SSL Proxy on page 33

active-directory-access

Syntax

```
active-directory-access {
  domain domain-name {
    user username;
      password password;
    domain-controller domain-controller-name {
      address domain-controller-address;
    }
    ip-user-mapping {
      discovery-method {
        wmi {
          event-log-scanning-interval seconds;
          initial-event-log-timespan hours;
        }
      }
    }
    user-group-mapping {
      ldap {
        authentication-algorithm {
          simple;
        }
        ssl;
        base base;
        user name {
          password password;
        }
        address ip-address {
          port port;
        }
      }
    }
  }
}
```

Hierarchy Level [edit services user-identification]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description Identify the domain and domain controllers where the integrated user firewall feature is implemented; configure the IP address-to-user mapping information and the user-to-group mapping information for accessing the LDAP server.

Options

domain *domain-name*—Required. Name of the domain; the length of the name ranges from 1 through 64 characters. The SRX Series device can have the integrated user firewall feature configured in a maximum of two domains.

user *username*—Required. Active Directory account name.

Range: 1 through 64 characters.

password *password*—Required. Password of the Active Directory account.

Range: 1 through 128 characters.

domain-controller ***domain-controller-name***—Required. Name of the domain controller; the length of the name can range from 1 through 64 characters. A maximum of 10 domain controllers can be configured.

address ***domain-controller-address***—Required. IP address of the domain controller.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- user-identification (Services) on page 430
- LDAP Functionality in Integrated User Firewall on page 150

active-directory-authentication-table

Syntax

```
active-directory-authentication-table {
    priority priority;
}
```

Hierarchy Level [edit security user-identification authentication-source]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description An authentication table is generated by polling Active Directory domain controllers for source identity information about active users. Each entry in the table correlates an authenticated user with an IP address and associated user groups. That information is used for matching in IP-based firewall policies. The user information must be retrieved from the table before policy lookup can proceed and traffic is allowed to pass through the firewall.

Options

priority *priority*—Specify the priority of the Active Directory authentication table. The priority determines the sequence for searching among various other authentication tables to retrieve a user role. The priorities of the following tables are considered: local authentication table, firewall authentication table, Active Directory authentication table, and UAC authentication table.

Each authentication table is given a unique priority value. The lower the value, the higher the priority. A table with priority 120 is searched before a table with priority 200. Setting the priority value of a table to 0 is equivalent to disabling the table and eliminating it from the search sequence.

Range: A unique value from 0 through 65535.

Default: The default priority of the Active Directory authentication table is 125.

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

Related Documentation

- *authentication-source (Security)*
- Overview of Integrated User Firewall on page 141
- *Understanding User Role Firewalls*
- *Understanding the User Identification Table*

address (Services)

Syntax address *ip-address*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the IP address of the IC Series device with which the SRX Series devices should communicate.

This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation

- Understanding Unified Access Control on page 101
- Acquiring User Role Information from an Active Directory Authentication Server on page 101

address (Services User Identification)

Syntax address (*ip-address* | *hostname*);

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query web-server]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure for the integrated ClearPass authentication and enforcement feature the address of the ClearPass webserver that the SRX Series device communicates with. The SRX Series device requests user authentication and identity information for an individual user from the ClearPass webserver whose address is configured. If you configure the user query function, the SRX Series device can obtain this information for a specific user when it does not receive it from the ClearPass Policy Manager through Web API POST requests.

Required Privilege Level services—To view this statement in the configuration

services-control—To add this statement to the configuration.

admin-search

Syntax

```
admin-search {
    distinguished-name distinguished-name;
    password password;
}
```

Hierarchy Level [edit access ldap-options search],
[edit access profile *profile-name* ldap-options search]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify that a Lightweight Directory Access Protocol (LDAP) administrator search is performed. To perform an administrator search, you must specify administrator credentials, which are used in the bind as part of performing the search.

Options The remaining statements are explained separately.

Default: Anonymous search. To perform an administrator search, you must specify administrator credentials, which are used in the bind as part of performing the search.

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

application (Security Policies)

Syntax

```
application {
  [application];
  any;
}
```

Hierarchy Level

[edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* match]

[edit security policies global policy *policy-name* match]

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Specify the IP or remote procedure call (RPC) application or set of applications to be used as match criteria.

Options

application-name-or-set—Name of the predefined or custom application or application set used as match criteria.

any—Any predefined or custom applications or application sets.

NOTE: A custom application that does not use a well-known destination port for the application will not be included in the any option, and must be named explicitly.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- *Security Policies Overview*

application-services (Security Policies)

Syntax

```
application-services {
  application-firewall {
    rule-set rule-set-name;
  }
  application-traffic-control {
    rule-set rule-set-name;
  }
  gprs-gtp-profile profile-name;
  gprs-sctp-profile profile-name;
  idp;
  redirect-wx | reverse-redirect-wx;
  ssl-proxy {
    profile-name profile-name;
  }
  uac-policy {
    captive-portal captive-portal;
  }
  utm-policy policy-name;
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit]

Release Information Statement modified in Junos OS Release 11.1.

Description Enable application services within a security policy.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Application Firewall Overview*

assemble

Syntax

```
assemble {
  common-name common-name;
}
```

Hierarchy Level [edit access ldap-options],
[edit access profile *profile-name* ldap-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify that a user's LDAP distinguished name (DN) is assembled through the use of a common name identifier, the username, and base distinguished name.

Options **common-name** *common-name*—Common name identifier used as a prefix for the username during the assembly of the user's distinguished name. For example, **uid** specifies " user id," and **cn** specifies "common name."

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

authentication-entry-timeout (Services User Identification)

Syntax authentication-entry-timeout *minutes*;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure for the integrated ClearPass authentication and enforcement feature the timeout interval after which idle entries in the ClearPass authentication table expire.

Options *minutes*—–Timeout interval. The timout interval begins from when the user authentication entry is added to the ClearPass authentication table. If a value of 0 is specified, the entries will never expire.

Range: 10 through 1440 minutes

Default: 30 minutes

Required Privilege Level services—To view this statement in the configuration

services-control—To add this statement to the configuration.

authentication-source (Services User Identification)

Syntax

```
authentication-source name {
  authentication-entry-timeout minutes;
  no-user-query;
  traceoptions {
      file {
        filename;
        files number;
        match regular-expression;
        size maximum-file-size;
        (world-readable |no-world-readable);
      }
      flag flag;
      level level ;
      no-remote-trace;
    }
  }
}
```

Hierarchy Level [edit services user-identification]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure ClearPass as the authentication source for the integrated ClearPass authentication and enforcement feature. You must specify aruba-clearpass as the value of authentication source *name*, followed by its defining characteristics. You cannot specify the authentication source alone—that is, apart from its configuration parameters that qualify it.

The ClearPass Policy Manager (CPPM), as the authentication source and client of the SRX Series device HTTP server, initiates a connection to the SRX Series device using the Web API that the SRX Series device exposes to it. The CPPM sends user authentication and identity information to the SRX Series device across this connection using HTTP or HTTPS POST request messages.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

authentication-source (Services User Identification Device Identity)

Syntax authentication-source *authentication-source* (active-directory | network-access-controller)

Hierarchy Level [edit services user-identification device-information]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Specify the device identity authentication source. The integrated user firewall device identity authentication feature enables you to control access to resources based on the identity of the device and not that of the user of the device. Supported authentication sources include Active Directory and third-party network access systems.

The SRX Series device obtains the device identity information for authenticated devices from the authentication source. After the SRX Series device obtains the device information, it creates a device identity authentication table to use to store device identity entries.

The SRX Series device searches the device identity authentication table for a device match when traffic issuing from a user's device arrives at the SRX Series device. If it finds a match, the SRX Series device searches for a matching security policy. If it finds a matching security policy, the security policy's action is applied to the traffic.

Options

active-directory—Specifies Microsoft Active Directory as the authentication source.

The SRX Series device obtains the device identity information for authenticated devices from Active Directory. It reads the Active Directory domain controller event logs to obtain the IP addresses of devices logged into the domain and authenticated by Windows. Then, for each authenticated device, it obtains from the Active Directory LDAP server the names of the groups to which the device belongs, based on the IP addresses of the devices.

network-access-controller—Specifies the authentication source as that of a third-party network access controller (NAC) system. If your network environment is configured for a NAC solution and you decide to take this approach, the NAC system sends the device identity information to the SRX Series device. The SRX Series device exposes a RESTful Web services API implementation that enables you to send the device identity information to the SRX Series device in a formal XML structure. If you take this approach, you must verify that your NAC solution works with the SRX Series device.

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- Understanding Access Control to Network Resources Based on Device Identity Information on page 185
- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192

banner (Access FTP HTTP Telnet Authentication)

Syntax

```
banner {
  fail string;
  login string;
  success string;
}
```

Hierarchy Level [edit access firewall-authentication pass-through (ftp | http | telnet)]

Release Information Statement introduced in Junos OS Release 8.5.
HTTPS for Web authentication is supported on SRX Series Services Gateways from Junos OS Release 12.1X44-D10 and on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways from Junos OS Release 15.1X49-D40.

Description Configure the banners that appear to users during the FTP, HTTP, HTTPS, and Telnet firewall authentication process. The banners appear during login, after successful authentication, and after failed authentication.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Understanding Firewall Authentication Banner Customization on page 83

banner (Access Web Authentication)

Syntax

```
banner {
  success string;
}
```

Hierarchy Level [edit access firewall-authentication web-authentication]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure the banner that appears to users during the Web authentication process. The banner appears during login, after successful authentication, and after failed authentication.

Options The remaining statements are explained separately. See CLI Explorer.

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

base-distinguished-name

Syntax base-distinguished-name *base-distinguished-name*;

Hierarchy Level [edit access ldap-options],
[edit access profile*profile-name* ldap-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the base distinguished name (DN), which can be used in one of the following ways:

- If you are using the **assemble** statement so that the user's distinguished name is being assembled, the base distinguished name is appended to a username to generate the user's distinguished name. The resulting distinguished name is used in the LDAP bind call.
- If you are using the **search** statement so that the user's distinguished name is found by a search, the search is restricted to the subtree of the base distinguished name.

Options *base-distinguished-name*—Series of basic properties that define the user. For example in the base distinguished name **o=juniper, c=us**, where **c** stands for country, and **o** for organization.

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

ca-certificate (Services User Identification)

Syntax ca-certificate *certificate-file*;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query https]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specifies the certificate file that the SRX Series device uses to verify the Clearpass server's certificate for the SSL connection that is used for the user query function. As the ClearPass administrator, you must export the server's certificate from the CPPM and import it to the SRX Series device. Afterward, you must configure the ca-certificate path and the certificate filename on the SRX Series device. Here is an example:

'/var/tmp/RADIUSServerCertificate.crt'

This configuration is part of the Integrated ClearPass Authentication and Enforcement feature user query function configuration. User query enables the SRX Series device to query the ClearPass Policy Manager (CPPM) for authentication and identity information for an individual user under certain circumstance when it does not receive that information from the CPPM though the Web API POST requests.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

ca-profile (Services)

Syntax ca-profile *ca-profile*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the certificate authority (CA) of the certificate that the SRX Series device should use in communications with an Infranet Enforcer. The SRX Series device uses the CA to validate the IC Series UAC Appliance server certificate.

Use this statement if you have loaded certificates from multiple certificate authorities (CAs) onto your SRX Series device and you need to configure the device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series appliance .

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

captive-portal (Services UAC)

Syntax

```
captive-portal redirect-policy-name{
  redirect-traffic (all | unauthenticated);
  redirect-url redirect-url;
}
```

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify the preconfigured security policy for captive portal on the Junos OS Enforcer to enable the captive portal feature. The captive portal policy is configured as part of the UAC policy.

By configuring the captive portal feature, you can redirect traffic destined for protected resources to the IC Series device or to the URL you configure on the Junos OS Enforcer.

Options The remaining statements are explained separately. See CLI Explorer.

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

captive-portal (Services UAC Policy)

Syntax

```
captive-portal captive-portal-policy-name;
```

Hierarchy Level

```
[edit security policies from-zone zone-name to-zone zone-name policy policy-name then
  permit application-services uac-policy]
```

Release Information Statement introduced in Junos OS Release 10.2.

Description Create the captive portal policy in the UAC security policy. You use the captive portal policy to configure the captive portal feature on the Junos OS Enforcer. By configuring the captive portal feature, you can redirect traffic destined for protected resources to the IC Series device or to the URL you configure on the Junos OS Enforcer.

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

Related Documentation
- *Security Policies Overview*

certficate (System Services)

Syntax certificate *certificate-filename*;

Hierarchy Level [edit system services webapi https]

Release Information Statement introduced in Junos OS release 12.3X48-D30.

Description Configures a custom certificate to be used for the Integrated ClearPass Authentication and Enforcement feature Web API (webapi) configuration when the HTTPS protocol is configured.

When you configure the Web API (webapi) function to use HTTPS, you can use the default certificate, a custom one, or a certificate generated by the PKI local store.

If you configure a custom certificate, you must configure a certificate key with it. Here is an example of how to configure a certificate and certificate key:

```
set system services webapi https certificate /var/tmp/certificate.crt
set system services webapi https certificate-key /var/tmp/certificate.key
```

NOTE: The Web API supports only the PEM format for the custom certificate and certificate key.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

certificate-key (System Services)

Syntax certificate-key *filename*;

Hierarchy Level [edit system services webapi https]

Release Information Statement introduced in Junos OS release 12.3X48-D30.

Description Configures the filename of the certificate key to use with the specified custom certificate for the Web API (webapi) HTTPS configuration. A certificate key is required if a custom certificate file is used.

NOTE: The Integrated ClearPass Authentication and Enforcement feature Web API supports only the PEM format for the custom certificate and certificate key.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

certificate-verification

Syntax certificate-verification [optional | required | warning]

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 12.1.

Description This option determines whether server certificate verification is required when initiating a connection between an SRX Series device and a Junos Pulse Access Control Service in a UAC configuration. If no CA profile contains the certificate authority (CA) that signed the configured server certificate for the Access Control Service, this option determines whether the commit check should fail, a warning should be displayed, or the connection should be made without any warning.

NOTE: For strict security, this option should be reset to required, and the proper CA certificate should be specified in the CA profile.

Options

- **optional**—Certificate verification is not required. If the CA certificate is not specified in the **ca-profile** option, the commit check passes and no warning is issued.
- **required**—Certificate verification is required. If the CA certificate is not specified in the **ca-profile** option, an error message is displayed, and the commit check fails. Use this option to ensure strict security.

Default: **warning**—Certificate verification is not required. A warning message is displayed during commit check if the CA certificate is not specified in the **ca-profile** option.

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

Related Documentation

- Understanding Communications Between the Junos OS Enforcer and the IC Series UAC Appliance on page 93
- *Understanding User Role Firewalls*

client (System Services)

Syntax client *ip-address*;

Hierarchy Level [edit system services webapi]

Release Information Statement introduced in Junos OS release 12.3X48-D30.

Description Configures the IP address of the client. For the Integrated ClearPass Authentication and Enforcement feature Web API daemon configuration, the client is the ClearPass Policy Manager (CPPM).

The SRX Series Web API daemon acts as an HTTP(S) server. The CPPM client sends POST request messages containing user authentication and identity information to the Web API daemon. The SRX Series device accepts information only from the configured address of the client.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

client-id (Services User Identification)

Syntax client-id *client-id*;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS release 12.3X48-D30.

Description Configures the client ID that the SRX Series device requires to obtain an access token for the Integrated ClearPass Authentication and Enforcement user query function. The client ID must be consistent with the API client configured on the CPPM.

The ClearPass endpoint API requires use of OAuth (RFC 6749) to authenticate and authorize the SRX Series device access. The SRX Series device uses the Client Credentials grant type access token, which is one of the two types that ClearPass supports.

If it is configured, the user query function allows the SRX Series device to query the CPPM for authentication and identity information about individual users when it does not receive this information from the CPPM through the SRX Series Web API daemon (webapi).

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

client-group

Syntax client-group [*group-names*];

Hierarchy Level [edit access profile *profile-name* client *client-name*]
[edit access profile *profile-name session-options*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify a list of client groups that the client belongs to. If the group list is not defined as part of the client profile, the client group configured at the **[edit access profile session-options]** hierarchy level is used.

Options *group-names* —Names of one or more groups the client belongs to, separated by spaces—for example **g1, g2, g3**. The total length of the group name string cannot exceed 256 characters.

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

client-idle-timeout (Access Profile)

Syntax client-idle-timeout *minutes*;

Hierarchy Level [edit access profile *profile-name* session-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the grace period that begins after an authenticated user terminates all sessions and connections. Authentication is not required if a new connection is initiated during the grace period by the same user.

Options *minutes* —Number of minutes of idle time that elapse before the session is terminated.

Range: 10 through 255 minutes

Default: 10 minutes

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

client-name-filter

Syntax

```
client-name-filter client-name {
    count number;
    domain-name domain-name;
    separator special-character;
}
```

Hierarchy Level [edit access profile *profile-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Define client-name-related restrictions. Clients whose names follow these restrictions are authenticated on the server.

Options *client-name*—Name of the client.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Junos OS Security Configuration Guide*

client-secret (Services User Identification)

Syntax client-secret *client-secret*;

Hierarchy Level [edit ervices user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS release 12.3X48-D30.

Description Configures the client secret used with the client ID that the SRX Series device requires to obtain an access token for the Integrated ClearPass Authentication and Enforcement user query function. The client secret must be consistent with the client secret configured on the CPPM.

The ClearPass endpoint API requires use of OAuth (RFC 6749) to authenticate and authorize SRX Series device access. The SRX Series device uses the Client Credentials grant type access token, which is one of the two types that ClearPass supports.

If it is configured, the user query function allows the SRX Series device to query the CPPM for authentication and identity information about individual users when it does not receive this information from the CPPM through the SRX Series Web API daemon (webapi).

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

client-session-timeout (Access Profile)

Syntax client-session-timeout *minutes*;

Hierarchy Level [edit access profile *profile-name* session-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the amount of time after which user sessions are terminated, regardless of user activity (also known as a forced or hard authentication timeout).

Options ***minutes*** —Number of minutes after which user sessions are terminated.

Range: 1 through 10,000 minutes

Default: Off

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

Related Documentation
- *Junos OS Security Configuration Guide*

configuration-file

Syntax *server-name* configuration-file *filepath*;

Hierarchy Level [edit access securid-server]

Release Information Statement introduced in Release 9.1 of Junos OS.

Description Specify the path of the SecurID server configuration file. The file is copied on the devices in some directory location—for example, **/var/db/securid/sdconf.rec**.

Options
- ***server-name***—Name of the SecurID authentication server.
- ***filepath***—Path of the SecurID server configuration file.

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

Related Documentation
- *Junos OS Security Configuration Guide*

connect-method (Services User Identification)

Syntax connect-method (http | https);

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query web-server]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the application protocol used for the SRX Series device connection to the ClearPass Policy Manager (CPPM) for user query requests. You identify the connection protocol as part of the configuration that identifies the CPPM server. The user query function allows the SRX Series device to request from the CPPM user authentication and identity information for an individual user.

Options **HTTP**—Protocol that the CPPM uses to connect to the SRX Series device.

HTTPS—Secure version of the protocol that the CPPM uses to connect to the SRX Series device.

Default: HTTPS—The connect-method configuration is optional. If it is not configured, HTTPS is assumed.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

count

Syntax count *number*;

Hierarchy Level [edit access profile *profile-name* client-name-filter *client-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the number of characters to be stripped from a client name, from right to left, until the specified number of characters are deleted. The resulting name is sent to the authentication server.

Options ***number***—Number of characters to be stripped in a client name.

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

Related Documentation

- *Junos OS Security Configuration Guide*

custom-ciphers

Syntax custom-ciphers [rsa-with-rc4-128-md5 RSA | 128bit rc4 | md5 hash rsa-with-rc4-128-sha RSA | 128bit rc4 |sha hash rsa-with-des-cbc-sha RSA | des cbc | sha hash rsa-with-3des-ede-cbc-sha RSA | 3des ede/cbc | sha hash rsa-with-aes-128-cbc-sha RSA | 128 bit aes/cbc | sha hash rsa-with-aes-256-cbc-sha RSA | 256 bit aes/cbc |sha hash rsa-export-with-rc4-40-md5 RSA-export | 40 bit rc4 | md5 hash rsa-export-with-des40-cbc-sha RSA-export | 40 bit des/cbc | sha hash rsa-with-null-md5 RSA | no symmetric cipher | md5 hash rsa-with-null-sha RSA | no symmetric cipher | sha hash];

Hierarchy Level [edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Display the custom cipher list. This statement is supported in the SRX550M, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

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

Related Documentation

- SSL Proxy Overview on page 13
- Configuring SSL Proxy on page 23
- Enabling Debugging and Tracing for SSL Proxy on page 33

debug-level (System Services)

Syntax debug-level *level*;

Hierarchy Level [edit system services webapi]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the trace level for the integrated ClearPass authentication and enforcement Web API daemon (webapi).

Options ***level***—A flag that specifies the type of logs to be written to the log file for the Web API daemon (webapi).

alert—Matches alert messages.

crit—Matches critical messages.

emerg—Matches emergency messages.

error—Matches error messages.

notice—Matches notification messages.

warn—Matches warning messages.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

debug-log (System Services)

Syntax debug-log *filename*;

Hierarchy Level [edit system services webapi]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the name of the log file to which trace messages for the integrated ClearPass authentication and enforcement Web API daemon (webapi) are written.

The debug level flag determines the kind of logs that are written to this file. Possible values are:

alert—Matches alert messages.

crit—Matches critical messages.

emerg—Matches emergency messages.

error—Matches error messages.

notice—Matches notification messages.

warn—Matches warning messages.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

default-certificate (System Services)

Syntax default-certificate;

Hierarchy Level [edit system services webapi https]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify that the default certificate is to be used for the integrated ClearPass authentication and enforcement Web API daemon (webapi) HTTPS configuration. To ensure security, the Junos OS default certificate key size is 2084 bits.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

default-profile

Syntax default-profile *profile-name*;

Hierarchy Level [edit access firewall-authentication pass-through]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the authentication profile to use if no profile is specified in a policy.

Options ***profile-name***—Name of the profile.

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

Related Documentation
- *Junos OS Security Configuration Guide*

delay-query-time (Services User Identification)

Syntax delay-query-time *delay-time-in-seconds*;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description If the CPPM does not send to the SRX Series device authentication and identity information for a particular user, the SRX Series device can request that information for the user if you configure the user query function.

Delays can occur from when the CPPM initially posts user authentication information to the SRX Series device to when the SRX Series device updates its ClearPass authentication table with that information. In its transit, the user identity information must first pass through the CPPM device's control plane and the control plane of the SRX Series device.

During that period, traffic might arrive at the SRX Series device that is generated by an access request from a user whose authentication and identity information is in transit from the CPPM to the SRX Series device. Rather than allow the SRX Series device to respond automatically by sending a user query request *immediately*, you can set the delay time parameter specifying in seconds how long the SRX Series device should wait before sending the request.

After the delay timeout expires, the SRX Series device sends the query to the CPPM and creates a pending entry for the user in the Routing Engine authentication table. During this period, any arriving traffic matches the default policy whose action on the traffic you can configure.

Options ***delay-time-in-seconds***—Amount of time for the SRX Series device to delay before sending queries to the Aruba ClearPass Policy Manager (CPPM) for authentication and identity information for individual users

Range: 0 through 60 seconds

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

distinguished-name (Access)

Syntax distinguished-name *distinguished-name*;

Hierarchy Level [edit access ldap-options search admin-search],
[edit access profile *profile-name* ldap-options search admin-search]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the distinguished name of an administrative user. The distinguished name is used in the bind for performing the LDAP search.

Options ***distinguished-name***—Set of properties that define the user. For example, **cn**=admin, **ou**=eng, **o**=juniper, **dc**=net.

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

Related Documentation
- *Junos OS Security Configuration Guide*

domain-name (Access Profile)

Syntax domain-name ***domain-name***;

Hierarchy Level [edit access profile ***profile-name*** client-name-filter ***client-name***]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify a domain name that must be in a client's name during the authentication process.

Options ***domain-name***—Domain name that must be in a client name. The name must not exceed 128 characters.

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

Related Documentation
- *Junos OS Security Configuration Guide*

enable-flow-tracing (Services)

Syntax enable-flow-tracing;

Hierarchy Level [edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Enable flow tracing for the profile. This statement is supported on the SRX550M, SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices.

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

Related Documentation

- SSL Proxy Overview on page 13
- Configuring SSL Proxy on page 23
- Enabling Debugging and Tracing for SSL Proxy on page 33

enable-session-cache

Syntax enable-session-cache;

Hierarchy Level [edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. This statement is supp

Description Enable SSL session cache. This statement is supported on the SRX550M, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices.

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

Related Documentation

- SSL Proxy Overview on page 13
- Configuring SSL Proxy on page 23
- Enabling Debugging and Tracing for SSL Proxy on page 33

end-user-profile

Syntax

```
end-user-profile profile-name profile-name
domain-name domain-name;
{
attribute device-category {
   string string-value;
}
attribute device-identity {
   string string-value;
}
attribute device-vendor {
   string string-value;
}
attribute device-type {
   string string-value;
}
attribute device-os {
   string string-value;
}
attribute device-os-version {
   string string-value;
   }
}
```

Hierarchy Level [edit services user-identification device-information]

Release Information Statement introduced in Junos OS 15.1X49-D70.

Description Specify the name of the device identity profile, also referred to as the **end-user-profile**, and either one or more of its attributes or the name of the Active Directory domain to which the device belongs.

The device identity profile is a key component of the SRX Series device identity feature, which enables you to control access to network resources based on the identity of the user's device, not the identity of the user of the device. The device identity profile includes the domain name and a collection of attributes that characterize the device.

NOTE: You cannot configure the device identity profile without specifying either the domain that the device belongs to at least one of its attributes.

Options

- profile-name *profile-name*—Name of the device identity profile; for example, marketing-west-coast. The profile is specified in the **source-end-user-profile** field of a security policy.
- domain *domain-name*—Name of the domain to which the device belongs; for example, domain1.
- attribute device-identity *string*––Name given to the device, for example, my-device1.

- attribute device-category *string*––Category of the device, for example, laptop.
- attribute device-vendor *string*—Name of the manufacturer of the device, for example, Lenovo.
- attribute device-type *string*––Type of device; for example, ThinkPad.
- attribute device-os *string*-—Operating system running on the device; for example, Windows.
- attribute device-os-version *string*-—Version of the operating system that is running on the device; for example, 10.1.

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

Related Documentation
- Understanding Access Control to Network Resources Based on Device Identity Information on page 185
- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192

fail

Syntax
fail *string*;

Hierarchy Level
[edit access firewall-authentication pass-through default-profile *profile-name* (ftp | http | telnet) banner]

Release Information
Statement introduced in Release 8.5 of Junos OS.

Description
Specify the banner that a client sees if the authentication process fails.

Options
string—Banner text. Maximum length of the message text is 250 characters. Enclose the banner text within spaces or special characters—for example, quotation marks (" ").

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

Related Documentation
- *Junos OS Security Configuration Guide*

file (Services)

Syntax

```
file file-name; {
  files;
  match;
  no-world-readable size;
  world-readable;
}
```

Hierarchy Level [edit services ssl traceoptions]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices, and vSRX..

Description Specify the trace file information. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

- **files**—Specify the maximum number of trace files. Range: 2 through 1000.
- **match**—Specify the regular expression for lines to be logged.
- **no-world-readable size**—Do not allow any user to read the log file.
- **size**—Specify the maximum trace file size. Range: 10,240 to 1,073,741,824.
- **world-readable**—Allow any user to read the log file.

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

Related Documentation

- Configuring SSL Proxy on page 23

files (Services)

Syntax files *files*;

Hierarchy Level [edit services ssl traceoptions file *file-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the maximum number of trace files.

Options **files**—Specify the maximum number of trace files.

Range: 2 through 1000

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

Related Documentation

- Configuring SSL Proxy on page 23

file (Services User Identification)

Syntax

```
file {
  filename
  files number;
  match regular-expression;
  size maximum-file-size;
  (world-readable | no-world-readable);
}
```

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass traceoptions]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the name of the trace log file and its characteristics to which messages for the behavior of the authentication source are logged. For the SRX Series device integrated ClearPass authentication and enforcement feature, the authentication source is the Aruba ClearPass Policy Manager (CPPM).

Options ***filename***—Name of the log file.

files *max-number-of-files*—Specifies the maximum number of trace files.
Range: 2 through 1000

match *regular-expression*—Specifies a regular expression that determines which lines are logged.

no-world-readable—Denies users the ability to read the log file.

size *max-file-size*—Specifies the trace file maximum file size.
Range: 10,240 through 1,073,741,824.

world-readable—Allows users to read the log file.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

file (System Logging)

Syntax

```
file filename {
    allow-duplicates;
    any (alert | any | critical | emergency | error | info | none | notice | warning);
    archive {
        archive-sites {
            url password;
        }
        (binary-data | no-binary-data);
        files number;
        size size;
        start-time start-time;
        transfer-interval transfer-interval;
        (world-readable | no-world-readable);
    }
    authorization (alert | any | critical | emergency | error | info | none | notice | warning);
    change-log (alert | any | critical | emergency | error | info | none | notice | warning);
    conflict-log (alert | any | critical | emergency | error | info | none | notice | warning);
    daemon (alert | any | critical | emergency | error | info | none | notice | warning);
    dfc (alert | any | critical | emergency | error | info | none | notice | warning);
    explicit-priority;
    external (alert | any | critical | emergency | error | info | none | notice | warning);
    firewall (alert | any | critical | emergency | error | info | none | notice | warning);
    ftp (alert | any | critical | emergency | error | info | none | notice | warning);
    interactive-commands (alert | any | critical | emergency | error | info | none | notice | warning);
    kernel (alert | any | critical | emergency | error | info | none | notice | warning);
    match "regular-expression";
    ntp (alert | any | critical | emergency | error | info | none | notice | warning);
    pfe (alert | any | critical | emergency | error | info | none | notice | warning);
    security (alert | any | critical | emergency | error | info | none | notice | warning);
    structured-data {
        brief;
    }
    user (alert | any | critical | emergency | error | info | none | notice | warning);
}
```

Hierarchy Level [edit system syslog]

Release Information Statement introduced before Junos OS Release 12.1X47 for SRX Series.

Description Specify the file in which to log data.

Options

- *filename*—Specify the name of the file in which to log data.
- *allow-duplicates*—Do not suppress the repeated messages.
- *any*—Specify all facilities information.
  - *alert*—Specify the conditions that should be corrected immediately.
  - *critical*—Specify the critical conditions.
  - *emergency*—Specify the conditions that cause security functions to stop.
  - *error*—Specify the general error conditions.

    - *info*—Specify the information about normal security operations.
    - *none*—Do not specify any messages.
    - *notice*—Specify the conditions that should be handled specifically.
    - *warning*—Specify the general warning conditions.
- *archive*—Specify the archive file information.
    - *archive-sites*—Specify a list of destination URLs for the archived log files.
        - *url*—Specify the primary and failover URLs to receive archive files.
    - *binary-data*—Mark file such that it contains binary data.
    - *no-binary-data*—Do not mark the file such that it contains binary data.
    - *files*—Specify the number of files to be archived. Range: 1 through 1000 files.
    - *size*—Specify the size of files to be archived. Range: 65,536 through 1,073,741,824 bytes.
    - *world-readable*—Allow any user to read the log file.
    - *no-world-readable*—Do not allow any user to read the log file.
    - *start-time*—Specify the start time for file transmission. Enter the start time in the yyyy-mm-dd.hh:mm format.
    - *transfer-interval*—Specify the frequency at which to transfer the files to archive sites.
- *authorization*—Specify the authorization system.
- *change-log*—Specify the configuration change log.
- *conflict-log*—Specify the configuration conflict log.
- *daemon*—Specify the various system processes.
- *dfc*—Specify the dynamic flow capture.
- *explicit-priority*—Include the priority and facility in messages.
- *external*—Specify the local external applications.
- *firewall*—Specify the firewall filtering system.
- *ftp*—Specify the FTP process.
- *interactive-commands*—Specify the commands executed by the UI.
- *kernel*—Specify the kernel information.
- *match*—Specify the regular expression for lines to be logged.
- *ntp*—Specify the NTP process.
- *pfe*—Specify the Packet Forwarding Engine.
- *security*—Specify the security-related information.

- *structured-data*—Log the messages in structured log format.
  - *brief*—Omit English language text from the end of the logged message.
- *user*—Specify the user processes.
  - *info*—Specify the informational messages.

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

filter (Security)

Syntax filter threat-attack

Hierarchy Level [edit security log stream *stream-name*]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the log stream filter to transmit only threat and attack logs to the ClearPass Policy Manager (CPPM). The integrated ClearPass authentication and enforcement feature sends to the CPPM threat and attack logs detected by the SRX Series device security modules. You can use these reports to inform your approach to hardening the CPPM security policy. Setting the log stream filter to threat-attack ensures that the SRX Series device and the log server are not overburdened by irrelevant logs.

NOTE: Unlike for other features that use a filter for log streams, threat-attack is the only log stream filter supported for integrated ClearPass. Therefore, it is not shown here as an option.

Required Privilege Level security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

firewall-authentication

Syntax

```
firewall-authentication {
  pass-through {
    default-profile profile-name;
    ftp {
      banner {
        fail string;
        login string;
        success string;
      }
    }
    http {
      banner {
        fail string;
        login string;
        success string;
      }
    telnet {
      banner {
        fail string;
        login string;
        success string;
      }
  }
  traceoptions {
    file {
      filename;
      files number;
      flag flag;
      match regular-expression;
      no-remote-trace;
      size maximum-file-size;
      (world-readable | no-world-readable);
    }
  }
  web-authentication {
    banner {
      success string;
    }
    default-profile profile-name;
  }
}
```

Hierarchy Level [edit access]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure default firewall authentication settings used by firewall authentication policies that restrict and permit access of firewall users to protected resources behind a firewall.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation
- *Dynamic VPN Overview*
- Firewall User Authentication Overview on page 9

firewall-authentication (Security)

Syntax

```
firewall-authentication {
  traceoptions {
    flag flag;
  }
}
```

Hierarchy Level
[edit security]

Release Information
Statement introduced in Junos OS Release 8.5.

Description
Define data-plane firewall authentication tracing options.

Options
- **flag**—Trace operation to perform. To specify more than one trace operation, include multiple flag statements.
  - **all**—Enable all tracing operations.
  - **authentication**—Trace data-plane firewall authentication events.
  - **proxy**—Trace data-plane firewall authentication proxy events.
- **detail**—Display moderate amount of data.
- **extensive**—Display extensive amount of data.
- **terse**—Display minimum amount of data.

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

Related Documentation
- Firewall User Authentication Overview on page 9
- *Understanding Logical System Firewall Authentication*

firewall-authentication (Security Policies)

Syntax

```
firewall-authentication {
  pass-through {
    access-profile profile-name;
    client-match user-or-group-name;
    ssl-termination-profile profile-name;
    web-redirect;
    web-redirect-to-https;
  }
  user-firewall {
    access-profile profile-name;
    domain domain-name
    ssl-termination-profile profile-name;
    web-redirect;
    web-redirect-to-https;
  }
  web-authentication {
    client-match user-or-group-name;
  }
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit]

Release Information Statement introduced in Junos OS Release 8.5.
Support added for the **user-firewall** option in Junos OS Release 12.1X45-D10.
Support for the **ssl-termination-profile** and **web-redirect-to-https** options added on high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.
Starting with Junos OS Release 15.1X49-D70, support for the **web-redirect** and **web-redirect-to-https** options under **user-firewall** added on SRX300, SRX320, SRX340, SRX345, SRX550M, SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX Services Gateways.

Description Configure firewall authentication methods.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Understanding User Role Firewalls*

firewall-authentication (User Identification)

Syntax firewall-authentication priority *priority*;

Hierarchy Level [edit security user-identification authentication-source]

Release Information Statement introduced in Junos OS Release 12.1X45-D10. Support for **disable** option dropped in Junos OS Release 12.1X47-D10.

Description Enables the firewall authentication table as an authentication source. The priority of this table among other authentication tables establishes the search sequence used to identify user and role values.

Options ***priority***—A unique value between 0 and 65535 that determines the sequence for searching multiple tables to retrieve a user role. Each table is given a unique priority value. The lower the value, the higher the priority. A table with priority 120 is searched before a table with priority 200.

Default: 150

Setting the priority value of the firewall authentication table to 0 is equivalent to disabling the table and eliminating it from the search sequence.

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

Related Documentation
- *authentication-source (Security)*
- *Understanding User Role Firewalls*

firewall-authentication-service

Syntax firewall-authentication-service (enable | disable);

Hierarchy Level [edit system processes]

Release Information Statement introduced in Junos OS Release 8.5.

Description Enable or disable the firewall authentication service process.

Options
- **enable**—Start the firewall authentication service process.
- **disable**—Stop the firewall authentication service process.

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

Related Documentation
- Firewall User Authentication Overview on page 9

firewall-user

Syntax

```
firewall-user {
    password password;
}
```

Hierarchy Level [edit access profile *profile-name* client *client-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify a client as a firewall user and the associated password (encrypted).

Options **password** ***password***—Password used by the firewall user during local authentication.

Range: 1 through 128 characters

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

Related Documentation

- *Junos OS Security Configuration Guide*

flag (Services)

Syntax flag (*all* | *cli-configuration* | *initiation* | *proxy* | *selected-profile* | *termination*);

Hierarchy Level [edit services ssl traceoptions]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Description Specify the tracing flag parameters.

Options

- *all*—Trace all the parameters.
- *cli-configuration*—Trace CLI configuration events.
- *initiation*—Trace initiation service events.
- *proxy*—Trace proxy service events.
- *selected-profile*—Trace events for profiles with **enable-flow-tracing** set.
- *termination*—Trace termination service events.

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

Related Documentation

- Configuring SSL Proxy on page 23

from-zone (Security Policies)

Syntax

```
from-zone zone-name to-zone zone-name {
  policy policy-name {
    description description;
    match {
      application {
        [application];
        any;
      }
      destination-address {
        [address];
        any;
        any-ipv4;
        any-ipv6;
      }
      source-address {
        [address];
        any;
        any-ipv4;
        any-ipv6;
      }
      source-identity {
        [role-name];
        any;
        authenticated-user;
        unauthenticated-user;
        unknown-user;
      }
      source-end-user-profile {
        profile-name;
      }
    }
    scheduler-name scheduler-name;
    then {
      count {
        alarm {
          per-minute-threshold number;
          per-second-threshold number;
        }
      }
      deny;
      log {
        session-close;
        session-init;
      }
      permit {
        application-services {
          application-firewall {
            rule-set rule-set-name;
          }
          application-traffic-control {
            rule-set rule-set-name;
          }
```

```
                gprs-gtp-profile profile-name;
                gprs-sctp-profile profile-name;
                idp;
                redirect-wx | reverse-redirect-wx;
                ssl-proxy {
                  profile-name profile-name;
                }
                uac-policy {
                  captive-portal captive-portal;
                }
                utm-policy policy-name;
              }
              destination-address {
                drop-translated;
                drop-untranslated;
              }
              firewall-authentication {
                pass-through {
                  access-profile profile-name;
                  client-match user-or-group-name;
                  ssl-termination-profile profile-name;
                  web-redirect;
                  web-redirect-to-https;
                }
                user-firewall {
                  access-profile profile-name;
                  domain domain-name
                  ssl-termination-profile profile-name;
                }
                web-authentication {
                  client-match user-or-group-name;
                }
              }
              services-offload;
              tcp-options {
                initial-tcp-mss mss-value;
                reverse-tcp-mss mss-value;
                sequence-check-required;
                sequence-check-required;
                syn-check-required;
              }
              tunnel {
                ipsec-group-vpn group-vpn;
                ipsec-vpn vpn-name;
                pair-policy pair-policy;
              }
            }
            reject;
          }
        }
      }
```

Hierarchy Level [edit security policies]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **services-offload** option added in Junos OS Release 11.4. Support for the **source-identity** option added in Junos OS Release 12.1. Support for the **description** option added in Junos OS Release 12.1. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20.

Description Specify a source zone and destination zone to be associated with the security policy.

Options

- **from-zone** *zone-name*—Name of the source zone.
- **to-zone** *zone-name*—Name of the destination zone.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

ftp (Access)

Syntax

```
ftp {
  banner {
    fail string;
    login string;
    success string;
  }
}
```

Hierarchy Level [edit access firewall-authentication pass-through]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure banners for the FTP login prompt, successful authentication, and failed authentication.

Options The remaining statements are explained separately.

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

group-profile (Access)

Syntax

```
group-profile profile-name {
  ppp {
    cell-overhead;
    encapsulated-overhead;
    framed-pool address-pool-name;
    idle-timeout seconds;
    interface-id interface-identifier;
    keepalive seconds;
    primary-dns IP address;
    primary-wins IP address;
    secondary-dns IP address;
    secondary-dns IP address;
  }
}
```

Hierarchy Level [edit access]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Configure a group profile to define Point-to-Point Protocol (PPP) attributes. Any client referencing the configured group profile inherits all the group profile attributes.

Options

- **ppp**—Configure Point-to-Point Protocol (PPP) attributes.
- **cell-overhead**—Configure the session to use Asynchronous Transfer Mode (ATM)-aware egress shaping.
- **framed-pool *pool-name***—Configure a framed-pool.
- **idle-timeout**—Configure the idle timeout for a user.
- **interface-id**—Configure the interface identifier.
- **keep-alive**—Configure the keepalive interval for an L2TP tunnel.
- **primary-dns**—Specify the primary-dns IP address.
- **secondary-dns**—Specify the secondary-dns IP address.
- **primary-wins**—Specify the primary-wins IP address.
- **secondary-wins**—Specify the secondary-wins IP address.

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

Related Documentation

- *Junos OS Security Configuration Guide*

http (Access)

Syntax

```
http {
  banner {
    fail string;
    login string;
    success string;
  }
}
```

Hierarchy Level [edit access firewall-authentication pass-through]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure banners for the HTTP login prompt, successful authentication, and failed authentication.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11

http (Services)

Syntax

```
http {
  interfaces [ interface-names ];
  port port;
}
```

Hierarchy Level [edit system services web-management]

Release Information Statement introduced in Junos OS Release 8.5.

Description Unencrypted HTTP connection setting.

Options

interfaces [*interface-names*]—Name of one or more interfaces on which to allow the HTTP service.

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

- Example: Configuring HTTPS Traffic to Trigger Pass-Through Authentication on page 43
- Example: Configuring HTTPS Traffic to Trigger Web Authentication on page 60
- Firewall User Authentication Overview on page 9

http (Services User Identification)

Syntax http port *port-number*;

Hierarchy Level [edit services user-identification authentication-source *name* user-query web-server *name* connect-method]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure HTTP as the connection protocol to use for the SRX Series integrated ClearPass authentication and enforcement feature's connection to the ClearPass Policy Manager (CPPM) webserver for individual user authentication queries. You identify the connection protocol as part of the configuration that identifies the CPPM webserver (mutually exclusive with HTTPS).

If the SRX Series devices does not find an authentication entry for a user in its local ClearPass authentication table, it can query the Aruba ClearPass webserver for this information.

NOTE: This configuration assumes that aruba-clearpass is specified as the authentication source.

Options ***port-number***—Port numbr to use for the HTTP connection protocol.

Required Privilege Level

services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

http (System Services)

Syntax http port *port-number*;

Hierarchy Level [edit system services webapi]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify HTTP as the communication protocol for the Web API function of the SRX Series integrated ClearPass authentication and enforcement feature.

The SRX Series device exposes to the ClearPass Policy Manager (CPPM) the Web API for it to use to initiate a connection and then use that connection to send to the SRX Series device user authentication and identity information.

This statement also specifies the port number to use for the HTTP connection. The port number is optional.

NOTE: If you deploy HTTP along with a Web management application, you must ensure that they run on different service ports.

Options ***port-number***—Port for HTTP to use for the Web API function.

Default: 8080

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

https (Services)

Syntax

```
https {
  interfaces [ interface-names ];
  local-certificate local-certificate-name;
  pki-local-certificate pki-local-certificate-name;
  port port;
  system-generated-local-certificate name;
}
```

Hierarchy Level [edit system services web-management]

Release Information Statement introduced on high-end SRX Series Services Gateways, from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.

Description Encrypted HTTPS connections.

Options ***interface-names***—Name of one or more interfaces on which to allow the HTTPS service.

local-certificate-name —Name of the X.509 certificate for a Secure Sockets Layer (SSL) connection. An SSL connection is configured at the **[edit security certificates local]** hierarchy.

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

- Example: Configuring HTTPS Traffic to Trigger Pass-Through Authentication on page 43
- Example: Configuring HTTPS Traffic to Trigger Web Authentication on page 60
- Firewall User Authentication Overview on page 9

https (Services User Identification)

Syntax

```
https (
  certificate local-certificate;
  certificate-key local-certificate-key;
  default-certificate;
  pki-local-certificate certificate-name;
  port port-number;
)
```

Hierarchy Level

[edit services user-identification authentication-source *name* user-query web-server *name* connect-method]

Release Information

Statement introduced in Junos OS Release 12.3X48-D30.

Description

Configure HTTPS as the connection protocol used for the SRX Series connection to the ClearPass Policy Manager (CPPM) for user query requests. You identify the connection protocol as part of the configuration that identifies the CPPM webserver.

The integrated ClearPass authentication and enforcement user query function allows the SRX Series device to request from the CPPM user authentication and identity information for an individual when the SRX Series ClearPass authentication table does not contain that information.

NOTE: This configuration assumes that aruba-clearpass is specified as the authentication source.

Options

https—Use the encrypted HTTPS protocol. (Mutually exclusive with HTTP.)

Default: HTTPS

default-certificate—Use the default HTTPS certificate.

For security reasons, the HTTPS default-certificate key size 2048.

filename—Custom certificate file.

The Web API supports only the Privacy-Enhanced Mail (PEM) format for the custom certificate and certificate key configuration.

local-certificate-key—Web API daemon service certificate key. This parameter is required if a custom service certificate file is configured.

pki-certificate—Use the local X.509 PKI certificate.

port-number—HTTPS service port.

Range: 1 through 65535.

Default: 8443

Required Privilege Level

services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

https (System Services)

Syntax

```
https (
  certificate local-certificate;
  certificate-key local-certificate-key;
  default-certificate;
  pki-local-certificate certificate-name;
  port port-number;
)
```

Hierarchy Level [edit system services webapi connect-method]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify HTTPS as the communication protocol for the Web API function of the SRX Series integrated ClearPass authentication and enforcement feature. When you configure HTTPS, you specify the service certificate and certificate key. You can also specify the port to be used.

The Web API daemon, acting as an HTTPS server, allows the ClearPass Policy Manager (CPPM), acting as the client, to send POST request messages to it. The CPPM, which is the authentication source for this feature, sends to the SRX Series device user authentication and identity information.

NOTE: If you deploy HTTPS with a Web management application, ensure that they run on different service ports.

Options **https**—Specifies use of the encrypted HTTPS protocol. (Mutually exclusive with HTTP.)

default-certificate—Configures the Web API daemon (webapi) to use the default HTTPS certificate.

Default: key size, 2048

filename—Configures the Web API daemon to use the specified, custom certificate file.

For certificate and certificate key configuration, the Web API function supports only the Privacy-Enhanced Mail (PEM) format.

local-certificate-key—Configures the Web API daemon service certificate key. This parameter is required if a custom service certificate file is configured.

certificate-name—Configures the Web API daemon to use the local X.509 PKI certificate.

port-number—Configures the HTTPS service port.

Range: For port number, 1 through 65,535.

Default: For port, 8443.

Required Privilege Level

system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

infranet-controller

Syntax

```
infranet-contoller host-name {
  address ip-address;
  ca-profile [ca-profile];
  interface interface-name;
  password password;
  port port-number;
  server-certificate-subject subject;
}
```

Hierarchy Level

[edit services unified-access-control]

Release Information

Statement introduced in Junos OS Release 9.4.

Description

To configure an Infranet Controller, specify the hostname of the IC Series device with which the SRX Series device should communicate. Possible values for this statement range from 1 to 31 characters.

This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

One or more IC Series devices can be configured as Infranet Controllers on the SRX Series device. There is no maximum number of IC Series devices that can be configured. However, only one IC Series device can be active at any time. The others are failover devices. A round-robin algorithm determines which of the configured IC Series devices is the active Infranet Controller. If the active Infranet Controller becomes inoperative, the algorithm is reapplied to the remaining IC Series devices that are configured to establish the new active Infranet Controller.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- Firewall User Authentication Overview on page 9

interface (Services)

Syntax interface *interface-name*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the SRX Series interface through which the IC Series device should connect.

This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation
- port (Services) on page 373
- password (Services) on page 362

interval (Services)

Syntax interval *seconds*;

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the value in seconds that the SRX Series device should expect to receive a heartbeat signal from the IC Series device (default 30). This configuration statement is used in conjunction with the **timeout** statement to test active communications with the IC Series device. The value of the **interval** statement must be smaller than the value of **timeout** statement.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation
- timeout (Services) on page 406
- timeout-action on page 407

ip-address (Access Profile)

Syntax ip-address *address*

Hierarchy Level [edit access profile *name* client *name* xauth]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Specify the IP address for the client.

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

Related Documentation

- *Junos OS Security Configuration Guide*

ip-user-mapping

Syntax

```
ip-user-mapping {
  discovery-method {
    wmi {
      event-log-scanning-interval seconds;
      initial-event-log-timespan hours;
    }
  }
}
```

Hierarchy Level [edit services user-identification active-directory domain]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description Control how the SRX Series device accesses a domain controller in order to monitor and scan security event logs on the domain controller. By parsing the event log, the SRX Series gets IP address-to-user mappings. This process is part of the integrated user firewall feature. The **ip-user-mapping** statement is optional because WMI is the default discovery method and its properties have default values.

The other available method the SRX Series uses to retrieve address-to-user mapping information is manual (on-demand) probing of a domain PC.

Options

discovery-method—Method of discover IP address-to-user mappings.

wmi—Windows Management Instrumentation (WMI) is the discovery method used to access the domain controller.

event-log-scanning-interval *seconds*—Optional. Interval at which the SRX Series scans the event log on the domain controller.

Range: 5 through 60 seconds

Default: 10 seconds

initial-event-log-timespan *hours*—Optional. Time of the earliest event log on the domain controller that the SRX Series will initially scan. This argument applies to the initial deployment only. After WMIC and the user identification start working, the SRX Series scans only the latest event log.

Range: 1 through 168 hours

Default: 1 hour

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

Related Documentation

- active-directory-access on page 294
- clear services user-identification active-directory-access on page 456
- request services user-identification active-directory-access ip-user-probe on page 460
- user-identification (Services) on page 430
- show services user-identification active-directory-access statistics on page 529

- traceoptions (Active Directory Access) on page 417

ldap-options

Syntax

```
ldap-options {
  assemble {
    common-name common-name;
  }
  base-distinguished-name base-distinguished-name;
  revert-interval seconds;
  search {
    admin-search {
      distinguished-name distinguished-name;
      password password;
    }
    search-filter filter-name;
  }
}
```

Hierarchy Level [edit access],
[edit access profile *profile-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure LDAP authentication options.

Options The remaining options are explained separately.

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

Related Documentation

- *Junos OS Security Configuration Guide*

ldap-server

Syntax

```
ldap-server server-address {
  port port-number;
  retry attempts;
  routing-instance routing-instance-name;
  source-address source-address;
  timeout seconds;
  no-tls-certificate-check;
  tls-min-version (v1.1 | v1.2);
  tls-peer-name;
  tls-timeout;
  tls-type {
    start-tls;
  }
}
```

Hierarchy Level

[edit access]
[edit access profile *profile-name*]

Release Information

Statement introduced in Release 8.5 of Junos OS.

Description

Specify that the device uses a Lightweight Directory Access Protocol (LDAP) server for authentication.

Options

server-address—Address of the LDAP authentication server.

The remaining statements are explained separately.

Required Privilege Level

access—To view this statement in the configuration.
access-control—To add this statement to the configuration.

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11
- LDAP Functionality in Integrated User Firewall on page 150

level (Services)

Syntax level [*brief* | *detail* | *extensive* | *verbose]*];

Hierarchy Level [edit services ssl traceoptions]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the level of debugging the output. This statement is supported on the SRX550M, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

- *brief*—Specify brief debugging output.
- *detail*—Specify detailed debugging output.
- *extensive*—Specify extensive debugging output.
- *verbose*—Specify verbose debugging output.

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

Related Documentation

- Configuring SSL Proxy on page 23

level (Services User Identification)

Syntax level (brief | detail | extensive | verbose);

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass traceoptions]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the level of messages o be written to the trace log file about authentication source behavior.

For the integrated ClearPass authentication enforcement feature, the authentication source is Aruba ClearPass.

Options

all—Matches all levels.

error—Matches error conditions.

info—Matches informational messages.

notice—Matches conditions that require special handling.

verbose—Matches verbose messages.

warning—Matches warning messages.

Required Privilege Level

services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

lifetime-seconds (Security IKE)

Syntax lifetime-seconds *seconds*;

Hierarchy Level [edit security ike proposal *proposal-name*]

Release Information Statement introduced in Junos OS Release 8.5. Default value modified in Junos OS Release 10.2.

Description Specify the lifetime (in seconds) of an IKE security association (SA). When the SA expires, it is replaced by a new SA and security parameter index (SPI) or terminated.

Options ***seconds***—Lifetime of the IKE SA.
Range: 180 through 86,400 seconds
Default: 28,800 seconds

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

Related Documentation
- *IPsec VPN Overview*
- *Understanding User Authentication Methods*

link (Access)

Syntax link *pool-name*;

Hierarchy Level [edit access address-assignment pool]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Configure the name of the secondary address-assignment pool that is linked to a primary address-assignment pool. The secondary pool provides a backup pool for local address assignment.

Options *pool-name*—Name of the address assignment pool.

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

Related Documentation
- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11

local-authentication-table

Syntax local-authentication-table priority *priority*;

Hierarchy Level [edit security user-identification authentication-source]

Release Information Statement introduced in Junos OS Release 12.1. Support for **disable** option dropped in Junos OS Release 12.1X47-D10.

Description An authentication table created on the SRX Series device using the **request security user-identification local-authentication-table add** command.

Options priority ***priority***—A unique value between 0 and 65535 that determines the sequence for searching multiple tables to retrieve a user role. Each table is given a unique priority value. The lower the value, the higher the priority. A table with priority 120 is searched before a table with priority 200. The default priority value of the local authentication table is 100.

Setting the priority value of the local authentication table to 0 is equivalent to disabling the table and eliminating it from the search sequence.

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

Related Documentation

- *Understanding User Role Firewalls*
- *Understanding the User Identification Table*

log (Services)

Syntax

```
log {
    all;
    errors;
    info;
    sessions-allowed;
    sessions-dropped;
    sessions-ignored;
    sessions-whitelisted;
    warning;
}
```

Hierarchy Level [edit services ssl proxy profile *profile-name* actions]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the logging actions.

Options

- **all**—Log all events.
- **errors**—Log all error events.
- **info**—Log all information events.
- **sessions-allowed**—Log SSL session allowed events after an error.
- **sessions-dropped**—Log only SSL session dropped events.
- **sessions-ignored**—Log session ignored events.
- **sessions-whitelisted**—Log SSL session whitelisted events.
- **warning**—Log all warning events.

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

Related Documentation

- Configuring SSL Proxy on page 23

login (Access)

Syntax login *string*;

Hierarchy Level [edit access firewall-authentication pass-through default-profile *profile-name* (ftp | http | telnet) banner]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the login banner for users using FTP, HTTP, and Telnet during the authentication process.

Options ***string***—Banner text. Maximum length of the message text is 250 characters. Enclose the banner text within spaces or special characters—for example quotation marks (" ").

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

Related Documentation
- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11

match (Services)

Syntax match *match*;

Hierarchy Level [edit services ssl traceoptions file *file-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the regular expression for lines to be logged. This statement is supported on the SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options **match**—Specify the regular expression for lines to be logged.

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

Related Documentation
- Configuring SSL Proxy on page 23

network (Access)

Syntax network

Hierarchy Level [edit access address-assignment pool <name> family (inet | inet6)]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Specify the IPv4 network address for the pool. This attribute is mandatory. For an IPv6 pool, you will set the IPv6 network prefix.

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

Related Documentation
- *Junos OS Security Configuration Guide*

no-remote-trace (Services)

Syntax no-remote-trace;

Hierarchy Level [edit services ssl traceoptions]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Disable remote tracing.

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

Related Documentation
- Configuring SSL Proxy on page 23

no-remote-trace (Services User Identification)

Syntax no-remote-trace;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass traceoptions]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Disable remote tracing.

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

no-user-query (Services User Identification)

Syntax no-user-query;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Disable the integrated ClearPass authentication and enforcement user query function, if it is configured. You can use the no-user-query statement to turn off the user query function without having to delete the configuration.

The user query function allows the SRX Series device to query the ClearPass webserver for authentication and identity information for an individual user whose information was not posted to the SRX Series device by ClearPass.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

no-tls-certificate-check

Syntax no-tls-certificate-check;

Hierarchy Level [edit access profile *profile-name* ldap-server *ip-address*]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Specify validation of the server certificate not required. SRX Series devices support an additional check on the Lightweight Directory Access Protocol (LDAP) server's certificate during the Transport Layer Security (TLS) handshake for LDAP authentication. If the validation of the server certificate is not required, you can use this option to ignore the validation and accept the certificate without checking. By default, this option is disabled.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Example: Configuring Integrated User Firewall on page 153

pass-through

Syntax

```
pass-through {
  default-profile profile-name;
  ftp {
    banner {
      fail string;
      login string;
      success string;
    }
  }
  http {
    banner {
      fail string;
      login string;
      success string;
    }
  telnet {
    banner {
      fail string;
      login string;
      success string;
    }
}
```

Hierarchy Level [edit access firewall-authentication]

Release Information Statement introduced in Junos OS Release 8.5.
HTTPS for pass-through authentication is supported on high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.

Description Configure pass-through , when a host or user from one zone needs to access a protected resource in another zone. A user must use an FTP, Telnet, or HTTP client to access the IP address of the protected resource and get authenticated by the firewall. The device uses FTP, Telnet, and HTTP to collect username and password information. Subsequent traffic from the user or host is allowed or denied based on the result of this authentication. After the user is authenticated, the firewall proxies the connection.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11

password (Access)

Syntax password *password*;

Hierarchy Level [edit access ldap-options search admin-search],
[edit access profile *profile-name* ldap-options search admin-search]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure the plain-text password for the administrative user. This password is used in the bind for performing the LDAP search.

Options ***password***—Administrative user password.

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

Related Documentation
- *Junos OS Security Configuration Guide*

password (Services)

Syntax password *password*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the password that the SRX Series device should send to the IC Series device to establish communications. The SRX Series device sends the password in its first message to the IC Series device.

This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation
- ca-profile (Services) on page 306
- server-certificate-subject on page 392

password (System Services)

Syntax password *password*;

Hierarchy Level [edit system services webapi user]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the password for the integrated ClearPass authentication and enforcement feature Web API daemon (webapi) user.

Range: 1 through 128 characters.

The Web API daemon, acting as an HTTP server, exposes to the Aruba ClearPass Policy Manager (CPPM) an API that allows the CPPM, acting as a client, to send POST request messages to it. The CPPM, which serves as the authentication source, initiates the session to the SRX Series device and sends it user authentication and identity information.

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

permit (Security Policies)

Syntax

```
permit {
  application-services {
    application-firewall {
      rule-set rule-set-name;
    }
    application-traffic-control {
      rule-set rule-set-name;
    }
    gprs-gtp-profile profile-name;
    gprs-sctp-profile profile-name;
    idp;
    redirect-wx | reverse-redirect-wx;
    ssl-proxy {
      profile-name profile-name;
    }
    uac-policy {
      captive-portal captive-portal;
    }
    utm-policy policy-name;
  }
  destination-address {
    drop-translated;
    drop-untranslated;
  }
  firewall-authentication {
    pass-through {
      access-profile profile-name;
      client-match user-or-group-name;
      ssl-termination-profile profile-name;
      web-redirect;
      web-redirect-to-https;
    }
    user-firewall {
      access-profile profile-name;
      domain domain-name
      ssl-termination-profile profile-name;
    }
    web-authentication {
      client-match user-or-group-name;
    }
  }
  services-offload;
  tcp-options {
    sequence-check-required;
    syn-check-required;
  }
  tunnel {
    ipsec-group-vpn group-vpn;
    ipsec-vpn vpn-name;
    pair-policy pair-policy;
  }
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **tcp-options** added in Junos OS Release 10.4. Support for the **services-offload** option added in Junos OS Release 11.4. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10.

Description Specify the policy action to perform when packets match the defined criteria.

Options The remaining statements are explained separately. See CLI Explorer.

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

pki-local-certificate (Services)

Syntax pki-local-certicate *pki-certificate*;

Hierarchy Level [edit services webapi https]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the Web API daemon to use the local X.509 PKI certificate for HTTPS when HTTPS is specified as the communication protocol. The SRX Series integrated ClearPass authentication and enforcement feature exposes the Web API to the ClearPass Policy Mananger (CPPM) to allow the CPPM to initiate a connection to the SRX Series device. For this feature, ClearPass acts as the authentication source. The CPPM uses the HTTPS connection to send user authentication and identity information to the SRX Series device.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

policies

Syntax

```
policies {
  default-policy (deny-all | permit-all);
  from-zone zone-name to-zone zone-name {
    policy policy-name {
      description description;
      match {
        application {
          [application];
          any;
        }
        destination-address {
          [address];
          any;
          any-ipv4;
          any-ipv6;
        }
        source-address {
          [address];
          any;
          any-ipv4;
          any-ipv6;
        }
        source-identity {
          [role-name];
          any;
          authenticated-user;
          unauthenticated-user;
          unknown-user;
        }
      }
      scheduler-name scheduler-name;
      then {
        count {
          alarm {
            per-minute-threshold number;
            per-second-threshold number;
          }
        }
        deny;
        log {
          session-close;
          session-init;
        }
        permit {
          application-services {
            application-firewall {
              rule-set rule-set-name;
            }
            application-traffic-control {
              rule-set rule-set-name;
            }
            gprs-gtp-profile profile-name;
```

```
            gprs-sctp-profile profile-name;
            idp;
            redirect-wx | reverse-redirect-wx;
            ssl-proxy {
              profile-name profile-name;
            }
            uac-policy {
              captive-portal captive-portal;
            }
            utm-policy policy-name;
          }
          destination-address {
            drop-translated;
            drop-untranslated;
          }
          firewall-authentication {
            pass-through {
              access-profile profile-name;
              client-match user-or-group-name;
              ssl-termination-profile profile-name;
              web-redirect;
              web-redirect-to-https;
            }
            user-firewall {
              access-profile profile-name;
              domain domain-name
              ssl-termination-profile profile-name;
            }
            web-authentication {
              client-match user-or-group-name;
            }
          }
          services-offload;
          tcp-options {
            sequence-check-required;
            syn-check-required;
          }
          tunnel {
            ipsec-group-vpn group-vpn;
            ipsec-vpn vpn-name;
            pair-policy pair-policy;
          }
        }
        reject;
      }
    }
}
global {
  policy policy-name {
    description description;
    match {
      application {
        [application];
        any;
      }
      destination-address {
```

```
        [address];
        any;
        any-ipv4;
        any-ipv6;
    }
    from-zone {
        [zone-name];
        any;
    }
    source-address {
        [address];
        any;
        any-ipv4;
        any-ipv6;
    }
    source-identity {
        [role-name];
        any;
        authenticated-user;
        unauthenticated-user;
        unknown-user;
    }
    to-zone {
        [zone-name];
        any;
    }
}
scheduler-name scheduler-name;
then {
    count {
        alarm {
            per-minute-threshold number;
            per-second-threshold number;
        }
    }
    deny;
    log {
        session-close;
        session-init;
    }
    permit {
        application-services {
            application-firewall {
                rule-set rule-set-name;
            }
            application-traffic-control {
                rule-set rule-set-name;
            }
            gprs-gtp-profile profile-name;
            gprs-sctp-profile profile-name;
            idp;
            redirect-wx | reverse-redirect-wx;
            ssl-proxy {
                profile-name profile-name;
            }
            uac-policy {
```

```
                captive-portal captive-portal;
              }
              utm-policy policy-name;
            }
            destination-address {
              drop-translated;
              drop-untranslated;
            }
            firewall-authentication {
              pass-through {
                access-profile profile-name;
                client-match user-or-group-name;
                ssl-termination-profile profile-name;
                web-redirect;
                web-redirect-to-https;
              }
              web-authentication {
                client-match user-or-group-name;
              }
            }
            services-offload;
            tcp-options {
              initial-tcp-mss mss-value;
              reverse-tcp-mss mss-value;
              sequence-check-required;
              syn-check-required;
            }
          }
          reject;
        }
      }
    }
    policy-rematch;
    policy-stats {
      system-wide (disable | enable) ;
    }
    traceoptions {
      file {
        filename;
        files number;
        match regular-expression;
        size maximum-file-size;
        (world-readable | no-world-readable);
      }
      flag flag;
      no-remote-trace;
    }
  }
```

Hierarchy Level [edit security]

Release Information

Statement introduced in Junos OS Release 8.5.
Support for the **services-offload** option added in Junos OS Release 11.4.
Support for the **source-identitiy** option added in Junos OS Release 12.1.
Support for the **description** option added in Junos OS Release 12.1.
Support for the **ssl-termination-profile** and **web-redirect-to-https** options added on SRX5400, SRX5600, and SRX5800 devices starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.
Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10.
Support for the **domain** option, and for the **from-zone** and **to-zone** global policy match options, added in Junos OS Release 12.1X47-D10.
Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20. Support for the **extensive** option for **policy-rematch** added in Junos OS Release 15.1X49-D20.

Description

Configure network security policies.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- *Security Policies Overview*

pool (Access)

Syntax

```
pool pool-name {
  family {
    inet {
      dhcp-attributes {
        boot-file boot file name;
        boot-server boot server name;
        domain-name domain name;
        grace-period seconds;
        maximum-lease-time (seconds | infinite);
        name-server ipv4-address;
        name-server address;
        netbios-node-type (b-node | h-node | m-node | p-node);
        option dhcp option-identifier-code;
        option-match {
          option-82 {
            circuit-id match-value;
            remote-id match-value;
          }
        }
        router IPv4 address;
        server-identifier IP address;
        tftp-server server name;
        wins-server IPv4 address;
      }
      host hostname;
      network network address;
      range range-name {
        high upper-limit;
        low lower-limit;
      }
      xauth-attributes {
        primary-dns IP address;
        primary-wins IP address;
        secondary-dns IP address;
        secondary-wins IP address;
      }
    }
    inet6 {
      dhcp-attributes {
        dns-server IPv6-address;
        grace-period seconds;
        maximum-lease-time seconds;
        option dhcp-option-identifier-code;
        sip-server-address IPv6-address;
        sip-server-domain-name domain-name;
      }
      prefix IPv6-network-prefix;
      range range-name {
        high upper-limit;
        low lower-limit;
        prefix-length delegated-prefix-length;
      }
```

```
    link pool-name;
}
```

Hierarchy Level [edit access address-assignment]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Configure the name of an address assignment pool. The remaining statements are explained separately.

Options ***pool-name***—Name assigned to the address-assignment pool.

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

Related Documentation
- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11

port (Access LDAP)

Syntax port *port-number*;

Hierarchy Level [edit access ldap-server *server-address*],
[edit access profile *profile-name* ldap-server *server-address*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure the port number on which to contact the LDAP server.

Options ***port-number***—Port number on which to contact the LDAP server.
Default: 389

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

Related Documentation
- *Junos OS Security Configuration Guide*

port (Services)

Syntax port *port-number*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the port on the IC Series device through which the SRX Series device should establish connections (default 11123). Possible values for this statement range from 1 through 65,535.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation

- interface (Services) on page 348
- password (Services) on page 362

port (System Services)

Syntax port *port-number*;

Hierarchy Level [edit system services webapi http]
[edit system services webapi https]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the SRX Series device TCP port to use for incoming HTTP or HTTPS connection requests initiated by the ClearPass Policy Manager (CPPM). The SRX Series device integrated ClearPass authentication and enforcement feature exposes its Web API (webapi) to the CPPM. The CPPM uses the Web API to establish a connection to the SRX Series device and send user authentication and identity information to it.

Options ***port-number***—For HTTP connection protocol.

Range: 1 through 65535.

Default: 8080

port ***port-number***—For HTTPS connection protocol.

Range: 1 through 65535.

Default: 8443

Required Privilege Level system—To view this statement in the configuration.

system-control—To add this statement to the configuration.

preferred-ciphers

Syntax preferred-ciphers (custom | medium | strong | weak);

Hierarchy Level [edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Select the preferred ciphers.

Options
- **custom**—Configure custom cipher suite and order of preference.
- **medium**—Use ciphers with key strength of 128 bits or greater.
- **strong**—Use ciphers with key strength of 168 bits or greater.
- **weak**—Use ciphers with key strength of 40 bits or greater.

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

Related Documentation
- Firewall User Authentication Overview on page 9
- SSL Proxy Overview on page 13

prefix (Access IPv6)

Syntax prefix *IPv6-network prefix*;

Hierarchy Level [edit access address-assignment pool pool-name family inet6]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Specify the IPv6 prefix for the IPv6 address-assignment pool. This statement is mandatory for IPv6 address-assignment pools.

Options *IPv6-network-prefix*—IPv6 prefix.

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

Related Documentation
- Firewall User Authentication Overview on page 9

priority (Security User Identification)

Syntax

```
authentication-source {
    active-directory priority priority;
    aruba-clearpass priority priority;
    firewall-authentication priority priority;
    local-authentication-table priority priority;
    unified-access-control priority priority;
}
```

Hierarchy Level [edit security user-identification]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Set the lookup priority to identify the order in which the SRX Series device checks its configured authentication tables for user authentication information. Authentication tables are searched in order based on their priority setting in which lowest value takes precedence.

For the integrated ClearPass authentication and enforcement feature, the SRX Series device must be configured to search the ClearPass authentication table first.

NOTE: Note that both the authentication source, Aruba ClearPass, and the SRX Series ClearPass authentication table are both referred to as aruba-clearpass in the CLI and its output.

You need to set this value only if the local authentication table, whose default value is 100, also resides on the Packet Forwarding Engine. In that case, you must configure a higher priority value, such as 120, for the local authentication table.

Options ***priority***—Aruba-clearpass authentication table search priority.

Range: 1 through 65535.

Default: 110.

Default values for other authentication tables:

- Local authentication table: 100
- Active Directory (AD) table: 125
- UAC authentication table: 150
- Firewall authentication table: 200

Required Privilege Level security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

protocol-version

Syntax protocol-version (all | tls1 | tls11 | tls12);

Hierarchy Level [edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. The **tls11** and **tls12** options are introduced in 15.1X49-D30.

Description Specify the accepted SSL protocol version.

Options

- **all**—Accept all versions of TLS.
- **TLS version 1.0**—Accept TLS version 1.0. It provides secure communication over networks by providing privacy and data integrity between communicating applications
- **TLS version 1.1**—Accept TLS version 1.1. This enhanced version of TLS provides protection against cipher-block chaining (CBC) attacks.
- **TLS version 1.2** —Accept TLS version 1.2. This enhanced version of TLS provides improved flexibility for negotiation of cryptographic algorithms.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- SSL Proxy Overview on page 13

query-api (Services User Identification)

Syntax query-api *query-api*

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure **query-api** to specify the path of the URL that the SRX Series device uses to query the ClearPass Policy Manager (CPPM) webserver for authentication and identity information for an individual user. For the SRX Series device to be able to make a request, you must have configured it to obtain an access token. See token-api (Services User Identification).

The integrated ClearPass authentication and enforcement user query function supplements the Web API function (webapi) by allowing the SRX Series device to obtain from the CPPM authentication information for an individual user whose information does not already exist in the SRX Series ClearPass authentication table.

Consider the following **query-api** example:

```
api/v1/insight/endpoint/ip/$IP$
```

The SRX Series device generates the complete URL for the user query request by combining the query-api string with the connection method (HTTPS) and the CPPM webserver IP address ({$server}).

```
https://{$server}/api/v1/insight/endpoint/ip/$IP$
```

In this example, the SRX Series device replaces the variables with the following values resulting in a specific URL request for the individual user:

```
https://203.0.113.76/api/v1/insight/endpoint/ip/192.0.2.98
```

Under normal circumstances, the ClearPass webserver sends user authentication information to the SRX Series device in POST request messages and the SRX Series device writes that information to its ClearPass authentication table. When the SRX Series device receives an access request from a user, it searches its ClearPass authentication table for an entry for that user.

It can happen that the SRX Series device might not have received authentication for a user from the CPPM because the user has not yet been authenticated by the CPPM. For example, the user might have joined the network through an access layer not on a managed switch or WLAN. When the CPPM receives the user query from the SRX Series device, it authenticates the user and returns the authentication information to the device.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

radius-options (Access)

Syntax

```
radius-options {
  revert-interval seconds;
}
```

Hierarchy Level

[edit access];
[edit access profile *profile-name*]

Release Information

Statement introduced in Release 8.5 of Junos OS.

Description

Configure RADIUS options.

Options

The remaining statement is explained separately.

Required Privilege Level

access—To view this statement in the configuration.
access-control—To add this statement to the configuration.

Related Documentation

- *Junos OS Security Configuration Guide*

radius-server (Access)

Syntax

```
radius-server server-address {
    port port-number;
    retry attempts;
    routing-instance routing-instance-name;
    secret password;
    source-address source-address;
    timeout seconds;
}
```

Hierarchy Level

[edit access],
[edit access profile *profile-name*]

Release Information

Statement modified in Junos OS Release 8.5.

Description

Configure RADIUS for Layer 2 Tunneling Protocol (L2TP) or Point-to-Point Protocol (PPP) authentication.

To configure multiple RADIUS servers, include multiple **radius-server** statements. The servers are tried in order and in a round-robin fashion until a valid response is received from one of the servers or until all the configured retry limits are reached.

Options

server-address—Address of the RADIUS authentication server.

The remaining statements are explained separately.

Required Privilege Level

access—To view this statement in the configuration.
access-control—To add this statement to the configuration.

Related Documentation

- Firewall User Authentication Overview on page 9

range (Access)

Syntax

```
range range-name {
  high upper-limit;
  low lower-limit;
  prefix-length delegated-prefix-length;
}
```

Hierarchy Level

[edit access address-assignment pool pool-name family inet6]
[edit access address-assignment pool pool-name family inet]

Release Information

Statement introduced in Release 10.4 of Junos OS.

Description

Configure an IP name range used within an address-assignment pool. For IPv4, you do not create a prefix-length.

Options

- *range-name*—Name of the range.
- **high** *upper-limit*—Upper limit of IPv6 address range.
- **low** *lower-limit*—Lower limit of IPv6 address range.
- **prefix-length** *delegated-prefix-length*—IPv6 delegated prefix length.

Required Privilege Level

access—To view this statement in the configuration.
access-control—To add this statement to the configuration.

Related Documentation

- Firewall User Authentication Overview on page 9

rate-limit (Security Log)

Syntax rate-limit *rate-limit*;

Hierarchy Level [edit security log stream *stream-name*]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description The Integrated Authentication and Enforcement feature sends threat and attack logs generated by the SRX Series device security modules to the ClearPass Policy Manager (CPPM) to use in its security policy assessment.

The logs are sent in stream mode. To avoid overburdening the SRX Series device and the log server, you can control the rate at which these logs are sent. By setting a rate-limit value, you can constrain the number of logs that are sent in 1 second. After the limit is reached, no more logs are sent.

Range: 1 through 65,535.

NOTE: For high-end multicore systems that use SPUs, the number of log messages sent per SPU is a divided rate:

rate = configured-rate/number-of-SPUs

Rate limiting on high-end platforms is generally not as accurate as it is on low-end platforms, because the generation of logs is not entirely balanced between SPUs.

Required Privilege Level security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

redirect-traffic

Syntax redirect-traffic (all | unauthenticated);

Hierarchy Level [edit services unified-access-control captive-portal *policy*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify to redirect traffic destined for protected sources to the IC Series device. You can choose to redirect all traffic or only unauthenticated traffic.

Options

- **all**—Redirect all traffic destined for the protected sources to the IC Series device. Specify this option if you want to redirect all traffic (IPsec or source IP) to the currently connected IC Series device or to an IP address or domain name that you specify in a redirect URL.
- **unauthenticated**—Redirect unauthenticated traffic destined for the protected sources to the IC Series device. Select this option if your deployment uses source IP only or a combination of source IP and IPsec. The Junos OS Enforcer redirects clear-text traffic from unauthenticated users to the currently connected IC Series device or to an IP address or domain name that you specify in a redirect URL.

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

Related Documentation

- Firewall User Authentication Overview on page 9

redirect-url

Syntax redirect-url *url*;

Hierarchy Level [edit services unified-access-control captive-portal *policy*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Specify to redirect traffic destined for protected sources to a specified URL.

You can configure the following options in the redirect URL string:

- ***%dest-url%***—Specifies the protected resource which the user is trying to access.
- ***%enforcer-id%***—Specifies the ID assigned to the Junos OS Enforcer by the IC Series device.
- ***%policy-id%***—Specifies the encrypted policy ID for the security policy that redirected the traffic.
- ***%dest-ip%***—Specifies the IP address or hostname of the protected resource that the user is trying to access.
- ***%ic-ip%***—Specifies the IP address or hostname of the IC Series device to which the Junos OS Enforcer is currently connected.

If you do not specify the redirect URL, the Junos OS Enforcer uses the following default configuration:

```
https://%ic-ip%/?target = %dest-url%  &enforcer = %enforcer-id%  &policy =
%policy-id%  &dest-ip = %dest-ip%
```

NOTE: The maximum size of a redirect payload is 1450 bytes. The size of the redirect URL is restricted to 1407 bytes (excluding a few HTTP headers). If a user accesses a destination URL that is larger than 1407 bytes, the Infranet Controller authenticates the payload, calculates the exact length of the redirect URL, and trims the destination URL so that it can fit into the redirect URL. The destination URL can be fewer than 1407 bytes based on what else is present in the redirect URL (for example, policy ID). The destination URL in the default redirect URL is trimmed so that the redirect packet payload size is limited to 1450 bytes. If the length of the payload is larger than 1450 bytes, the excess length is trimmed and the user is directed to the destination URL that has been resized to 1450 bytes.

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

Related Documentation

- Firewall User Authentication Overview on page 9

retry (Access LDAP)

Syntax retry *attempts*;

Hierarchy Level [edit access ldap-server *server-address*],
[edit access profile *profile-name* ldap-server *server-address*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the number of retries that a device can attempt to contact an LDAP server.

Options ***attempts***—Number of retries that the device is allowed to attempt to contact an LDAP server.

Range: 1 through 10

Default: 3

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

Related Documentation
- *Junos OS Security Configuration Guide*

retry (Access RADIUS)

Syntax retry *attempts*;

Hierarchy Level [edit access radius-server *server-address*],
[edit access profile *profile-name* radius-server *server-address*]

Release Information Statement modified in Release 8.5 of Junos OS.

Description Specify the number of retries that a device can attempt to contact a RADIUS authentication server.

Options ***attempts***—Number of retries that the device is allowed to attempt to contact a RADIUS server.

Range: 1 through 10

Default: 3

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

Related Documentation
- *Junos OS Security Configuration Guide*

revert-interval (Access LDAP)

Syntax revert-interval *seconds*;

Hierarchy Level [edit access ldap-options],
[edit access profile *profile-name* ldap-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the amount of time that elapses before the primary server is contacted if a backup server is being used.

Options ***seconds***—Number of seconds that elapse before the primary server is contacted.

Range: 60 through 4,294,967,295 seconds

Default: 600 seconds

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

Related Documentation
- *Junos OS Security Configuration Guide*

revert-interval (Access RADIUS)

Syntax revert-interval *seconds*;

Hierarchy Level [edit access radius-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the amount of time that elapses before the primary server is contacted if a backup server is being used.

Options ***seconds***—Number of seconds that elapse before the primary server is contacted.

Range: 60 through 4,294,967,295 seconds

Default: 600 seconds

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

Related Documentation
- *Junos OS Security Configuration Guide*

root-ca (Services)

Syntax root-ca *root-certificate*;

Hierarchy Level [edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Root certificate for interdicting server certificates in proxy mode. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options *root-ca-name*—Specify root certificate for interdicting server certificates in proxy mode.

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

Related Documentation
- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

routing-instance (Access LDAP)

Syntax routing-instance *routing-instance-name*;

Hierarchy Level [edit access ldap-server *server-address*],
[edit access profile *profile-name* ldap-server *server-address*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure the routing instance used to send LDAP packets to the LDAP server. A routing instance is a collection of routing tables, the interfaces contained in the routing tables, and the routing protocol parameters that control the information in the routing tables.

Options ***routing-instance-name***—Name of the routing instance.

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

Related Documentation
- *Junos OS Security Configuration Guide*

routing-instance (Access RADIUS)

Syntax routing-instance *routing-instance-name*;

Hierarchy Level [edit access radius-server *server-address*],
[edit access profile *profile-name* radius-server *server-address*]

Release Information Statement modified in Release 8.5 of Junos OS.

Description Configure the routing instance used to send RADIUS packets to the RADIUS server. A routing instance is a collection of routing tables, the interfaces contained in the routing tables, and the routing protocol parameters that control the information in the routing tables.

Options ***routing-instance-name*** —Name of the routing instance.

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

Related Documentation
- *Junos OS Security Configuration Guide*

search

Syntax
```
search {
  admin-search {
    distinguished-name distinguished-name;
    password password;
  }
  search-filter filter-name;
}
```

Hierarchy Level [edit access ldap-options],
[edit access profile *profile-name* ldap-options]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify that a search is used to get a user's LDAP distinguished name (DN). The search is performed based on the search filter and the part typed in by the user during authentication.

Options The remaining statements are explained separately.

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

Related Documentation
- *Junos OS Security Configuration Guide*

search-filter

Syntax search-filter *filter-name*;

Hierarchy Level [edit access ldap-options search],
[edit access profile ***profile-name*** ldap-options search]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify that a search filter is used to find the user's LDAP distinguished name (DN). For example, a filter of **cn** specifies that the search matches a user whose common name is the username.

Options ***filter-name***—Name of the filter used to find the user's distinguished name.

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

Related Documentation
- *Junos OS Security Configuration Guide*

secret (Access Profile)

Syntax secret ***password***;

Hierarchy Level [edit access profile ***profile-name*** radius-server ***server-address***]

Release Information Statement modified in Release 8.5 of Junos OS.

Description Specify the RADIUS secret password, which is shared between the router and the RADIUS server. The device uses this secret to encrypt the user's password that is sent to the RADIUS server.

Options ***password***—RADIUS secret. Maximum length is 256 characters.

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

Related Documentation
- *Junos OS Security Configuration Guide*

securid-server

Syntax

```
securid-server {
    server-name configuration-file filepath;
}
```

Hierarchy Level [edit access]

Release Information Statement introduced in Release 9.1 of Junos OS.

Description Configure SecurID server for SecurID authentication type.

Options The remaining statement is explained separately.

NOTE: You can configure only one SecurID server. SecurID challenges are not yet supported.

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

Related Documentation

- *Junos OS Security Configuration Guide*

separator

Syntax separator *special-character*;

Hierarchy Level [edit access profile *profile-name* client-name-filter *client-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify a character to identify where stripping of characters occurs in a client name. Stripping removes characters to the right of each instance of the specified character, plus the character itself. The stripping begins with the rightmost separator character.

Use the **separator** statement with the **count** statement to determine which characters in a client name are stripped. If the specified number of separator characters (count) exceeds the actual number of separator characters in the client name, stripping stops at the last available separator character.

Options ***special-character***—Character used to identify where to start the stripping of characters in a client name.

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

Related Documentation
- *Junos OS Security Configuration Guide*

server-certificate (Services)

Syntax server-certificate *server-certificate*;

Hierarchy Level [edit services ssl termination profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the local certificate identifier. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options **server-certificate**—Specify the name of the local certificate identifier.

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

Related Documentation
- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

server-certificate-subject

Syntax server-certificate-subject *subject*;

Hierarchy Level [edit services unified-access-control infranet-controller *hostname*]

Release Information Statement introduced in Junos OS Release 9.4.

Description Optionally specify the full subject name of the certificate that the SRX Series device should use to validate the IC Series device's server certificate.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

Related Documentation
- ca-profile (Services) on page 306
- password (Services) on page 362

session-options (Access Profile)

Syntax
```
session-options {
  client-group [group-names];
  client-idle-timeout minutes;
  client-session-timeout minutes;
}
```

Hierarchy Level [edit access profile *profile-name*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Define options that control a user's session after successful authentication.

Options The remaining statements are explained separately.

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

Related Documentation
- *Junos OS Security Configuration Guide*

size (Services)

Syntax size *size*;

Hierarchy Level [edit services ssl traceoptions file *file-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the maximum trace file size. This statement is supported on the SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options **size**—Specify the maximum trace file size.

Range: 10,240 to 1,073,741,824.

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

Related Documentation
- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

source-address (Access LDAP)

Syntax source-address *source-address*;

Hierarchy Level [edit access ldap-server server-address],
[edit access profile *profile-name* ldap-server *server-address*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure a source address for each configured LDAP server. Each LDAP request sent to a LDAP server uses the specified source address.

Options ***source-address***—Valid IP address configured on one of the device interfaces.

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

Related Documentation
- *Junos OS Security Configuration Guide*

source-address (Access RADIUS)

Syntax source-address *source-address*;

Hierarchy Level [edit access radius-server *server-address*],
[edit access profile *profile-name* radius-server *server-address*]

Release Information Statement modified in Junos OS Release 8.5.

Description Configure a source address for each configured RADIUS server. Each RADIUS request sent to a RADIUS server uses the specified source address.

Options *source-address—Valid IP address configured on one of the device interfaces.*

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

Related Documentation

- Firewall User Authentication Overview on page 9

source-end-user-profile

Syntax source-end-user-profile *device-identity-profile-name*;

Hierarchy Level [edit security policies from-zone *from-zone* to-zone *to-zone* policy *policy-name* match]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description The **source-end-user-profile** field in a security policy enables you to specify a device identity profile that identifies the traffic source based on the device from which the traffic issued. The security policy action is applied to traffic issuing from a device if the device matches the attributes specified in the profile and it matches the rest of the security policy parameters.

The device identity profile feature provides a solution for cases in which you cannot or do not want to use the user identity to control access to network resources. The device identity feature allows you to use the identity of a device and its attributes to control access to network resources instead of the identity of the user of that device.

You might want to control network access based on the device identity for various reasons. For example, you might allow your users to use their own devices (BYOD) to access network resources and you do not want to use captive portal authentication. Also, some companies might have older switches that do not support 802.1, or they might not have a Network Access Control (NAC) system.

Options **device-identity-profile-name**—Device identity profile that specifies characteristics that can apply to one or more devices.

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

Related Documentation

- Understanding Access Control to Network Resources Based on Device Identity Information on page 185
- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192

source-address (Access RADIUS)

Syntax source-address *source-address*;

Hierarchy Level [edit access radius-server *server-address*],
[edit access profile *profile-name* radius-server *server-address*]

Release Information Statement modified in Junos OS Release 8.5.

Description Configure a source address for each configured RADIUS server. Each RADIUS request sent to a RADIUS server uses the specified source address.

Options *source-address—Valid IP address configured on one of the device interfaces.*

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

Related Documentation

- Firewall User Authentication Overview on page 9

source-identity-log (Security)

Syntax source-identity-log

Hierarchy Level [edit security zones security-zone *zone-name*]

Release Information Statement introduced in Junos OS Release 15.1X49-D60.

Description Specify the source-identity-log parameter as part of the configuration for a zone to enable it to trigger user identity logging when that zone is used as the source zone (from-zone) in a security policy. If a zone is configured for zone-based user identity logging and it is used as the source zone in a security policy, the system logs the user identity of any user who belongs to that zone and whose traffic matches the security policy's terms.

A zone configured for zone-based user identity logging is reusable. That is, you can use it as the source zone in any security policy.

For zone-based user identity logging to occur, you must have configured the session initialization (session-init) and the session termination (session-close) events as actions for the security policy.

Zone-based user identity logging allows you to broaden the scope of users whose identities are recorded in the session log. The source-identity security policy tuple writes the user or group name to log, but it restricts application of the security policy to the specified user or user group.

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

Related Documentation

- Understanding How to Include User Identity Information in the Session Log File Based on the Source Zone on page 177
- Example: Configuring Integrated User Firewall to Write User Identity to the Session Log Based On the Source Zone on page 179
- Overview of Integrated User Firewall on page 141
- Example: Configuring Integrated User Firewall on page 153

ssl (Services)

Syntax

```
ssl {
  initiation {
    profile profile-name {
      actions {
        ignore-server-auth-failure;
      }
      client-certificate;
      custom-ciphers [cipher];
      enable-flow-tracing;
      enable-session-cache;
      preferred-ciphers (custom | medium | strong | weak);
      protocol-version (all | tls1 | tls11 | tls12);
      trusted-ca (all | [ca-profile] );
    }
  }
  proxy {
    global-config {
      session-cache-timeout seconds;
    }
    profile profile-name {
      actions {
        crl {
          disable;
          if-not-present (allow | drop);
          ignore-hold-instruction-code;
        }
        disable-session-resumption;
        ignore-server-auth-failure;
        log {
          all;
          errors;
          info;
          sessions-allowed;
          sessions-dropped;
          sessions-ignored;
          sessions-whitelisted;
          warning;
        }
        renegotiation {
          (allow | allow-secure | drop);
        }
      }
      custom-ciphers [cipher];
      enable-flow-tracing;
      preferred-ciphers (custom | medium | strong | weak);
      root-ca root-certificate;
      trusted-ca (all | [ca-profile] );
      whitelist [global-address-book-addresses];
    }
  }
  termination {
    profile profile-name {
```

```
        custom-ciphers [cipher];
        enable-flow-tracing;
        enable-session-cache;
        preferred-ciphers (custom | medium | strong | weak);
        protocol-version (all | tls1 | tls11 | tls12);
        server-certificate certificate-identifier;
      }
    }
    traceoptions {
      file {
        filename;
        files number;
        match regular-expression;
        (no-world-readable | world-readable);
        size maximum-file-size;
      }
      flag flag;
      level [brief | detail | extensive | verbose];
      no-remote-trace;
    }
}
```

Hierarchy Level [edit services]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. The **crl** statement is supported from 15.1X49-D30. The **protocol-version** statement is updated to include **tls11** and **tls12** from Junos OS Release 15.1X49-D30.

Description Specify the configuration for Secure Socket Layer (SSL) support service. This statement is supported on the SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options T

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

ssl-termination-profile

Syntax ssl-termination-profile *profile-name*;

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit firewall-authentication pass-through]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the SSL termination profile used for SSL offloading.

Options ***profile-name***—Specify the name of the SSL termination profile used to the SSL offload.

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

Related Documentation
- *Security Policies Overview*

success

Syntax success *string*;

Hierarchy Level [edit access firewall-authentication pass-through default-profile *name* (ftp | http | telnet) banner],
[edit access firewall-authentication web-authentication]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Specify the banner (message) that users see when trying to connect using FTP, HTTP, or Telnet after successful authentication.

Options ***string***—Banner text. Maximum length of the message text is 250 characters. Enclose the banner text within spaces or special characters—for example, quotation marks (" ").

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

Related Documentation
- *Junos OS Security Configuration Guide*

telnet (Access)

Syntax

```
telnet {
  banner {
    fail string;
    login string;
    success string;
  }
}
```

Hierarchy Level [edit access firewall-authentication pass-through]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure banners for Telnet login prompt, successful authentication, and failed authentication.

Options The remaining statements are explained separately.

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

Related Documentation

- *Junos OS Security Configuration Guide*

termination (Services)

Syntax

```
termination {
  profile profile-name {
    custom-ciphers [cipher];
    enable-flow-tracing;
    enable-session-cache;
    preferred-ciphers (custom | medium | strong | weak);
    protocol-version (all | tls1 | tls11 | tls12);
    server-certificate certificate-identifier;
  }
}
```

Hierarchy Level [edit services ssl]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. The **protocol-version** statement is updated to include **tls11** and **tls12** from Junos OS Release 15.1X49-D30.

Description Specify the configuration for Secure Socket Layer (SSL) termination support service.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

test-only-mode

Syntax test-only-mode (true | false):

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 9.4.

Description Configure the device in test-only mode to log access decisions from the IC Series device without actually enforcing the decisions. When configured in test-only mode, the SRX Series device enables all UAC traffic to go through so you can test the implementation without impeding traffic.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

then (Security Policies)

Syntax

```
then {
  count {
    alarm {
      per-minute-threshold number;
      per-second-threshold number;
    }
  }
  deny;
  log {
    session-close;
    session-init;
  }
  permit {
    application-services {
      application-firewall {
        rule-set rule-set-name;
      }
      application-traffic-control {
        rule-set rule-set-name;
      }
      gprs-gtp-profile profile-name;
      gprs-sctp-profile profile-name;
      idp;
      redirect-wx | reverse-redirect-wx;
      ssl-proxy {
        profile-name profile-name;
      }
      uac-policy {
        captive-portal captive-portal;
      }
      utm-policy policy-name;
    }
    destination-address {
      drop-translated;
      drop-untranslated;
    }
    firewall-authentication {
      pass-through {
        access-profile profile-name;
        client-match user-or-group-name;
        ssl-termination-profile profile-name;
        web-redirect;
        web-redirect-to-https;
      }
      user-firewall {
        access-profile profile-name;
        domain domain-name
        ssl-termination-profile profile-name;
      }
      web-authentication {
        client-match user-or-group-name;
      }
```

```
        }
        services-offload;
        tcp-options {
            initial-tcp-mss mss-value;
            reverse-tcp-mss mss-value;
            sequence-check-required;
            syn-check-required;
        }
        tunnel {
            ipsec-group-vpn group-vpn;
            ipsec-vpn vpn-name;
            pair-policy pair-policy;
        }
    }
    reject;
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name*]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **services-offload** option added in Junos OS Release 11.4. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20.

Description Specify the policy action to be performed when packets match the defined criteria.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

timeout (Access LDAP)

Syntax timeout *seconds*;

Hierarchy Level [edit access ldap-server *server-address*]
[edit access profile *profile-name* ldap-server *server-address*]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Configure the amount of time that the local device waits to receive a response from an LDAP server.

Options ***seconds***—Amount of time to wait.
Range: 1 through 90 seconds
Default: 3 seconds

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

Related Documentation
- *Junos OS Security Configuration Guide*

timeout (Access RADIUS)

Syntax timeout *seconds*;

Hierarchy Level [edit access radius-server *server-address*]
[edit access profile *profile-name* radius-server *server-address*]

Release Information Statement modified in Release 8.5 of Junos OS.

Description Configure the amount of time that the local device waits to receive a response from a RADIUS server.

Options ***seconds***—Amount of time to wait.
Range: 1 through 90 seconds
Default: 3 seconds

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

Related Documentation
- Firewall User Authentication Overview on page 9

timeout (Services)

Syntax timeout *seconds*;

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify the value, in seconds, that the SRX Series device should wait to get a heartbeat response from an IC Series UAC Appliance (default is 300). If the SRX Series device does not receive it in the specified time, it takes the action specified by the **timeout-action** configuration statement. It also tries again to make a connection to the IC Series appliance. After the second failed attempt, the SRX Series device fails over to the next IC Series appliance in the cluster. The SRX Series device continues trying to reach IC Series appliances in the cluster until a connection is established.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series appliance. When working with a cluster of IC Series appliances, the Junos OS Enforcer connects to one at a time, failing over to other IC Series appliances in the cluster as required.

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

Related Documentation

- interval (Services) on page 348
- timeout-action on page 407

timeout-action

Syntax timeout-action (close | no-change | open):

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 9.4.

Description Specify what the SRX Series device should do when a timeout occurs and the device cannot connect to an Infranet Enforcer.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series UAC Appliance.

Options

- **close**—Close existing sessions and block any further traffic. This is the default option.
- **no-change**—Preserve existing sessions and require authentication for new sessions.
- **open**—Preserve existing sessions and allow new sessions access.

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

Related Documentation

- interval (Services) on page 348
- timeout (Services) on page 406

tls-min-version

Syntax tls-min-version (v1.1 | v1.2);

Hierarchy Level [edit access profile *profile-name* ldap-server *ip-address*]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Configure Transport Layer Security (TLS) version to limit the lowest supported versions of TLS that are enabled for SSL connections.

Options

v1.1—Accept TLS version 1.1. This enhanced version of TLS provides protection against cipher-block chaining (CBC) attacks.

v1.2 —Accept TLS version 1.2. This enhanced version of TLS provides improved flexibility for negotiation of cryptographic algorithms.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Example: Configuring Integrated User Firewall on page 153

tls-peer-name

Syntax tls-peer-name*peer-host-name*;

Hierarchy Level [edit access profile *profile-name* ldap-server *ip-address*]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Configure the peer hostname to be authenticated.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11
- LDAP Functionality in Integrated User Firewall on page 150

tls-timeout

Syntax tls-timeout *seconds*;

Hierarchy Level [edit access profile *profile-name* ldap-server *ip-address*]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Specify timeout value on the Transport Layer Security (TLS) handshake. The TLS handshake is responsible for the encryption keys exchange necessary to establish secure sessions between client and server.

Range: 3 through 90 seconds.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11
- LDAP Functionality in Integrated User Firewall on page 150

tls-type

Syntax

```
tls-type {
  start-tls;
}
```

Hierarchy Level [edit access profile *profile-name* ldap-server *ip-address*]

Release Information Statement introduced in Junos OS Release 15.1X49-D70.

Description Configure Lightweight Directory Access Protocol (LDAP) over Secure Sockets Layer/Transport Layer Security (SSL/TLS) for secure communication. Transport Layer Security StartTLS extension for LDAP is used for the firewall user authentication and the integrated user firewall authentication for obtaining username and role information through firewall authentication. StartTTLS allows protocol data transfers between the LDAP server and client over the TLS layer after successful negotiation between the peers. StartTLS upgrades an existing insecure LDAP connection to a secure Secure Sockets Layer/Transport Layer Security (SSL/TLS) connection.

Options

- **start-tls**—Configure LDAP over StartTLS. The StartTLS communications occurs over TCP port 389.

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

Related Documentation

- Firewall User Authentication Overview on page 9
- Obtaining Username and Role Information Through Firewall Authentication on page 11
- LDAP Functionality in Integrated User Firewall on page 150

token-api (Services User Identification)

Syntax token-api *token-api*

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Configure the token API that is used in generating the URL for acquiring an access token. The token API is combined with the connection method and the IP address of the ClearPass webserver to produce the complete URL used for acquiring an access token.

For example, if the token API is oauth, the connection method is HTTPS, and the IP address of the ClearPass webserver is 192.0.2.199, the complete URL for acquiring an access token would be https://192.0.2.199/api/oauth. This is a required parameter. There is no default value.

The SRX Series device user query function requires an access token to be able to query the ClearPass webserver. If the user query function is configured, the SRX Series device can request from the ClearPass webserver user authentication and identity information for an individual user.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

to-zone (Security Policies)

Syntax

```
to-zone zone-name {
  policy policy-name {
    description description;
    match {
      application {
        [application];
        any;
      }
      destination-address {
        [address];
        any;
        any-ipv4;
        any-ipv6;
      }
      source-address {
        [address];
        any;
        any-ipv4;
        any-ipv6;
      }
      source-identity {
        [role-name];
        any;
        authenticated-user;
        unauthenticated-user;
        unknown-user;
      }
    }
    scheduler-name scheduler-name;
    then {
      count {
        alarm {
          per-minute-threshold number;
          per-second-threshold number;
        }
      }
      deny;
      log {
        session-close;
        session-init;
      }
      permit {
        application-services {
          application-firewall {
            rule-set rule-set-name;
          }
          application-traffic-control {
            rule-set rule-set-name;
          }
          gprs-gtp-profile profile-name;
          gprs-sctp-profile profile-name;
          idp;
```

```
            redirect-wx | reverse-redirect-wx;
            ssl-proxy {
              profile-name profile-name;
            }
            uac-policy {
              captive-portal captive-portal;
            }
            utm-policy policy-name;
          }
          destination-address {
            drop-translated;
            drop-untranslated;
          }
          firewall-authentication {
            pass-through {
              access-profile profile-name;
              client-match user-or-group-name;
              ssl-termination-profile profile-name;
              web-redirect;
              web-redirect-to-https;
            }
            web-authentication {
              client-match user-or-group-name;
            }
          }
          services-offload;
          tcp-options {
            sequence-check-required;
            syn-check-required;
          }
          tunnel {
            ipsec-group-vpn group-vpn;
            ipsec-vpn vpn-name;
            pair-policy pair-policy;
          }
        }
        reject;
      }
    }
  }
```

Hierarchy Level [edit security policies from-zone *zone-name*]

Release Information Statement introduced in Junos OS Release 8.5. Support for the **services-offload** and **junos-host** options added in Junos OS Release 11.4. Support for the **source-identity** option added in Junos OS Release 12.1. Support for the **ssl-termination-profile** and **web-redirect-to-https** options added in Junos OS Release 12.1X44-D10.

Description Specify a destination zone to be associated with the security policy.

Options

- **zone-name**—Name of the destination zone object.
- **junos-host**—Default security zone for self-traffic of the device.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

traceoptions (Access)

Syntax

```
traceoptions {
  file filename {
    files number;
    match regular-expression;
    size maximum-file-size;
    <world-readable | no-world-readable>;
  }
  flag flag;
}
```

Hierarchy Level [edit access firewall-authentication]

Release Information Statement introduced in Release 8.5 of Junos OS.

Description Define Routing Engine firewall authentication tracing options.

Options

- **file *filename***—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**.
- **files *number***—(Optional) Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed to ***trace-file.0***, then ***trace-file.1***, and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten.
- If you specify a maximum number of files, you also must specify a maximum file size with the size option and a filename.

  Range: 2 through 1000 files

  Default: 10 files

- **match *regular-expression***—(Optional) Refine the output to include lines that contain the regular expression.
- **size*maximum-file-size***—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named ***trace-file*** reaches this size, it is renamed ***trace-file.0***. When the **trace-file** again reaches its maximum size, ***trace-file.0*** is renamed ***trace-file.1*** and ***trace-file*** is renamed ***trace-file.0***. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.
- If you specify a maximum file size, you also must specify a maximum number of trace files with the files option and filename.

  Syntax: ***x*** **k** to specify KB, ***x*** **m** to specify MB, or ***x*** **g** to specify GB

  Range: 10 KB through 1 GB

  Default: 128 KB

- **world-readable | no-world-readable**—(Optional) By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option

enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

- **flag *flag***—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. You can include the following flags.
- **all**—All tracing operations
- **authentication**—Trace authentication events
- **configuration**—Trace configuration events
- **setup**—Trace setup of firewall authentication service

Required Privilege Level

trace—To view this statement in the configuration.
trace-control—To add this statement to the configuration.

Related Documentation

- Firewall User Authentication Overview on page 9

traceoptions (Active Directory Access)

Syntax

```
traceoptions {
  file filename ;
  flag {
    active-directory-authentication;
    all;
    configuration;
    db;
    ip-user-mapping;
    ip-user-probe;
    ipc;
    user-group-mapping;
    wmic;
  }
  level {
    all
    error
    info
    notice
    verbose
    warning
  }
  no-remote-trace;
}
```

Hierarchy Level [edit services user-identification active-directory-access]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description Define Active Directory trace options for the integrated user firewall feature.

Options

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**.

flag—Trace the operation or operations to perform on the integrated user firewall. To specify more than one trace operation, include multiple flag statements.

- **active-directory-authentication**—Trace the building of and modifications to the Active Directory authentication table.
- **all**—Trace everything.
- **configuration**—Trace configuration events.
- **db**—Trace the database.
- **ip-user-mapping**—Trace the ip-user-mapping module.
- **ip-user-probe**—Trace PC client probing.
- **ipc**—Trace communication events with the Packet Forwarding Engine.
- **user-group-mapping**—Trace the process of getting user-to-group-mapping.

wmic—Trace the Windows Management Instrumentation Client process.

level—Level of trace operation to perform.

all—Match all levels.

error—Match error conditions.

info—Match informational messages.

notice—Match conditions that should be handled specially.

verbose—Match verbose messages.

warning—Match warning messages.

no-remote-trace—Disallow tracing from a remote device.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- active-directory-access on page 294
- user-identification (Services) on page 430
- Overview of Integrated User Firewall on page 141

traceoptions (Security Firewall Authentication)

Syntax

```
traceoptions {
  flag {
    all <detail | extensive | terse>;
    authentication <detail | extensive | terse>;
    proxy <detail | extensive | terse>;
  }
}
```

Hierarchy Level [edit security firewall-authentication]

Release Information Statement introduced in Junos OS Release 8.5.

Description Define data-plane firewall authentication tracing options.

Options

- **flag**—Trace operation to perform. To specify more than one trace operation, include multiple **flag** statements.
  - **all**—Enable all tracing operations
  - **authentication**—Trace data-plane firewall authentication events
  - **proxy**—Trace data-plane firewall authentication proxy events
- **detail**—Display moderate amount of data in trace.
- **extensive**—Display extensive amount of data in trace.
- **terse**—Display minimum amount of data in trace.

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

Related Documentation

- Firewall User Authentication Overview on page 9

traceoptions (Services SSL)

Syntax

```
traceoptions {
  file {
    filename;
    files number;
    match regular-expression;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  flag flag;
  level [brief | detail | extensive | verbose];
  no-remote-trace;
}
```

Hierarchy Level [edit services ssl]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the trace file information. This statement is supported on the SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

- *file-name*—Specify the name of file in which to write trace information.
  - **files**—Specify the maximum number of trace files. Range: 2 to 1000.
  - **match**—Specify the regular expression for lines to be logged.
  - **no-world-readable size**—Do not allow any user to read the log file.
  - **size**—Specify the maximum trace file size. Range: 10,240 to 1,073,741,824.
  - **world-readable**—Allow any user to read the log file.
- **flag**—Trace operation to perform. To specify more than one trace operation, include multiple **flag** statements.
  - **all**—Trace with all flags enabled
  - **compilation**—Trace rule set compilation events
  - **configuration**—Trace configuration events
  - **ipc**—Trace process inter communication events
  - **lookup**—Trace rule set lookup events
- **level**—Set the level of debugging the output option.
  - **brief**—Match brief messages.
  - **detail**—Match detail messages.
  - **extensive**—Match extensive messages.
  - **verbose**—Match verbose messages.
- **no-remote-trace**—Set remote tracing as disabled.

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

traceoptions (Services UAC)

Syntax

```
traceoptions {
  file {
    filename;
    files number;
    match regular-expression;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  flag flag;
  no-remote-trace;
}
```

Hierarchy Level [edit services unified-access-control]

Release Information Statement introduced in Junos OS Release 9.4.

Description Define Unified Access Control (UAC) tracing options.

Use this statement when you are configuring the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series UAC Appliance.

Options

flag—Trace operation to perform. To specify more than one trace option, include multiple flag statements.

- **all**—Trace with all flags enabled
- **config**—Trace configuration information for all UAC-related configurations. This includes all configuration controlled through the **unified-access-control** statements at the **edit services** hierarchy level. It also includes other standard Junos OS configurations required for UAC enforcement such as zones, policies, and interfaces.
- **connect**—Trace communications between the Junos OS Enforcer and the IC Series appliance, including SSL handshakes and timeouts.
- **ipc**—Trace interprocess communications. Use this option to trace communications between the Routing Engine (RE) and the UACD enforcement plugin inside the Packet Forwarding Engine (PFE).

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- Understanding Unified Access Control on page 101
- Acquiring User Role Information from an Active Directory Authentication Server on page 101

traceoptions (Services User Identification)

Syntax

```
traceoptions {
  file {
    filename;
    files number;
    match regular-expression;
    size maximum-file-size;
    (world-readable | no-world-readable);
  }
  flag flag;
  level level ;
  no-remote-trace;
}
```

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the name of the trace log file and its characteristics. Messages about the behavior of the authentication source are written to this log file. Aruba ClearPass Policy Manager (CPPM) is the authentication source for the SRX Series device integrated ClearPass authentication and enforcement feature.

Required Privilege Level

services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

trusted-ca (Services)

Syntax trusted-ca (all | [*ca-profile*]);

Hierarchy Level

[edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]
[edit services ssl initiation profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the list of trusted certificate authority profiles. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices, and vSRX.

Options

- *trusted-ca-name*—Specify the certificate authority profile name.
- **all**—Select all certificate authority profiles.

Required Privilege Level

services—To view this statement in the configuration.
services-control—To add this statement to the configuration.

Related Documentation

- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

uac-policy (Application Services)

Syntax

```
uac-policy {
  captive-portal captive-portal;
}
```

Hierarchy Level

[edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit application-services]

Release Information

Statement modified in Junos OS Release 9.4.

Description

Enable Unified Access Control (UAC) for the security policy. This statement is required when you are configuring the SRX Series device to act as a Junos OS Enforcer in a UAC deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series UAC Appliance .

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- *Understanding User Role Firewalls*
- *Example: Configuring a User Role Firewall on an SRX Series Device*

uac-service

Syntax

```
uac-service {
  command binary-file-path;
  disable;
  failover (alternate-media | other-routing-engine);
}
```

Hierarchy Level [edit system processes]

Release Information Statement introduced in Junos OS Release 8.5.

Description Specify the unified access control daemon process.

Options

- **command *binary-file-path***—Path to the binary process.
- **disable**—Disable the unified access control daemon process.
- **failover**—Configure the device to reboot if the software process fails four times within 30 seconds, and specify the software to use during the reboot.
  - **alternate-media**—Configure the device to switch to backup media that contains a version of the system if a software process fails repeatedly.
  - **other-routing-engine**—Instruct the secondary Routing Engine to take mastership if a software process fails. If this statement is configured for a process, and that process fails four times within 30 seconds, then the device reboots from the secondary Routing Engine.

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

Related Documentation

- Firewall User Authentication Overview on page 9

unified-access-control (Security)

Syntax unified-access-control priority *priority*;

Hierarchy Level [edit security user-identification authentication-source]

Release Information Statement introduced in Junos OS Release 12.1. Support for **disable** option dropped in Junos OS Release 12.1X47-D10.

Description An authentication table pushed from a configured authentication device, such as the Junos Pulse Access Control Service.

Options priority ***priority***—A unique value between 0 and 65535 that determines the sequence for searching multiple tables to retrieve a user role. Each authentication table is given a unique priority value. The lower the value, the higher the priority. A table with priority 120 is searched before a table with priority 200. The default priority value of the unified-access-control authentication table is 200.

Setting the priority value of the unified-access-control authentication table to 0 is equivalent to disabling the table and eliminating it from the search sequence.

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

Related Documentation

- *authentication-source (Security)*
- *Understanding User Role Firewalls*
- *Understanding the User Identification Table*

unified-access-control (Services)

Syntax

```
unified-access-control {
  captive-portal redirect-policy-name{
    redirect-traffic (all | unauthenticated);
    redirect-url redirect-url;
  }
  certificate-verification [ optional | required | warning ];
  infranet-contoller host-name {
    address ip-address;
    ca-profile [ca-profile];
    interface interface-name;
    password password;
    port port-number;
    server-certificate-subject subject;
  }
  interval seconds;
  test-only-mode;
  timeout seconds;
  timeout-action (close | no-change | open);
  traceoptions {
    file {
      filename;
      files number;
      match regular-expression;
      (no-world-readable | world-readable);
      size maximum-file-size;
    }
    flag flag;
    no-remote-trace;
  }
}
```

Hierarchy Level [edit services]

Release Information Statement introduced in Junos OS Release 9.4.

Description Use this statement to configure the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series device.

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

user-group-mapping

Syntax

```
user-group-mapping {
  ldap {
    address ip-address {
      port port;
    }
    authentication-algorithm {
      simple;
    }
    base base;
    ssl;
    user username {
      password password;
    }
  }
}
```

Hierarchy Level [edit services user-identification active-directory-access domain]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description Configure the SRX Series device to connect to an LDAP server, so that the server can provide the SRX Series with user-to-group mappings. These mappings are used to implement the integrated user firewall feature. The domain controller acts as the LDAP server in typical customer scenarios.

Most of this statement is optional, because the default communication method is LDAP and most arguments have default values. Only the LDAP keyword and the base are required.

Options

ldap—Required. LDAP is the protocol used to access the LDAP server to get user-to-group mappings.

address *ip-address*—Optional. Specify the IP address of the LDAP server. If no address is specified, the system uses one of the configured Active Directory domain controllers.

port *port*—Optional. Specify the port number of the LDAP server. If no port number is specified, the system uses port 389 for plaintext or port 636 for encrypted text.

authentication-algorithm—Optional. Specify the algorithm used while the SRX Series communicates with the LDAP server. The default method is Kerberos.

simple—Configure simple (plaintext) authentication method.

base *base*—Required. LDAP base distinguished name (DN).

ssl—Optional. Enable Secure Sockets Layer (SSL) to ensure secure transmission with the LDAP server. Disabled by default, which means that the password is sent in plaintext.

user *username*—Optional. Username of the LDAP account. If no username is specified, the system will use the configured domain controller's username.

password *password*—Optional. Specify the password for the account. If no password is specified, the system uses the configured domain controller's password.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- active-directory-access on page 294
- clear services user-identification active-directory-access on page 456
- show services user-identification active-directory-access statistics on page 529
- show services user-identification active-directory-access user-group-mapping on page 510
- traceoptions (Active Directory Access) on page 417
- user-identification (Services) on page 430
- LDAP Functionality in Integrated User Firewall on page 150

user-identification (Services)

Syntax

```
user-identification {
  active-directory-access {
    domain domain-name {
      user username;
        password password;
      domain-controller domain-controller-name {
        address domain-controller-address;
      }
      ip-user-mapping {
        discovery-method {
          wmi {
            event-log-scanning-interval seconds;
            initial-event-log-timespan hours;
          }
        }
      }
      user-group-mapping {
        ldap {
          address ip-address {
            port port;
          }
          authentication-algorithm {
            simple;
          }
          base base;
          ssl;
          user username {
            password password;
          }
        }
      }
    }
    authentication-entry-timeout minutes;
    filter {
      include address;
      exclude address;
    }
    no-on-demand-probe;
    wmi-timeout seconds;
    traceoptions {
      file file;
      flag {
        active-directory-authentication;
        all;
        configuration;
        db;
        ip-user-mapping;
        ip-user-probe;
        ipc;
        user-group-mapping;
        wmic;
      }
```

```
        level {
          all;
          error;
          info;
          notice;
          verbose;
          warning;
        }
        no-remote-trace;
      }
    }
```

Hierarchy Level [edit services]

Release Information Statement introduced in Junos OS Release 12.1X47-D10.

Description Configure the integrated user firewall feature, including access to the Active Directory domain and domain controller, IP address-to-user mapping, and user-to-group mapping. One or two Active Directories are allowed under one domain. The IP address-to-user mapping and user-to-group mapping are configured per domain.

Options **authentication-entry-timeout *minutes***—Timeout interval starting from the Active Directory/domain controller login time, the last active session, or the last successful probe. A setting of 0 means the authentication does not need a timeout. We recommend that you configure a setting of 0 when you disable on-demand-probe to prevent someone from accessing the Internet without logging in again.

Range: 10 through 1440 minutes

Default: 30 minutes

filter—Optional. Range of IP addresses that needs to be monitored or not monitored.

include *address*—Include IP address or range. Maximum of 20 addresses.

exclude *address*—Exclude IP address or range. Maximum of 20 addresses.

no-on-demand-probe—Do not use traffic to discover user. Default is disabled.

wmi-timeout *seconds*—Optional. Configures the number of seconds that the domain PC has to respond to the SRX Series device's query through WMI/DCOM.

- If the PC responds within that timeframe to the WMI query, the SRX creates an authentication entry for this PC.
- If the PC does not respond within that timeframe, the WMI query failed. In the case of a failed query, if the SRX had an authentication entry about the queried PC before the WMI query, that authentication entry is deleted. If the SRX had no authentication entry before the WMI query, the SRX does not create an authentication entry.

Range: 3 through 120 seconds

Default: 10 seconds

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation
- active-directory-access on page 294
- traceoptions (Active Directory Access) on page 417

webapi (System Services)

Syntax

```
webapi {
client ip-address;
(
  http {
    port  port-number;
  }
  https {
    certificate  certificate-filename;
    certificate-key  local-certificate-key;
    default-certificate
    pki-local-certificate;
    port port-number;
  }
  user {
    name;
    password password;
  }
  debug-log filename;
  debug-level level;
}
```

Hierarchy Level
[edit system services]

Release Information
Statement introduced in Junos OS Release 12.3X48-D30.

Description
Configure the Web API function daemon (webapi) component of the integrated ClearPass authentication and enforcement feature. The Web API daemon acts as a HTTP or HTTPS server. The SRX Series device exposes to the Aruba ClearPass Policy Manager (CPPM) the Web API that allows the CPPM, as a client, to send POST request messages to it that provide the SRX Series device with user authentication and identity information. The CPPM serves as the user authentication source for the SRX Series device.

The Web API function (webapi) facilitates efficient transmission of user authentication and identity information from the CPPM to the SRX Series device. The CPPM, which is the client in this relationship, initiates a session with the SRX Series device Web API daemon, which is the server in this relationship. However, the CPPM can do this only if you have configured the Web API function on the SRX Series device. For security reasons, the Web API daemon is not enabled by default.

The configuring statements are explained separately. See CLI Explorer.

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

webapi-clear-text (Security)

Syntax web-api-cleartext

Hierarchy Level [edit security zones security-zone *zone* host-inbound-traffic system-services]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Enable the Web API (webapi) service over HTTP host inbound traffic on TCP port 8080 for unencrypted data.

Required Privilege Level

security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

webapi-ssl (Security)

Syntax webapi-ssl

Hierarchy Level [edit security zones security-zone *zone* host-inbound-traffic system-services]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Enable the Web API service over HTTPS host inbound traffic on TCP port 8443.

Required Privilege Level

security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

web-authentication

Syntax

```
web-authentication {
  client-match user-or-group-name;
}
```

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit firewall-authentication]

Release Information Statement introduced in Junos OS Release 8.5.
HTTPS for Web authentication is supported on high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.

Description Specify that the policy allows access to users who have previously been authenticated by Web authentication. Web authentication must be enabled on one of the addresses on the interface to which the HTTP or HTTPS request is redirected.

Options **client-match *user-or-group*** —(Optional) Username or user group name.

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

Related Documentation

- *Understanding User Role Firewalls*

web-authentication (Access)

Syntax

```
web-authentication {
  banner {
    success string;
  }
  default-profile profile-name;
}
```

Hierarchy Level [edit access firewall-authentication]

Release Information Statement introduced in Junos OS Release 8.5.
HTTPS for Web authentication is supported on high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.

Description Specify that users go through the Web authentication process. The user uses HTTP or HTTPS to access an IP address on the device that is enabled for Web authentication. In this scenario, the user does not use HTTP or HTTPS to access the IP address of the protected resource. The user is prompted for a username and password, which are verified by the device. Subsequent traffic from the user or host to the protected resource is allowed or denied based on the results of this authentication. This method of authentication differs from pass-through authentication in that users need to access the protected resource directly after accessing the Web authentication IP address and being authenticated.

Options The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Firewall User Authentication Overview on page 9

web-authentication (Interfaces)

Syntax

```
web-authentication {
  http;
  https;
  redirect-to-https;
}
```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family *family-name* address *address*]

Release Information Statement introduced in Junos OS Release 9.2.
Support for **https** and **redirect-to-https** introduced for high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.

Description Enable the Web authentication process for firewall user authentication.

Options

http—Enable HTTP service.

https—Enable authentication through HTTPS.

redirect-to-https—Redirect Web authentication to HTTPS.

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

Related Documentation

- *Understanding Interfaces*

web-management (System Services)

Syntax

```
web-management {
  http {
    interfaces interface-names ;
    port port;
  }
  https {
    interfaces interface-names;
    local-certificate name;
    pki-local-certificate name;
    system-generated-certificate name;
    port port;
  }
  management url management url;
  session {
    idle-timout minutes;
    session-limit number;
  }
  traceoptions {
    file {
      filename;
      files number;
      match regular-expression;
      size maximum-file-size;
      (no-world-readable | world-readable);
    }
    flag flag;
    level level;
    no-remote-trace;
  }
}
```

Hierarchy Level [edit system services]

Release Information Statement introduced in Junos OS Release 9.0.
Support for **https** introduced for SRX5400, SRX5600, and SRX5800 devices starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices starting from Junos OS Release 15.1X49-D40.

Description Configure settings for HTTP or HTTPS access. HTTP access allows management of the device using the J-Web interface. HTTPS access allows secure management of the device using the J-Web interface. With HTTPS access, communication is encrypted between your browser and the webserver for your device.

Options

control—Disable the SBC process.

- **max-threads**—Maximum simultaneous threads to handle requests.

  Range: 0 through 16

http—Configure HTTP.

- **interface [*value*]**—Interface value that accepts HTTP access.

- **port *number***—TCP port for incoming HTTP connections.

  Range: 1 through 65,535

https—Configure HTTPS.

- **interface [*value*]**—Interface value that accept HTTP access.
- **port *number***—TCP port for incoming HTTP connections.

  Range: 1 through 65,535
- **local-certificate**—X.509 certificate to use from the configuration.
- **pki-local-certificate**—X.509 certificate to use from the PKI local store.
- **system-generated-certificate**—X.509 certificate generated automatically by the system.

management url *management url*—URL path for Web management access.

session—Configure the Web-management session.

- **idle-timout *minutes***—Default timeout of Web-management sessions in minutes.
- **session-limit *number***—Maximum number of Web-management sessions to allow.

traceoptions—Set the trace options.

- **file**—Configure the trace file information.
  - *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. All files are placed in the directory **/var/log**. By default, the name of the file is the name of the process being traced.
  - **files** ***number***— Maximum number of trace files. When a trace file named **trace-file** reaches its maximum size, it is renamed **trace-file.0**, then **trace-file.1**, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

    If you specify a maximum number of files, you also must specify a maximum file size with the **size** ***maximum file-size*** option.

    Range: 2 through 1000 files

    Default: 10 files
- **match** ***regular-expression***—Refine the output to include lines that contain the regular expression.
- **size** ***maximum-file-size***—Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB).

  Range: 10 KB through 1 GB

  Default: 128 KB

  If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** ***number*** option.
- **(world-readable | no-world-readable)**— By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.
- **flag** ***flag***—Specify which tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements. You can include the following flags.
  - **all**—Trace all areas.
  - **configuration**—Trace configuration.
  - **dynamic-vpn**—Trace dynamic VPN events.
  - **init**—Trace the daemon init process.
  - **mgd**—Trace MGD requests.
  - **webauth**—Trace Web authentication requests.
- **level** ***level*** —Specify the level of debugging output.
  - **all**—Match all levels.
  - **error**—Match error conditions.

- **info**—Match informational messages.
- **notice**—Match conditions that should be handled specially.
- **verbose**—Match verbose messages.
- **warning**—Match warning messages.

- **no-remote-trace**—Disable remote tracing.

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

Related Documentation
- Firewall User Authentication Overview on page 9
- *Dynamic VPN Overview*

web-redirect

Syntax
web-redirect;

Hierarchy Level
[edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit firewall-authentication pass-through user-firewall]

Release Information
Statement introduced in Junos OS Release 8.5.
Starting with Junos OS Release 15.1X49-D70, support for user-firewall added on SRX300, SRX320, SRX340, SRX345, SRX550M, SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX Services Gateways.

Description
Optionally, redirect HTTP requests to the device's internal webserver by sending a redirect HTTP response to the client system to reconnect to the webserver for user authentication. The interface on which the client's request arrived is the interface to which the request is redirected.

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

Related Documentation
- *Understanding User Role Firewalls*

web-redirect-to-https

Syntax web-redirect-to-https;

Hierarchy Level [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* then permit firewall-authentication pass-through user-firewall]

Release Information Statement introduced on high-end SRX Series Services Gateways starting from Junos OS Release 12.1X44-D10 and on vSRX, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 Services Gateways starting from Junos OS Release 15.1X49-D40.
Starting with Junos OS Release 15.1X49-D70, support for **user-firewall** added on SRX300, SRX320, SRX340, SRX345, SRX550M, SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX Services Gateways.

Description Redirect unauthenticated HTTP requests to the internal HTTPS webserver of the device.

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

Related Documentation
- *UTM Feature Guide for Security Devices*
- Firewall User Authentication Overview on page 9

web-server (Services)

Syntax web-server *server-name*;

Hierarchy Level [edit services user-identification authentication-source aruba-clearpass user-query]

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Specify the name of the webserver configuration on the SRX Series device used for the user query integrated ClearPass authentication and enforcement function. The webserver is the ClearPass server to which the SRX Series device connects to request authentication and identity information for an individual user.

When information for the individual user is not posted to the SRX Series device by ClearPass through Web API POST request messages, the SRX Series device can request this information from the ClearPass Policy Manager (CPPM) under certain circumstances. You must enable the user query function by configuring it.

Required Privilege Level services—To view this statement in the configuration.

services-control—To add this statement to the configuration.

whitelist (Services)

Syntax whitelist [global-address-book-addresses];

Hierarchy Level [edit services ssl proxy profile *profile-name*]
[edit services ssl termination profile *profile-name*]

Release Information Statement introduced in Junos OS Release 12.1X44-D10.

Description Specify the addresses exempted from the SSL proxy. This statement is supported on the SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options
- *whitelist-address*—Specify address from the global address book.

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

Related Documentation
- Configuring SSL Proxy on page 23
- Firewall User Authentication Overview on page 9

wins-server (Access)

Syntax wins-server *address*

Hierarchy Level [edit access address-assignment pool <name> family (inet | inet6) xauth-attributes]

Release Information Statement introduced in Release 10.4 of Junos OS.

Description Specify the wins-server IP address.

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

Related Documentation
- *Junos OS Security Configuration Guide*

CHAPTER 26

Operational Commands

- clear network-access requests pending
- clear network-access requests statistics
- clear network-access securid-node-secret-file
- clear security firewall-authentication history
- clear security firewall-authentication history address
- clear security firewall-authentication history identifier
- clear security firewall-authentication users
- clear security firewall-authentication users address
- clear security firewall-authentication users identifier
- clear security user-identification local-authentication-table
- clear services user-identification active-directory-access
- clear services user-identification authentication-table
- request services user-identification active-directory-access active-directory-authentication-table delete
- request services user-identification active-directory-access domain-controller
- request services user-identification active-directory-access ip-user-probe
- request services user-identification authentication-source aruba-clearpass user-query
- request services user-identification authentication-table delete
- show network-access requests pending
- show network-access requests statistics
- show network-access securid-node-secret-file
- show security firewall-authentication history
- show security firewall-authentication history address
- show security firewall-authentication history identifier
- show security firewall-authentication users
- show security firewall-authentication users address
- show security firewall-authentication users identifier
- show security policies

- show service user-identification authentication-source aruba-clearpass user-query counters
- show service user-identification authentication-source aruba-clearpass user-query status
- show services unified-access-control authentication-table
- show services user-identification authentication-table
- show services user-identification active-directory-access user-group-mapping
- show services user-identification device-information table
- show services unified-access-control counters
- show services unified-access-control policies
- show services unified-access-control roles
- show services unified-access-control status
- show services user-identification active-directory-access active-directory-authentication-table
- show services user-identification active-directory-access domain-controller status
- show services user-identification active-directory-access statistics
- show services user-identification active-directory-access user-group-mapping

clear network-access requests pending

Syntax clear network-access requests pending
<index *index-number* >

Release Information Command introduced in Release 8.5 of Junos OS.

Description Clear or cancel all pending authentication requests.

Options
- none—Clear all network access requests pending.
- **index** ***index-number*** —Clear the specified authentication request. To display index numbers, use the **show network-access requests pending** command.

Required Privilege Level clear

Related Documentation
- show network-access requests pending on page 469

List of Sample Output clear network-access requests pending on page 445

Sample Output

The following example displays the network access requests that are pending, clears the requests, and displays the results of the clear operation:

clear network-access requests pending

```
user@host> show network-access requests pending
Information about pending authentication entries
  Total pending authentication requests: 2
Index User                  Status
1     Sun                   Processing
2     Sam                   Processed

user@host> clear network-access requests pending
user@host> show network-access requests pending
Information about pending authentication entries
  Total pending authentication requests: 2
Index User                  Status
1     Sun                   Cancelled by Admin
2     Sam                   Cancelled by Admin
```

clear network-access requests statistics

Syntax clear network-access requests statistics

Release Information Command introduced in Release 8.5 of Junos OS.

Description Clear general authentication statistics for the configured authentication type.

Required Privilege Level clear

Related Documentation
- *authentication-order (Access Profile)*
- show network-access requests statistics on page 471

Output Fields This command produces no output.

clear network-access securid-node-secret-file

Syntax clear network-access securid-node-secret-file

Release Information Command introduced in Junos OS Release 9.1.

Description Delete the node secret file for the SecurID authentication type.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9
- configuration-file on page 314
- securid-server on page 390
- show network-access securid-node-secret-file on page 472

List of Sample Output clear network-access securid-node-secret-file on page 447

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear network-access securid-node-secret-file

```
user@host> clear network-access securid-node-secret-file
```

clear security firewall-authentication history

Syntax clear security firewall-authentication history
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OSRelease 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear all firewall authentication history information.

Options

- **node**—(Optional) For chassis cluster configurations, clear all firewall authentication history on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output

clear security firewall-authentication history on page 448
clear security firewall-authentication history node 1 on page 448

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication history

```
user@host> clear security firewall-authentication history
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication history node 1

```
user@host> clear security firewall-authentication history node 1
node1:
--------------------------------------------------------------------------
```

clear security firewall-authentication history address

Syntax clear security firewall-authentication history address *address*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear firewall authentication history for this source IP address.

Options

- **address** *address* —Source IP address for which to clear firewall authentication history.
- none—Clear all firewall authentication history for this address.
- **node**—(Optional) For chassis cluster configurations, clear firewall authentication history for this address on a specific node.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output

clear security firewall-authentication history address 10.0.0.1 on page 449
clear security firewall-authentication history address 192.0.2.2 node 1 on page 449

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication history address 10.0.0.1

```
user@host> clear security firewall-authentication history address 10.0.0.1
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication history address 192.0.2.2 node 1

```
user@host> clear security firewall-authentication history address 192.0.2.2 node 1
node1:
--------------------------------------------------------------------------
```

clear security firewall-authentication history identifier

Syntax clear security firewall-authentication history identifier *identifier*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear firewall authentication history information for the authentication with this identifier.

Options

- **identifier** *identifier* —Identification number of the authentication for which to clear authentication history.
- none—Clear all firewall authentication history information for the authentication with this identifier.
- **node**—(Optional) For chassis cluster configurations, clear firewall authentication history on a specific node for the authentication with this identifier.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output
clear security firewall-authentication history identifier 2 on page 450
clear security firewall-authentication history identifier 2 node 1 on page 450

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication history identifier 2

```
user@host> clear security firewall-authentication history identifier 2
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication history identifier 2 node 1

```
user@host> clear security firewall-authentication history identifier 2 node 1
node1:
--------------------------------------------------------------------------
```

clear security firewall-authentication users

Syntax clear security firewall-authentication users
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear firewall authentication tables for all users.

Options

- **node**—(Optional) For chassis cluster configurations, clear firewall authentication details for all users on a specific node.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9
- show security firewall-authentication users on page 481

List of Sample Output

clear security firewall-authentication users on page 452
clear security firewall-authentication users node 1 on page 452

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication users

```
user@host> clear security firewall-authentication users node 1
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication users node 1

```
user@host> clear security firewall-authentication users node 1
node1:
--------------------------------------------------------------------------
```

clear security firewall-authentication users address

Syntax clear security firewall-authentication users address *address*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear information about the users at the specified IP address that are currently authenticated.

Options

- **address** *address* —IP address for which to clear user firewall authentication information.
- none—Clear all the firewall authentication information for users at this IP address.
- **node**—(Optional) For chassis cluster configurations, clear user firewall authentication entries on a specific node.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output

clear security firewall-authentication users address 198.51.100.11 on page 453
clear security firewall-authentication users address 198.51.100.11 node 1 on page 453

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication users address 198.51.100.11

```
user@host> clear security firewall-authentication users address 198.51.100.11
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication users address 198.51.100.11 node 1

```
user@host> clear security firewall-authentication users address 198.51.100.11 node 1
node1:
--------------------------------------------------------------------------
```

clear security firewall-authentication users identifier

Syntax clear security firewall-authentication users identifier *identifier*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Clear firewall authentication details about the user with this identification number.

Options

- none—Identification number of the user for which to clear authentication details.
- **node**—(Optional) For chassis cluster configurations, clear the firewall authentication details on a specific node (device) in the cluster for the user with this identification number.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level clear

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output
clear security firewall-authentication users identifier 2 on page 454
clear security firewall-authentication users identifier 2 node 1 on page 454

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear security firewall-authentication users identifier 2

```
user@host> clear security firewall-authentication users identifier 2
node0:
--------------------------------------------------------------------------
node1:
--------------------------------------------------------------------------
```

Sample Output

clear security firewall-authentication users identifier 2 node 1

```
user@host> clear security firewall-authentication users identifier 2 node 1
node1:
--------------------------------------------------------------------------
```

clear security user-identification local-authentication-table

Syntax clear security user-identification local-authentication-table

Release Information Command introduced in Junos OS Release 12.1.

Description This command removes all entries from the local authentication table.

Required Privilege Level clear

Related Documentation
- *Understanding AppSecure Services*
- Firewall User Authentication Overview on page 9

List of Sample Output clear security user-identification local-authentication-table on page 455

Output Fields When you enter this command, all entries are cleared from the local authentication table.

Sample Output

clear security user-identification local-authentication-table

```
user@host> clear security user-identification local-authentication-table
user@host> show security user-identification local-authentication-table all
Total entries: 0
```

clear services user-identification active-directory-access

Syntax

```
clear services user-identification active-directory-access
  (active-directory-authentication-table |
statistics (ip-user-mapping | ip-user-probe | user-group-mapping))
```

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Delete entries from the Active Directory authentication table or statistics related to integrated user firewall mappings.

Options

- **active-directory-authentication-table**—Remove all entries from the Active Directory authentication table.
- **statistics**—Remove the specified type of statistics:
  - **ip-user-mapping**—IP address-to-user mappings
  - **ip-user-probe**—PC probe statistics
  - **user-group-mapping**—User-to-group mappings

Required Privilege Level clear

Related Documentation

- ip-user-mapping on page 350
- request services user-identification active-directory-access ip-user-probe on page 460
- show services user-identification active-directory-access statistics on page 529
- show services user-identification active-directory-access user-group-mapping on page 510
- user-group-mapping on page 428
- user-identification (Services) on page 430

Output Fields This command produces no output.

clear services user-identification authentication-table

Syntax clear services user-identification authentication-table authentication-source *authentication-source* (all | active-directory | aruba-clearpass)

Release Information Statement introduced in Junos OS Release 12.3X48-D30.

Description Clear the contents of the ClearPass authentication table. The ClearPass authentication table, which is created by the SRX Series device on the Packet Forwarding Engine, is populated with user authentication and identity information received from Aruba ClearPass. Aruba ClearPass is the authentication source for the integrated ClearPass feature. You must aruba-clearpass as the authentication source.

Options *authentication-source*—For the SRX Series integrated ClearPass feature, you must specify aruba-clearpass to indicate that ClearPass is the authentication source and that the authentication table relies on user information from the ClearPass Policy Manager.

Additional Information The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level clear

List of Sample Output clear services user-identification authentication-table authentication-source on page 457

Output Fields If there are no entries in the ClearPass authentication table, the following warning message is displayed after you enter the clear command.

There is no authentication-table entry.

If there are entries in the ClearPass authentication table, no messages are displayed after you enter the **clear** command.

Sample Output

clear services user-identification authentication-table authentication-source

```
user@host> clear services user-identification authentication-table authentication-source
aruba-clearpass
warning: "There is no authentication-table entry."
```

request services user-identification active-directory-access active-directory-authentication-table delete

Syntax

request services user-identification active-directory-access
active-directory-authentication-table delete
(domain *name* | ip-address *ip-address* | group *group-name* <domain *name*> | user *name*
<domain *name*>

Release Information

Command introduced in Junos OS Release 12.1X47-D10.

Description

Delete entries from the active directory authentication table by domain, address, group, or user. This command provides the network administrator with flexibility and control over the table entries beyond what is automatically added to or deleted from the table. For example, if a person leaves the company, the corresponding username can be deleted; after a department reorganization, a group can be deleted.

Options

- **domain** *name*—Delete the entries from the authentication table for the specified domain.
- **ip-address** *ip-address*—Delete the entry from the authentication table for the specified IP address.
- **group** *group-name*—Delete the entries from the authentication table for the specified group.
  - **domain** *name*—Delete the group only from the specified domain.
- **user** *name*—Delete the entries from the authentication table for the specified username.
  - **domain** *name*—Delete the user only from the specified domain.

Required Privilege Level

maintenance

Related Documentation

- show services user-identification active-directory-access active-directory-authentication-table on page 522
- user-identification (Services) on page 430
- Understanding Active Directory Authentication Tables on page 144

Output Fields

This command produces no output.

request services user-identification active-directory-access domain-controller

Syntax request services user-identification active-directory-access domain-controller discovery domain *name*

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Discover and display the name and address of all domain controllers in the specified domain.

Options

- domain *name*—Name of the domain for which to get and display domain controller names and addresses.

Required Privilege Level maintenance

Related Documentation

- active-directory-access on page 294
- show services user-identification active-directory-access domain-controller status on page 526
- user-identification (Services) on page 430

List of Sample Output request services user-identification active-directory-access domain-controller discovery domain <domain-name> on page 459

Output Fields This command displays the discovered domain controllers.

Sample Output

request services user-identification active-directory-access domain-controller discovery domain <domain-name>

```
user@host> request services user-identification active-directory-access domain-controller
discovery domain example.net
Domain: example.net
 Domain controller: example-dc.example.net
   Address: 192.0.2.2
```

request services user-identification active-directory-access ip-user-probe

Syntax request services user-identification active-directory-access ip-user-probe address *ip-address* <domain *name*>

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Probe the PC at the specified IP address to get an authentication entry, which is used for the integrated user firewall feature. You can display the authentication table to see the results. If the probe succeeded, there will be a valid authentication entry. If the probe failed, there will be an invalid authentication entry.

Options

- address *ip-address*—Probe the PC at this IP address.

  domain *name*—Probe the IP address in the specified domain.

Required Privilege Level maintenance

Related Documentation

- clear services user-identification active-directory-access on page 456
- show services user-identification active-directory-access active-directory-authentication-table on page 522
- show services user-identification active-directory-access statistics on page 529
- user-identification (Services) on page 430

List of Sample Output show services user-identification active-directory-access active-directory-authentication-table address <ip-address> on page 460

Output Fields The following command displays the results of the IP address probe:

Sample Output

show services user-identification active-directory-access active-directory-authentication-table address <ip-address>

```
user@host> show services user-identification active-directory-access
active-directory-authentication-table address 192.0.2.3
Domain: example.net
 Source-ip: 192.0.2.3
  Username: user1
  Groups:group1
  State: Valid
  Source: wmic
  Access start date: 2014-03-10
  Access start time: 13:59:56
  Age time: 1437
```

request services user-identification authentication-source aruba-clearpass user-query

Syntax request services user-identification authentication-source *authentication-source* user-query address *ip-address*

Release Information Command introduced in Junos OS Release 12.3X48-D30.

Description Manually send to the ClearPass website a request for user authentication and identity information for an individual user. The command specifies the IP address of the user's device to identify the user whose information you want to obtain. If the user query command executes successfully, an entry for the user (IP address) has been created in the ClearPass authentication table, and no output is displayed.

The user query function is part of the SRX Series integrated ClearPass authentication and enforcement feature. The user query function, if configured, allows the SRX Series device to send requests for individual user information. This command also allows you to manually send requests. Normally administrators send query requests manually to troubleshoot issues.

The user query function supplements use of the Web API function. The SRX Series device exposes to ClearPass a Web API that ClearPass uses to send POST request messages to the SRX Series device. These messages contain user authentication and identity information.

Options *ip-address*—The IP address of the user's device for whom you are manually requesting authentication information.

Required Privilege Level maintenance

List of Sample Output request services user-identification authentication-source authentication-source user-query address ip-address on page 461

Sample Output

request services user-identification authentication-source authentication-source user-query address ip-address

```
user@host> request services user-identification authentication-source aruba-clearpass user-query
address 40.0.0.1
```

request services user-identification authentication-table delete

Syntax request services user-identification authentication-table delete (ip-address *ip-address* | authentication-source (all | active-directory | *authentication-source* (domain *domain-name* | group *group-name* |user *user-name*))

Release Information Command introduced in Junos OS Release 12.3X48-D30.

Description Delete entries from the ClearPass authentication table based on the IP address of the user's device, or on the authentication source and the name of a domain, a group, or a user. When only the authentication source is specified, the entire ClearPass authentication table is deleted. For the integrated ClearPass authentication and enforcement feature, the authentication source is always aruba-clearpass.

Options

ip-address—Deletes a user authentication entry from the ClearPass authentication table, and the Active Directory (AD) table, based on the IP address of the user's device.

authentication-source —Deletes user entries from the ClearPass authentication table. In the CLI, ClearPass as the authentication source is referred to by the value aruba-clearpass as is the ClearPass authentication table. To identify the user entries to be deleted, you specify a domain, a group, or a username.

domain-name—Deletes from the ClearPass authentication table user entries for users who belong to the specified domain.

group *group-name*—Deletes the entry entry from the ClearPass authentication table for users who belong to the group, regardless of whether they belong to other groups.

user *user-name*—Deletes the entry for the specified user from the ClearPass authentication table.

Required Privilege Level maintenance

List of Sample Output

request services user-identification authentication-table delete ip-address on page 463
request services user-identification authentication-table delete authentication-source aruba-clearpass domain on page 463
request services user-identification authentication-table delete authentication-source aruba-clearpass group on page 464
request services user-identification authentication-table delete authentication-source aruba-clearpass on page 466

Output Fields The following examples cover how to delete various user entries from the ClearPass authentication table based on the specified parameter. It also shows how to check to ensure that the user entries were deleted successfully.

Sample Output

request services user-identification authentication-table delete ip-address

```
The following command deletes the entry for the user whose device IP address is
specified.
user@host> request services user-identification authentication-table delete ip-address 50.0.0.1
```

Before you delete the entry:

To ensure that the entry exists in the ClearPass authentication table, use the following command to display the entry for the user. Note that the ClearPass authentication table includes the user entry with the IP address 50.0.0.1.

```
user@host> show services user-identification authentication-table ip-address 50.0.0.1
Domain: GLOBAL
  Source-ip: 50.0.0.1
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2015-12-14
    Access start time: 17:07:23
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
```

After you delete the user entry associated with the IP address, enter the command again to verify that the entry has been deleted.

```
user@host> show services user-identification authentication-table ip-address 50.0.0.1
warning: “This IP address isn’t in authentication table.”
```

request services user-identification authentication-table delete authentication-source aruba-clearpass domain

```
The following command deletes the specified domain.
user@host> request services user-identification authentication-table delete authentication-source
domain global
```

Before you delete the domain contents from the ClearPass authentication table, use the following command to display the domain information to ensure that it exists. Note that the ClearPass authentication table includes the global domain.

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass domain global extensive
Domain: GLOBAL
Total entries: 6
  Source-ip: 10.0.0.1
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 20.0.0.1
    Username: abew1
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
```

```
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
  Source-ip: 30.0.0.1
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 40.0.0.1
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
  Source-ip: 50.0.0.1
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 50.0.0.2
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

After you delete the domain, use the command again to verify that the domain and its user members was deleted.

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass domain global
warning: "There is no related auth entry in authentication-table."
```

request services user-identification authentication-table delete authentication-source aruba-clearpass group

The following command deletes the entries for any users who belong to the group posture-healthy.

```
user@host> request services user-identification authentication-table delete authentication-source
aruba-clearpass group posture-healthy
```

Before you delete the group contents from the ClearPass authentication table, use the following command to display it to ensure that the group is used in some user entries. Notice that the appropriate user entries contain the posture-healthy group.

```
Domain: GLOBAL
Total entries: 6
  Source-ip: 10.0.0.1
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 20.0.0.1
    Username: abew1
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
  Source-ip: 30.0.0.1
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 40.0.0.1
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
  Source-ip: 50.0.0.1
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 50.0.0.2
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
```

```
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

Enter the **show services user-identification authentication-table authentication-source aruba-clearpass group posture-healthy** to display the entries for the users who belong to the group posture-healthy.

Notice that the group name does not show up in the column for groups referenced by policy because it is not one. Notice, too, that the output contains information for only those users who belong to the group. It does not include an entry for the user abew1, who does not belong to the group.

```
Domain: GLOBAL
Source IP       Username       groups(Ref by policy)          state
10.0.0.1        viki2          accounting-grp-and-company-dev Valid
30.0.0.1        jxchan         marketing-access-for-pcs-limit Valid
40.0.0.1        lchen1         corporate-limited              Valid
50.0.0.1        guest1                                        Valid
50.0.0.2        guest2                                        Valid
```

After you delete the group, use the command again to verify that it has been deleted.

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass group posture-healthy
warning: "There is no related auth entry in authentication-table."
```

For further verification, you can use the following command to check the entry for one of the users who belonged to the group:

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass user viki2
warning: "There is no related auth entry in authentication-table."
```

request services user-identification authentication-table delete authentication-source aruba-clearpass

The following command deletes the ClearPass authentication table (aruba-clearpass).

```
user@host> request services user-identification authentication-table delete authentication-source
aruba-clearpass
```

Before you delete the ClearPass authentication table, use the following command to display it to ensure that the table exists.

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass
Domain: GLOBAL
Total entries: 6
  Source-ip: 10.0.0.1
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
```

```
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 20.0.0.1
    Username: abew1
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
  Source-ip: 30.0.0.1
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 40.0.0.1
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
  Source-ip: 50.0.0.1
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 50.0.0.2
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

To verify that you deleted the authentication table successfully, enter the command again:

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass

warning: "There is no authentication-table entry."
```

show network-access requests pending

Syntax
show network-access requests pending
<detail>
<index ***number*** **>**

Release Information Command introduced in Release 8.5 of Junos OS.

Description Display the status of pending authentication requests.

Options
- none—Show pending authentication requests.
- show network-access requests pendingshow network-access requests pending**detail**—Display detailed information about all pending requests.
- **index** ***number***—(Optional) Display detailed information about the request specified by this index number. Use the command without options to obtain a list of requests and index numbers.

Required Privilege Level view

Related Documentation
- clear network-access requests pending on page 445

List of Sample Output
show network-access requests pending on page 470
show network-access requests pending detail on page 470
show network-access requests pending index 1 on page 470

Output Fields Table 28 on page 469 lists the output fields for the show network-access requests pending command. Output fields are listed in the approximate order in which they appear.

Table 28: show network-access requests pending Output Fields

Field Name	Field Description
Index	Internal number identifying the pending request. Use this number to obtain more information on the record.
User	Originator of authentication request.
Status	The pending requests are requests and responses that are not yet sent back to the respective clients. The pending requests can be in one of the following states: • **Processing**: This request is being processed by the device. The authentication process has started but is not complete. • **Waiting on Auth Server**: The request is sent to an external authentication server, and the device is waiting for the response. • **Processed:** This request has completed authentication (success or failure). The results are not yet forwarded back to the client. • **Request cancelled by Admin**: This request was cancelled by the Admin. The reply with cancel code is not yet sent back to the client.

Table 28: show network-access requests pending Output Fields *(continued)*

Field Name	Field Description
Profile	The profile determines how the user is authenticated. Local clients defined with the statement **access profile client** are authenticated with the password authentication. Clients configured external to the device, on a RADIUS or LDAP server are authenticated with RADIUS or LDAP authentication.

Sample Output

show network-access requests pending

```
user@host> show network-access requests pending
Information about pending authentication entries
  Total pending authentication requests: 2
Index User                 Status
1     Sun                  Processing
2     Sam                  Processed
```

Sample Output

show network-access requests pending detail

```
user@host> show network-access requests pending detail
Information about pending authentication entries
  Total pending authentication requests: 2
Index: 1  User: Sun
  Status: Processing
  Profile: Sunnyvale-firewall-users
Index: 2  User: Sam
  Status: Processed
  Profile: Westford-profile
```

Sample Output

show network-access requests pending index 1

```
user@host> show network-access requests pending index 1
Index: 1  User: Sun
  Status: Processing
  Profile: Sunnyvale-firewall-users
```

show network-access requests statistics

Syntax **show network-access requests statistics**

Release Information Command modified in Release 9.1 of Junos OS.

Description Display authentication statistics for the configured authentication type.

Required Privilege Level view

Related Documentation
- clear network-access requests statistics on page 446

Output Fields Table 29 on page 471 lists the output fields for the network-access requests statistics command. Output fields are listed in the approximate order in which they appear.

Table 29: show network-access requests statistics Output Fields

Field Name	Field Description
Total requests received	Total number of authentication requests that the device received from clients.
Total responses sent	Total number of authentication responses that the device sent to the clients.
Success responses	Total number of clients that authenticated successfully.
Failure responses	Total number of clients that failed to authenticate.

show network-access requests statistics

```
user@host> show network-access requests statistics
General authentication statistics
  Total requests received: 100
  Total responses sent: 70
Radius authentication statistics
  Total requests received: 40
  Success responses: 20
  Failure responses: 20
LDAP authentication statistics
  Total requests received: 30
  Success responses: 15
  Failure responses: 15
Local authentication statistics
  Total requests received: 5
  Success responses: 2
  Failure responses: 3
Securid authentication statistics
  Total requests received: 15
  Success responses: 3
  Failure responses: 12
```

show network-access securid-node-secret-file

Syntax **show network-access securid-node-secret-file**

Release Information Command introduced in Release 9.1 of Junos OS.

Description Display the path to the node secret file for the SecurID authentication type.

Required Privilege Level view

Related Documentation

- configuration-file on page 314
- securid-server on page 390
- clear network-access securid-node-secret-file on page 447

List of Sample Output show network-access securid-node-secret-file on page 472

Output Fields Table 30 on page 472 lists the output fields for the network-access securid-node-secret-file command. Output fields are listed in the approximate order in which they appear.

Table 30: show network-access securid-node-secret-file Output Fields

Field Name	Field Description
SecurID Server	Name of the SecurID authentication server.
Node Secret File	Path to the node secret file.

Sample Output

show network-access securid-node-secret-file

```
user@host> show network-access securid-node-secret-file
SecurID server node secret file:
SecurID Server       Node Secret File
ace-server1          /var/db/securid/ace-server1/node-secret
```

show security firewall-authentication history

Syntax **show security firewall-authentication history**
<node (***node-id*** | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Display security firewall authentication history information.

Options
- none—Display history of firewall authentication information.
- **node**—(Optional) For chassis cluster configurations, display all firewall authentication history on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level view

Related Documentation
- *Understanding Logical System Firewall Authentication*
- Firewall User Authentication Overview on page 9

List of Sample Output
show security firewall-authentication history on page 474
show security firewall-authentication history node all on page 474

Output Fields Table 31 on page 473 lists the output fields for the **show security firewall-authentication history** command. Output fields are listed in the approximate order in which they appear.

Table 31: show security firewall-authentication history Output Fields

Field Name	Field Description
Authentications	Number of authentications.
Id	Identification number.
Source IP	IP address of the authentication source.
Date	Authentication date.
Time	Authentication time.
Duration	Authentication duration.
Status	Authentication status success or failure.

Table 31: show security firewall-authentication history Output Fields *(continued)*

Field Name	Field Description
User	Name of the user.

Sample Output

show security firewall-authentication history

```
user@host> show security firewall-authentication history
History of firewall authentication data:
  Authentications: 1
          Id Source Ip       Date       Time     Duration   Status   User
           1 203.0.113.1     2007-04-03 11:43:06 00:00:45   Success  hello
```

Sample Output

show security firewall-authentication history node all

```
user@host> show security firewall-authentication history node all
node0:
--------------------------------------------------------------------------
History of firewall authentication data:
Authentications: 2
Id Source Ip       Date       Time     Duration   Status   User
1 203.0.113.1      2008-01-04 12:00:10 0:05:49    Success  local1
2 203.0.113.1      2008-01-04 14:36:52 0:01:03    Success  local1
node1:
--------------------------------------------------------------------------
History of firewall authentication data:
  Authentications: 1
          Id Source Ip       Date       Time     Duration   Status   User
           203.0.113.1       2008-01-04 14:59:43 1193046:06: Success local1
```

show security firewall-authentication history address

Syntax **show security firewall-authentication history address** *ip-address* <node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Display security firewall authentication history for this source IP address.

Options

- **address** *ip-address* —IP address of the authentication source.
- none—Display all firewall authentication history for this address.
- **node**—(Optional) For chassis cluster configurations, display firewall authentication history for this address on a specific node.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level view

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output show security firewall-authentication history address 198.51.100.17 on page 476
show security firewall-authentication history address 198.51.100.17 node local on page 476

Output Fields Table 32 on page 475 lists the output fields for the **show security firewall-authentication history address** command. Output fields are listed in the approximate order in which they appear.

Table 32: show security firewall-authentication history address Output Fields

Field Name	Field Description
Username	User ID.
Source IP	IP address of the authentication source.
Authentication state	Status of authentication (success or failure).
Authentication method	Path chosen for authentication.
Access start date	Date when user authenticated.

Table 32: show security firewall-authentication history address Output Fields *(continued)*

Field Name	Field Description
Access start time	Time when user authenticated.
Duration of user access	Time duration of the accessing firewall.
Policy name	Name of the policy.
Source zone	User traffic received from the zone.
Destination zone	User traffic destined to the zone.
Access profile	Name of profile used for authentication.
Bytes sent by this user	Number of bytes sent by the user.
Bytes received by this user	Number of bytes received by the user.

Sample Output

show security firewall-authentication history address 198.51.100.17

```
user@host> show security firewall-authentication history address 198.51.100.17
Username: u1
Source IP: 198.51.100.17
Authentication state: Success
Authentication method: Pass-through using HTTP
Access start date: 2007-09-12
Access start time: 15:33:29
Duration of user access: 0:00:48
Policy name: Z1-Z2
Source zone: Z1
Destination zone: Z2
Access profile: profile-local
Bytes sent by this user: 0
Bytes received by this user: 449
```

Sample Output

show security firewall-authentication history address 198.51.100.17 node local

```
user@host> show security firewall-authentication history address 198.51.100.17 node local
node0:
--------------------------------------------------------------------------
Username: local1
Source IP: 198.51.100.17
Authentication state: Success
Authentication method: Pass-through using Telnet
Access start date: 2008-01-04
Access start time: 12:00:10
Duration of user access: 0:05:49
Policy name: POL1
Source zone: z1
Destination zone: z2
```

```
Access profile: p1
Bytes sent by this user: 0
Bytes received by this user: 0
Username: local1
Source IP: 198.51.100.17
Authentication state: Success
Authentication method: Pass-through using Telnet
Access start date: 2008-01-04
Access start time: 14:36:52
Duration of user access: 0:01:03
Policy name: POL1
Source zone: z1
Destination zone: z2
Access profile: p1
Bytes sent by this user: 2178
Bytes received by this user: 4172
```

show security firewall-authentication history identifier

Syntax **show security firewall-authentication history identifier** ***identifier***
<node (***node-id*** | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Display security firewall authentication history information for the authentication with this identifier.

Options

- **identifier** ***identifier***—Identifying number of the authentication process.
- none—Display all firewall authentication history information for the authentication with this identifier.
- **node**—(Optional) For chassis cluster configurations, display firewall authentication history on a specific node for the authentication with this identifier.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level view

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output
show security firewall-authentication history identifier 1 on page 479
show security firewall-authentication identifier 1 node primary on page 479

Output Fields Table 33 on page 478 lists the output fields for the **show security firewall-authentication history identifier** command. Output fields are listed in the approximate order in which they appear.

Table 33: show security firewall-authentication history identifier Output Fields

Field Name	Field Description
Username	User ID.
Source IP	IP address of the authentication source.
Authentication state	Status of authentication (success or failure).
Authentication method	Path chosen for authentication.
Access start date	Date when user authenticated.

Table 33: show security firewall-authentication history identifier Output Fields *(continued)*

Field Name	Field Description
Access start time	Time when user authenticated.
Duration of user access	Time duration of the accessing firewall.
Policy index	Identification number of the policy.
Policy name	Name of the policy.
Source zone	User traffic received from the zone.
Destination zone	User traffic destined to the zone.
Access profile	Name of profile used for authentication.
Bytes sent by this user	Number of bytes sent by the user.
Bytes received by this user	Number of bytes received by the user.
Client-groups	Name of the client group.

Sample Output

show security firewall-authentication history identifier 1

```
user@host> show security firewall-authentication history identifier 1
Username: hello
Source IP: 192.0.2.5
Authentication state: Success
Authentication method: Pass-through using Telnet
Access start date: 2007-04-03
Access start time: 11:43:06
Duration of user access: 00:00:45
Policy index: 4
Source zone: z2
Destination zone: z1
Access profile: profile1
Bytes sent by this user: 0
Bytes received by this user: 1050
Client-groups: Sunnyvale Bangalore
```

Sample Output

show security firewall-authentication identifier 1 node primary

```
user@host> show security firewall-authentication history identifier 1 node primary
node0:
--------------------------------------------------------------------------
Username: local1
Source IP: 192.0.2.5
Authentication state: Success
Authentication method: Pass-through using Telnet
```

```
Access start date: 2008-01-04
Access start time: 12:00:10
Duration of user access: 0:05:49
Policy name: POL1
Source zone: z1
Destination zone: z2
Access profile: p1
Bytes sent by this user: 0
Bytes received by this user: 0
```

show security firewall-authentication users

Syntax
show security firewall-authentication users
<node (*node-id* | all | local | primary)>

Release Information
Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description
Display firewall authentication details about all users.

Options
- none—Display details about all firewall authentication users.
- **node**—(Optional) For chassis cluster configurations, display firewall authentication details for all users on a specific node.
  - *node-id*—Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level
view

Related Documentation
- Firewall User Authentication Overview on page 9

List of Sample Output
show security firewall-authentication users on page 482
show security firewall-authentication users node 0 on page 482
show security firewall-authentication users node all on page 482

Output Fields
Table 34 on page 481 lists the output fields for the **show security firewall-authentication users** command. Output fields are listed in the approximate order in which they appear.

Table 34: show security firewall-authentication users Output Fields

Field Name	Field Description
Total users in table	Gives count of how many entries/users the command will display.
Id	Identification number.
Source IP	IP address of the authentication source.
Src zone	User traffic received from the zone.
Dst zone	User traffic destined to the zone.
Profile	Name of profile used for authentication.
Age	Idle timeout for the user.

Table 34: show security firewall-authentication users Output Fields *(continued)*

Field Name	Field Description
Status	Authentication status success or failure.
User	Name of the user.

Sample Output

show security firewall-authentication users

```
user@host> show security firewall-authentication users
Firewall authentication data:
  Total users in table: 1
          Id Source Ip       Src zone Dst zone Profile    Age Status   User
           1 192.0.2.5/24         z1       z2       p1       0 Success  local1
```

Sample Output

show security firewall-authentication users node 0

```
user@host> show security firewall-authentication users node 0
node0:
--------------------------------------------------------------------------
Firewall authentication data:
  Total users in table: 1
          Id Source Ip       Src zone Dst zone Profile    Age Status   User

           3 192.0.2.5/24        z1       z2       p1           1 Success  local1
```

Sample Output

show security firewall-authentication users node all

```
user@host> show security firewall-authentication users node all
node0:
--------------------------------------------------------------------------
Firewall authentication data:
  Total users in table: 1
          Id Source Ip       Src zone Dst zone Profile    Age Status   User
           3 192.0.2.5        z1       z2       p1           1 Success  local1

node1:
--------------------------------------------------------------------------
Firewall authentication data:
  Total users in table: 1
          Id Source Ip       Src zone Dst zone Profile    Age Status   User
           2 192.0.2.5       z1       z2       p1           1 Success  local1
```

show security firewall-authentication users address

Syntax **show security firewall-authentication users address** *ip-address*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Display information about the users at the specified IP address that are currently authenticated.

Options

- **address** *ip-address*—IP address of the authentication source.
- none—Display all the firewall authentication information for users at this IP address.
- **node**—(Optional) For chassis cluster configurations, display user firewall authentication entries on a specific node.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level view

Related Documentation

- *Understanding User Role Firewalls*

List of Sample Output

show security firewall-authentication users address 192.0.2.9 on page 484
show security firewall-authentication users address 192.0.2.9 node local on page 484
show security firewall-authentication users address 198.51.100.29 on page 485

Output Fields Table 35 on page 483 lists the output fields for the **show security firewall-authentication users address** command. Output fields are listed in the approximate order in which they appear.

Table 35: show security firewall-authentication users address Output Fields

Field Name	Field Description
Username	User ID.
Source IP	IP address of the authentication source.
Authentication state	Status of authentication (success or failure).
Authentication method	Path chosen for authentication.
Access time remaining	Duration for which the connection exists.

Table 35: show security firewall-authentication users address Output Fields *(continued)*

Field Name	Field Description
Lsys	The logical system where the traffic was received.
Source zone	User traffic received from the zone.
Destination zone	User traffic destined to the zone.
Policy index	Identification number of the policy.
Policy name	Name of the policy.
Access profile	Name of profile used for authentication.
Interface Name	Name of the interface.
Bytes sent by this user	Number of bytes sent by the user.
Bytes received by this user	Number of bytes received by the user.
Client-groups	Name of the client group.

Sample Output

show security firewall-authentication users address 192.0.2.9

```
user@host>show security firewall-authentication users address 192.0.2.9
Username: hello
Source IP: 192.0.2.9
Authentication state: Success
Authentication method: Pass-through using Telnet
Access time remaining: 0
Source zone: z2
Destination zone: z1
Policy index: 5
Access profile: profile1
Interface Name: ge-0/0/2.0
Bytes sent by this user: 0
Bytes received by this user: 0
Client-groups: my-group1-example, my-group2-example
```

Sample Output

show security firewall-authentication users address 192.0.2.9 node local

```
user@host> show security firewall-authentication users address 192.0.2.9 node local
node0:
--------------------------------------------------------------------------
Username: local1
Source IP: 192.0.2.9
Authentication state: Success
Authentication method: Pass-through using Telnet
Age: 2
```

```
Access time remaining: 4
Source zone: z1
Destination zone: z2
Policy name: POL1
Access profile: p1
Interface Name: reth1.0
Bytes sent by this user: 614
Bytes received by this user: 1880
```

show security firewall-authentication users address 198.51.100.29

```
user@host> show security firewall-authentication users address 198.51.100.29
Username: hello
Source IP: 198.51.100.29/24
Authentication state: Success
Authentication method: User-firewall
Age: 0
Access time remaining: 10
Lsys: root-logical-system
Source zone: N/A
Destination zone: N/A
Access profile: test
```

show security firewall-authentication users identifier

Syntax **show security firewall-authentication users identifier** *identifier*
<node (*node-id* | all | local | primary)>

Release Information Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description Display firewall authentication details about the user with this identification number.

Options

- **identifier** *identifier*—Identification number of the user for which to display authentication details.
- **node**—(Optional) For chassis cluster configurations, display the firewall authentication details security firewall authentication entry on a specific node (device) in the cluster for the user with this identification number.
  - *node-id* —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level view

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output
show security firewall-authentication users identifier 3 on page 487
show security firewall-authentication users identifier 3 node primary on page 487

Output Fields Table 36 on page 486 lists the output fields for the **show security firewall-authentication users identifier** command. Output fields are listed in the approximate order in which they appear.

Table 36: show security firewall-authentication users identifier Output Fields

Field Name	Field Description
Username	User ID.
Source IP	IP address of the authentication source.
Authentication state	Status of authentication (success or failure).
Authentication method	Path chosen for authentication.
Age	Idle timeout for the user.
Access time remaining	Duration for which the connection exists.

Table 36: show security firewall-authentication users identifier Output Fields *(continued)*

Field Name	Field Description
Source zone	User traffic received from the zone.
Destination Zone	User traffic destined to the zone.
Policy Name	Name of the policy.
Access profile	Name of profile used for authentication.
Interface Name	Name of the interface
Bytes sent by this user	Number of bytes sent by the user.
Bytes received by this user	Number of bytes received by the user.

Sample Output

show security firewall-authentication users identifier 3

```
user@host> show security firewall-authentication users identifier 3
Username: u1
Source IP: 198.51.100.39
Authentication state: Success
Authentication method: Pass-through using HTTP
Age: 1
Access time remaining: 254
Source zone: Z1
Destination zone: Z2
Policy name: Z1-Z2
Access profile: profile-local
Interface Name: ge-0/0/1.0
Bytes sent by this user: 0
Bytes received by this user: 449
```

Sample Output

show security firewall-authentication users identifier 3 node primary

```
user@host> show security firewall-authentication users identifier 3 node primary
node0:
--------------------------------------------------------------------------
Username: local1
Source IP: 198.51.100.39
Authentication state: Success
Authentication method: Pass-through using Telnet
Age: 1
Access time remaining: 5
Source zone: z1
Destination zone: z2
Policy name: POL1
Access profile: p1
Interface Name: reth1.0
Bytes sent by this user: 614
Bytes received by this user: 1880
```

show security policies

Syntax

show security policies
none
<detail>
policy-name *policy-name*
<global>

Release Information

Command modified in Junos OS Release 9.2. Support for IPv6 addresses added in Junos OS Release 10.2. Support for wildcard addresses added in Junos OS Release 11.1. Support for global policy added in Junos OS Release 11.4. Support for services offloading added in Junos OS Release 11.4. Support for source-identities added in Junos OS Release 12.1. The **Description** output field added in Junos OS Release 12.1. Support for negated address added in Junos OS Release 12.1X45-D10. The output fields for Policy Statistics expanded, and the output fields for the **global** and **policy-name** options expanded to include from-zone and to-zone global match criteria in Junos OS Release 12.1X47-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20. Output field and description for **source-end-user-profile** option added in Junos OS Release 15.1x49-D70.

Description

Display a summary of all security policies configured on the device. If a particular policy is specified, display information specific to that policy.

Options

- **none**—Display basic information about all configured policies.
- **detail**—(Optional) Display a detailed view of all of the policies configured on the device.
- **policy-name** ***policy-name***—(Optional) Display information about a specified policy.
- **global**—(Optional) Display information about global policies.

Required Privilege Level

view

Related Documentation

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

List of Sample Output

show security policies on page 492
show security policies policy-name detail on page 493
show security policies (Services-Offload) on page 494
show security policies (Device Identity) on page 494
show security policies detail on page 494
show security policies detail (TCP Options) on page 495
show security policies policy-name (Negated Address) on page 496
show security policies policy-name detail (Negated Address) on page 496
show security policies global on page 496

Output Fields

Table 37 on page 490 lists the output fields for the **show security policies** command. Output fields are listed in the approximate order in which they appear.

Table 37: show security policies Output Fields

Field Name	Field Description
From zone	Name of the source zone.
To zone	Name of the destination zone.
Policy	Name of the applicable policy.
Description	Description of the applicable policy.
State	Status of the policy: • **enabled**: The policy can be used in the policy lookup process, which determines access rights for a packet and the action taken in regard to it. • **disabled**: The policy cannot be used in the policy lookup process, and therefore it is not available for access control.
Index	Internal number associated with the policy.
Sequence number	Number of the policy within a given context. For example, three policies that are applicable in a from-zoneA-to-zoneB context might be ordered with sequence numbers 1, 2, 3. Also, in a from-zoneC-to-zoneD context, four policies might have sequence numbers 1, 2, 3, 4.
Source addresses	For standard display mode, the names of the source addresses for a policy. Address sets are resolved to their individual names. For detail display mode, the names and corresponding IP addresses of the source addresses for a policy. Address sets are resolved to their individual address name-IP address pairs.
Destination addresses	Name of the destination address (or address set) as it was entered in the destination zone's address book. A packet's destination address must match this value for the policy to apply to it.
source-end-user-profile	Name of the device identity profile (referred to as **end-user-profile** in the CLI) that contains attributes, or characteristics of a device. Specification of the device identity profile in the **source-end-user-profile** field is part of the device identity feature. If a device matches the attributes specified in the profile and other security policy parameters, then the security policy's action is applied to traffic issuing from the device.
Source addresses (excluded)	Name of the source address excluded from the policy.
Destination addresses (excluded)	Name of the destination address excluded from the policy.
Source identities	One or more user roles specified for a policy.

Table 37: show security policies Output Fields *(continued)*

Field Name	Field Description
Applications	Name of a preconfigured or custom application whose type the packet matches, as specified at configuration time. • **IP protocol**: The Internet protocol used by the application—for example, TCP, UDP, ICMP. • **ALG**: If an ALG is explicitly associated with the policy, the name of the ALG is displayed. If **application-protocol ignore** is configured, **ignore** is displayed. Otherwise, **0** is displayed. However, even if this command shows **ALG: 0**, ALGs might be triggered for packets destined to well-known ports on which ALGs are listening, unless ALGs are explicitly disabled or when **application-protocol ignore** is not configured for custom applications. • **Inactivity timeout**: Elapsed time without activity after which the application is terminated. • **Source port range**: The low-high source port range for the session application.
Destination Address Translation	Status of the destination address translation traffic: • **drop translated**—Drop the packets with translated destination addresses. • **drop untranslated**—Drop the packets without translated destination addresses.
Application Firewall	An application firewall includes the following: • **Rule-set**—Name of the rule set. • **Rule**—Name of the rule. ◦ **Dynamic applications**—Name of the applications. ◦ **Dynamic application groups**—Name of the application groups. ◦ **Action**—The action taken with respect to a packet that matches the application firewall rule set. Actions include the following: ▪ **permit** ▪ **deny** • **Default rule**—The default rule applied when the identified application is not specified in any rules of the rule set.
Action or Action-type	• The action taken in regard to a packet that matches the policy's tuples. Actions include the following: ◦ **permit** ◦ **firewall-authentication** ◦ **tunnel ipsec-vpn** *vpn-name* ◦ **pair-policy** *pair-policy-name* ◦ **source-nat pool** *pool-name* ◦ **pool-set** *pool-set-name* ◦ **interface** ◦ **destination-nat** *name* ◦ **deny** ◦ **reject** ◦ **services-offload**
Session log	Session log entry that indicates whether the **at-create** and **at-close** flags were set at configuration time to log session information.

Table 37: show security policies Output Fields *(continued)*

Field Name	Field Description
Scheduler name	Name of a preconfigured scheduler whose schedule determines when the policy is active and can be used as a possible match for traffic.

Policy statistics	• **Input bytes**—The total number of bytes presented for processing by the device.
	  ◦ **Initial direction**—The number of bytes presented for processing by the device from the initial direction.
	  ◦ **Reply direction**—The number of bytes presented for processing by the device from the reply direction.
	• **Output bytes**—The total number of bytes actually processed by the device.
	  ◦ **Initial direction**—The number of bytes from the initial direction actually processed by the device.
	  ◦ **Reply direction**—The number of bytes from the reply direction actually processed by the device.
	• **Input packets**—The total number of packets presented for processing by the device.
	  ◦ **Initial direction**—The number of packets presented for processing by the device from the initial direction.
	  ◦ **Reply direction**—The number of packets presented for processing by the device from the reply direction.
	• **Output packets**—The total number of packets actually processed by the device.
	  ◦ **Initial direction**—The number of packets actually processed by the device from the initial direction.
	  ◦ **Reply direction**—The number of packets actually processed by the device from the reply direction.
	• **Session rate**—The total number of active and deleted sessions.
	• **Active sessions**—The number of sessions currently present because of access control lookups that used this policy.
	• **Session deletions**—The number of sessions deleted since system startup.
	• **Policy lookups**—The number of times the policy was accessed to check for a match.

Per policy TCP Options	Configured syn and sequence checks, and the configured TCP MSS value for the initial direction and /or the reverse direction.

Sample Output

show security policies

```
user@host> show security policies
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Sequence number: 1
    Source addresses:
    sa-1-ipv4: 198.51.100.11/24
    sa-2-ipv6: 2001:db8:a0b:12f0::1/32
    sa-3-ipv6: 2001:db8:a0b:12f0::22/32
    sa-4-wc:   203.0.113.1/255.255.0.255
    Destination addresses:
    da-1-ipv4: 2.2.2.2/24
    da-2-ipv6: 2001:db8:a0b:12f0::8/32
    da-3-ipv6: 2001:db8:a0b:12f0::9/32
    da-4-wc:   192.168.22.11/255.255.0.255
```

```
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, application services, log, scheduled
    Application firewall : my_ruleset1
  Policy: p2, State: enabled, Index: 5, Sequence number: 2
    Source addresses:
    sa-1-ipv4:  198.51.100.11/24
    sa-2-ipv6:  2001:db8:a0b:12f0::1/32
    sa-3-ipv6:  2001:db8:a0b:12f0::22/32
    Destination addresses:
    da-1-ipv4:  2.2.2.2/24
    da-2-ipv6:  2001:db8:a0b:12f0::1/32
    da-3-ipv6:  2001:db8:a0b:12f0::9/32
    Source identities: role1, role4
    Applications: any
    Action: deny, scheduled
```

show security policies policy-name detail

```
user@host> show security policies policy-name p1 detail
Policy: p1, action-type: permit, State: enabled, Index: 4
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    sa-1-ipv4:  198.51.100.11/24
    sa-2-ipv6:  2001:db8:a0b:12f0::1/32
    sa-3-ipv6:  2001:db8:a0b:12f0::9/32
    sa-4-wc:    203.0.113.1/255.255.0.255
  Destination addresses:
    da-1-ipv4: 192.0.2.0/24
    da-2-ipv6: 2001:db8:a0b:12f0::1/32
    da-3-ipv6: 2001:db8:a0b:12f0::9/32
    da-4-wc:   192.168.22.11/255.255.0.255
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Destination Address Translation: drop translated
  Application firewall :
  Rule-set: my_ruleset1
    Rule: rule1
      Dynamic Applications: junos:FACEBOOK-ACCESS, junos:YMSG
      Dynamic Application groups: junos:web, junos:chat
      Action: deny
    Default rule: permit
  Session log: at-create, at-close
  Scheduler name: sch20
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                 272 bps
    Output bytes       :               18144                 545 bps
      Initial direction:                9072                 272 bps
      Reply direction  :                9072                 272 bps
    Input  packets     :                 216                   6 pps
```

```
    Initial direction:                108                    3 bps
    Reply direction  :                108                    3 bps
  Output packets     :                216                    6 pps
    Initial direction:                108                    3 bps
    Reply direction  :                108                    3 bps
  Session rate       :                108                    3 sps
  Active sessions    :                 93
  Session deletions  :                 15
  Policy lookups     :                108
```

show security policies (Services-Offload)

```
user@host> show security policies
Default policy: deny-all
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload, count
From zone: untrust, To zone: trust
  Policy: p2, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload
```

show security policies (Device Identity)

```
user@host>  show security policies
From zone: trust, To zone: untrust
       Policy: dev-id-marketing, State: enabled, Index: 5, Scope Policy: 0,
Sequence number: 1
         Source addresses: any
         Destination addresses: any
         source-end-user-profile: marketing-profile
         Applications: any
         Action: permit
```

show security policies detail

```
user@host> show security policies detail
Default policy: deny-all
Policy: p1, action-type: permit, services-offload:enabled , State: enabled, Index:
 4, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
```

```
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                272 bps
    Output bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                272 bps
    Input  packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Output packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Session rate       :                 108                   3 sps
    Active sessions    :                  93
    Session deletions  :                  15
    Policy lookups     :                 108
Policy: p2, action-type: permit, services-offload:enabled , State: enabled, Index:
 5, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p2 is for the sales team
  Sequence number: 1
  From zone: untrust, To zone: trust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

show security policies detail (TCP Options)

```
user@host> show security policies policy-name policy1 detail
node0:
--------------------------------------------------------------------------
Policy: policy1, action-type: permit, State: enabled, Index: 7, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 2
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
```

```
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
  Per policy TCP MSS: initial: 800, reverse: 900
```

show security policies policy-name (Negated Address)

```
user@host> show security policies policy-name p1
node0:
--------------------------------------------------------------------------
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses(excluded): as1
    Destination addresses(excluded): as2
    Applications: any
    Action: permit
```

show security policies policy-name detail (Negated Address)

```
user@host> show security policies policy-name p1 detail
node0:
--------------------------------------------------------------------------
Policy: p1, action-type: permit, State: enabled, Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses(excluded):
    ad1(ad): 255.255.255.255/32
    ad2(ad):  198.51.100.1/24
    ad3(ad):  198.51.100.6  ~ 198.51.100.56
    ad4(ad):  192.0.2.8/24
    ad5(ad):  198.51.100.99 ~ 198.51.100.199
    ad6(ad):  203.0.113.9/24
    ad7(ad):  203.0.113.23/24
  Destination addresses(excluded):
    ad13(ad2): 198.51.100.76/24
    ad12(ad2): 198.51.100.88/24
    ad11(ad2): 192.0.2.23 ~ 192.0.2.66
    ad10(ad2): 192.0.2.93
    ad9(ad2):  203.0.113.76 ~ 203.0.113.106
    ad8(ad2):  203.0.113.199
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

show security policies global

```
user@host> show security policies global policy-name Pa
node0:
--------------------------------------------------------------------------
  Global policies:
  Policy: Pa, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 1
    From zones: zone1, zone2
    To zones: zone3, zone4    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

show service user-identification authentication-source aruba-clearpass user-query counters

Syntax show service user-identification authentication-source aruba-clearpass user-query counters

Release Information Command introduced in Junos OS Release 12.3X48-D30.

Description Display statistics on the counters maintained by the user query function. The output identifies the ClearPass webserver as the destination of the user query requests. It displays the number of requests sent from the SRX Series device to the ClearPass webserver and the number of responses that the SRX Series device received from it. You can use this command to identify that a problem exists—the number of responses received is less than the number of requests sent.—and then analyze and correct it.

If there are no problems with the communication between the ClearPass Policy Manager (CPPM) and the SRX Series device, the number of requests sent is equal to the number of responses received and the number of error responses.

number-of-requests = number-of-responses + error-message-responses

The user query function is part of the SRX Series integrated ClearPass authentication and enforcement feature. The SRX Series device can automatically send requests for individual user authentication and identity information to ClearPass in the event that ClearPass does not post that information to it. For this to occur, you must have configured the user query function.

The SRX Series device exposes to ClearPass a Web API (webapi) that ClearPass uses to send POST request messages to it automatically. These messages contain user authentication and identity information.

The user query function supplements use of the SRX Series Web API function.

Options ***authentication-source***—Specify aruba-clearpass to identifies Aruba ClearPass as the authentication source.

Required Privilege Level view

Output Fields

- Webserver Address—The IP address of the ClearPass webserver.
- Access token—The token string that the SRX Series device obtains from ClearPass which allows the SRX Series device to query the ClearPass webserver for an individual user's authentication and identity information.
- Requests sent number—A counter that shows the number of individual user authentication information queries that the SRX Series device sent to the ClearPass webserver.
- Total response received number—A counter that shows the number of returns from the ClearPass webserver in response to the individual user authentication information queries that the SRX Series device sent to it. The number of responses should match the number of requests unless an error occurred.

- Error response received number—The number errors that occurred in relation to requests.
- Time of last response—A timestamp showing when the last response from the ClearPass webserver was received.

Sample Output

show service user-identification authentication-source aruba-clearpass user-query counters

```
user@host> show service user-identification authentication-source aruba-clearpass user-query
counters

    Web server Address: 4.0.0.20
    Access token: 433feffae5c3eb3ff8ffdc49f968b03437ca1ce5
    Request sent number: 7
    Total response received number: 7
    Error response received number: 0
    Time of last response: 2000-01-01 11:57:17
```

show service user-identification authentication-source aruba-clearpass user-query status

Syntax show service user-identification authentication-source *authentication-source* user-query status

Release Information Command introduced in Junos OS Release 12.3X48-D30.

Description Checks to determine if the ClearPass webserver is online. The SRX Series device sends user query requests to the ClearPass webserver. The user query function is part of the SRX Series ClearPass Authentication and Enforcement feature.

Options ***authentication-source***—Identifies the authentication source. For the integrated ClearPass feature, you must specify the predefined term aruba-clearpass to determine if the ClearPass webserver is online.

Required Privilege Level view

List of Sample Output
command-name (optional-text) on page 499
command-name (optional-text) on page 499
command-name (optional-text) on page 499

Output Fields

Sample Output

command-name (optional-text)

```
user@host> command-name option1 option2
Paste router command output here
```

command-name (optional-text)

```
user@host> command-name option1 option2
Paste
router command output here
```

command-name (optional-text)

```
user@host> command-name option1 option2
Paste router command
output here
```

show services unified-access-control authentication-table

Syntax show services unified-access-control authentication-table

Release Information Command introduced in Junos OS Release 9.4. Options updated in Junos OS Release 12.1.

Description Display a summary of the authentication table entries configured from the IC Series UAC Appliance. Authentication tables store mappings between traffic sessions and Unified Access Control (UAC) roles. The IC Series appliance uses the roles specified in the mappings to help determine which UAC policies to apply to a session.

Use this command when you have configured the SRX Series device to act as a Junos OS Enforcer in a UAC deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series appliance.

You can also use this command to display the content of the authentication table in a user role firewall implementation. The table, pushed from a supporting UAC device, provides the user roles associated with incoming traffic.

Options

- **detail**—Display a detailed view of all authentication table entries.
- **extended**—Display a view of all authentication table entries with the user roles listed.
- **identifier** ***id***—Display all authentication table entries with the specified identifier number.
- **ip** ***source-ip-address***—Display any authentication table entry for the specified IP address.
- **role** ***role-name***—Display all authentication table entries for the specified role name.
- **user** ***username***—Display all authentication table entries for the specified user.

Required Privilege Level view

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output

Sample Output

show services unified-access-control authentication-table

```
user@host>show services unified-access-control authentication-table
Id    Source IP       Username     Age      Role identifier
1     198.51.100.22          user1        0        0000000001.000005.0
Total: 1
```

show services unified-access-control authentication-table detail

```
user@host>show services unified-access-control authentication-table detail
Identifier: 1
  Source IP: 198.51.100.22
  Username: john
  Age: 0
  Role identifier       Role name
    0000000001.000005.0 Users
    1113249951.100616.0 PersonalFirewall
    1183670148.427197.0 UAC
Total: 1
```

show services unified-access-control authentication-table extended

```
user@host>show services unified-access-control authentication-table extended
Id    Source IP       Username     Age   Role name
3     10.214.161.195          johna      60    Users, PersonalFirewall
6     10.214.161.183          mayb      60    role-1
Total: 2
```

show services unified-access-control authentication-table identifier id

```
user@host>show services unified-access-control authentication-table identifier 1
Identifier: 1
  Source IP: 10.214.161.195
  Username: johna
  Age: 0
  Role identifier       Role name
    0000000001.000005.0 Users
    1113249951.100616.0 PersonalFirewall
    1183670148.427197.0 UAC
Total: 1
```

show services unified-access-control authentication-table ip

```
user@host>show services unified-access-control authentication-table ip 10.214.161.183
Id    Source IP       Username     Age      Role identifier
8     10.214.161.183         mayb          0        1420298444.225667.0
Total: 1
```

show services unified-access-control authentication-table role

```
user@host>show services unified-access-control authentication-table role role-1
Id    Source IP       Username     Age      Role identifier
6     10.214.161.183          maybe         60       1420298444.225667.0
Total: 1
```

show services unified-access-control authentication-table user username

```
user@host>show services unified-access-control authentication-table user prasanta
Id    Source IP       Username     Age      Role identifier
7     10.214.161.195         paul1     0        0000000001.000005.0
Total: 1
```

show services user-identification authentication-table

Syntax show services user-identification authentication-table ip-address *ip-address* | authentication-source *authentication-source* (brief | domain *domain-name* (<enter> | brief | extensive) | group *group-name* (<enter> | brief | extensive) | user *user-name* (<enter> | brief | extensive)) all | active directory

Release Information Command introduced in Junos OS release 12.3X48-D30.

Description Display the ClearPass authentication table contents for an individual user based on the IP address of the user's device, the entire ClearPass authentication table contents, users who belong to a domain, users who belong to a group, or a user's entry based on the user's name.

The ClearPass authentication table user entries include authentication and identity information that the SRX Series device obtains from the ClearPass Policy Manager (CPPM). ClearPass, which is the authentication source for the Integrated ClearPass Authentication and Enforcement feature, posts the user authentication information to the SRX Series device. The SRX Series device UserID daemon synchronizes the ClearPass user authentication information from the Routing Engine authentication table, which includes entries from other authentication sources, to the ClearPass authentication table on the Packet Forwarding Engine.

To supplement posting from the ClearPass authentication table, the SRX Series device supports a user query function that allows you to obtain authentication information for an individual user.

Options ***ip-address***—Displays information for a user identified by the IP address of their device.

authentication-source—The authentication source for the Integrated ClearPass Authentication and Enforcement feature. For this feature, you must specify the value aruba-clearpass.

Specify the following identifiers to control the degree and kind of information to display:

brief—The show command displays brief information for ClearPass authentication table user entries. For each domain, it displays the domain name and the number of users who belong to it. For each user, it shows the user's device IP address, username, groups that the user belongs to that are referenced by a security policy, and the state of the user entry.

domain —Specifies the name of domain whose user member information your want to view. You can specify extensive with domain to show extensive information for user entries for all of its members.

extensive—Shows extensive information for the ClearPass authentication table user entries. For each domain, extensive displays the domain name and the number of users who belong to it. For each user, it shows the user's device IP address, username, the groups that the user belongs to, the groups that the user belongs to that are referenced by a security policy, the state of the user entry, the authentication source

(Aruba ClearPass), the access start date and time, a timestamp showing the last time the entry was updated, and the age after which time the entry expires.

You can specify extensive without a qualifying identifier to display extensive information for all of the table's user entries. You can specify it in conjunction with domain, group, or user to display extensive information for that category of users—that is, all members of the domain, all users who belong to the group, or an individual user identified by their username.

group—Specifies the name of the group whose member information you want to view. You can specify extensive with group to show extensive information for users who belong to the group.

user—Specifies the name of the user whose information you want to view. You can specify extensive to show extensive information for that user.

Default: brief

Required Privilege Level view

List of Sample Output

show services user-identification authentication-table authentication-source aruba-clearpass on page 504
show services user-identification authentication-table authentication-source aruba-clearpass domain on page 506
show services user-identification authentication-table authentication-source aruba-clearpass group on page 507
show services user-identification authentication-table authentication-source aruba-clearpass user on page 509

Field Name	Field Description
Domain **Output Fields**	Name of the domain that the users belong to. If the CPPM does not send domain information to the SRX Series device for a user, the user belongs to the GLOBAL domain.
Total entries	Number of user entries in the ClearPass authentication table by domain.
For each entry:	
Source IP	The IP address of the user's device. If a user is logged in to the network with more than one device, a separate entry is created for the user for each device. It showing the devices IP address.
username	The name by which the user is logged in to the network.
Groups	A list of the groups that the user belongs to. The list can include a group that identifies the device posture.
State	The state of the entry. There are four states for an authentication entry: initial, valid, invalid, and pending. • An initial state is a temporary state, and it can be created from either a valid or an invalid entry. • A valid state indicates that the authentication entry has a valid IP address, domain, and username. • An invalid state indicates that the entry does not have a valid IP address, domain, and username. This can happen when the SRX Series device does not receive a query response from the CPPM. If the entry is invalid, it is put in the null domain. • A pending state indicates that the entry was created after the user query was sent and before the response was received.
Source	The name of the authentication source. For the Integrated ClearPass Authentication and Enforcement feature, this value is always aruba-clearpass.
Access start date	The date when the authentication entry was created by the SRX Series device.
Access start time	The time when the authentication entry was created by the SRX Series device.
Last updated timestamp	The time when ClearPass creates the user information. This value is taken from the timestamp field in the user information posted by ClearPass to the SRX Series device.
Age time:	The time after which the entry expires, as configured by the authentication-entry-timeout statement. If a value of 0 was specified, the entry never expires. When an expiration time is reached, the SRX Series device deletes the user entry from the ClearPass authentication table.

Sample Output

show services user-identification authentication-table authentication-source aruba-clearpass

Note that in the following example, the output would show the same results whether or not you specified brief. (The default behavior is to display brief output.)

user@host> **show services user-identification authentication-table authentication-source aruba-clearpass brief**

In this case, if there was more than one domain configured, the output would show the following kind of information for each domain.

```
Domain: GLOBAL
Total entries: 6
Source IP       Username       groups(Ref by policy)          state
203.0.113.21        viki2          accounting-grp-and-company-dev Valid
203.0.113.89        abew1          marketing-access-limited-grp   Valid
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1         corporate-limited              Valid
203.0.113.54        guest1                                        Valid
203.0.113.55        guest2                                        Valid
```

user@host> **show services user-identification authentication-table authentication-source aruba-clearpass extensive**

```
Domain: GLOBAL
Total entries: 6
  Source-ip: 203.0.113.21
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 203.0.113.89
    Username: abew1
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
  Source-ip: 203.0.113.52
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 203.0.113.53
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    Groups referenced by policy:corporate-limited
```

```
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
  Source-ip: 203.0.113.54
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 203.0.113.55
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

show services user-identification authentication-table authentication-source aruba-clearpass domain

Note that in the following example the output would show the same results whether or not you specified brief. The default behavior is to display brief output.

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass domain GLOBAL brief

Domain: GLOBAL
Total entries: 6
Source IP       Username       groups(Ref by policy)          state
203.0.113.21        viki2          accounting-grp-and-company-dev Valid
203.0.113.89        abew1          marketing-access-limited-grp   Valid
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1         corporate-limited              Valid
203.0.113.54        guest1                                        Valid
203.0.113.55            guest2                                        Valid

user@host> show services user-identification authentication-table authentication-source
aruba-clearpass domain GLOBAL extensive
Domain: GLOBAL
Total entries: 6
  Source-ip: 203.0.113.21
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 203.0.113.89
    Username: abew1
```

```
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
  Source-ip: 203.0.113.52
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 203.0.113.53
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    Groups referenced by policy:corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
  Source-ip: 203.0.113.54
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 203.0.113.55
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

show services user-identification authentication-table authentication-source aruba-clearpass group

Note that in the following example, the output would show the same results whether or not you specified brief. (The default behavior is to display brief output.)

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass group posture-healthy brief

Domain: GLOBAL
Source IP       Username       groups(Ref by policy)          state
203.0.113.21        viki2          accounting-grp-and-company-dev Valid
```

```
203.0.113.52        jxchan         marketing-access-for-pcs-limit Valid
203.0.113.53        lchen1         corporate-limited              Valid
203.0.113.54        guest1                                        Valid
203.0.113.55        guest2                                        Valid
```

```
user@host> show services user-identification authentication-table authentication-source
aruba-clearpass group posture-healthy extensive
Domain: GLOBAL
  Source-ip: 203.0.113.21
    Username: viki2
    Groups:posture-healthy, accounting-grp, accounting-grp-and-company-device,
    corporate-limited, [user authenticated]
    Groups referenced by policy:accounting-grp-and-company-device,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:20:30
    Last updated timestamp: 2015-12-22 04:02:48
    Age time: 0
  Source-ip: 203.0.113.52
    Username: jxchan
    Groups:posture-healthy, marketing-access-for-pcs-limited-group,
    marketing-general, sales-limited, corporate-limited, [user authenticated]
    Groups referenced by policy:marketing-access-for-pcs-limited-group,
    corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:22:48
    Last updated timestamp: 2015-12-22 05:46:21
    Age time: 0
  Source-ip: 203.0.113.53
    Username: lchen1
    Groups:posture-healthy, human-resources-grp, accounting-limited,
    corporate-limited, [user authenticated]
    Groups referenced by policy:corporate-limited
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:21:37
    Last updated timestamp: 2015-12-22 05:41:18
    Age time: 0
Source-ip: 203.0.113.54
    Username: guest1
    Groups:posture-healthy, guest, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:10
    Last updated timestamp: 2015-12-22 05:50:47
    Age time: 0
  Source-ip: 203.0.113.55
    Username: guest2
    Groups:posture-healthy, guest-device-byod, [user authenticated]
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:23:21
    Last updated timestamp: 2015-12-22 05:52:44
    Age time: 0
```

Sample Output

show services user-identification authentication-table authentication-source aruba-clearpass user

```
user@host> show services user-identification authentication-source aruba-clearpass user brief
abew1

Domain: GLOBAL
Source IP           Username       groups(Ref by policy)          state
203.0.113.89        abew1          marketing-access-limited-grp   Valid
```

```
user@host> show services user-identification authentication-source aruba-clearpass user
extensive abew1

Domain: GLOBAL
 Source-ip: 203.0.113.89
    Username: abew1
    Groups:posture-unknown, marketing-access-limited-grp, [user authenticated]
    Groups referenced by policy:marketing-access-limited-grp
    State: Valid
    Source: Aruba ClearPass
    Access start date: 2016-03-08
    Access start time: 17:31:40
    Last updated timestamp: 2015-12-22 04:18:48
    Age time: 0
```

show services user-identification active-directory-access user-group-mapping

Syntax show services user-identification active-directory-access user-group-mapping (group *name* | status | user *name*) domain *name*

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Display user-to-group mapping information used in the integrated user firewall feature. Note that the LDAP server is often part of the domain controller.

Options

- **group *group-name***—Display the users mapped to the specified group.
- **status**—Display the status of the last query to the LDAP server for user-group mapping.
- **user *name***—Display the groups for the specified username.
- **domain *name***—(Optional) Display the group, status, or user information for the specified domain.

Required Privilege Level view

Related Documentation

- LDAP Functionality in Integrated User Firewall on page 150
- user-group-mapping on page 428

List of Sample Output

show services user-identification active-directory-access user-group-mapping group domain on page 511
show services user-identification active-directory-access user-group-mapping status on page 511
show services user-identification active-directory-access user-group-mapping user on page 512

Output Fields Table 38 on page 510 lists the output fields for the **show services user-identification active-directory-access user-group-mapping group** command.

Table 38: show services user-identification active-directory-access user-group-mapping group Output Fields

Field Name	Field Description
Domain	Domain of the specified group.
Users	Usernames mapped to the specified group.

Table 39 on page 511 lists the output fields for the **show services user-identification active-directory-access user-group-mapping status** command.

Table 39: show services user-identification active-directory-access user-group-mapping status Output Fields

Field Name	Field Description
Domain	Domain for which the status is displayed.
LDAP server	IP address of the LDAP server.
Port	Port number on the LDAP server.
Last-query-status	Status of the last query from the SRX Series device.
Last-query-time	Year-month-date:hour:minutes:seconds when the SRX device last queried the LDAP server.

Table 40 on page 511 lists the output fields for the **show services user-identification active-directory-access user-group-mapping user** command.

Table 40: show services user-identification active-directory-access user-group-mapping user Output Fields

Field Name	Field Description
Domain controller	Domain controller about which the user information is displayed.
Groups	Groups to which the user belongs.
Referenced by policy	Groups to which the user belongs and that are referenced by a firewall policy.

Sample Output

show services user-identification active-directory-access user-group-mapping group domain

```
user@host> show services user-identification active-directory-access user-group-mapping group
finance domain www.apac-acme.net
show services user-identification active-directory-access user-group-mapping group
 finance-group
Domain: example-domain.net
Users: user1, user2
Domain: example2.domain.net
Users: user3
```

Sample Output

show services user-identification active-directory-access user-group-mapping status

```
user@host> show services user-identification active-directory-access user-group-mapping status
Domain: example-domain.net
LDAP server    Port      Last-query-status    Last-query-time
192.0.2.87      389       Query success        2014-02-07:15:50:52

Domain: example2.domain.net
LDAP server    Port      Last-query-status    Last-query-time
192.0.2.144     389       Idle                 0
```

Sample Output

show services user-identification active-directory-access user-group-mapping user

```
user@host> show services user-identification active-directory-access user-group-mapping user
user1
Domain example-domain.net
Groups: Dev, NAT, SBU
Referenced by policy: SBU
Domain: example2.domain.net
Groups: HR, USA
```

show services user-identification device-information table

Syntax show services user-identification device-information table all (brief | domain | extensive) | device-id *device-id* (brief |domain |extensive) | ip-address *ip-address*

Release Information Statement introduced in Junos OS Release 5.1X49-D70.

Description Display the contents of the device identity authentication table. The device identity authentication table includes entries for authenticated devices whose information is obtained from external authentication sources. A device identity entry contains the device's IP address, the device ID, and a list of groups that the device belongs to. It also contains attributes that are configured in the device identity profile—for example, the type of device, the vendor, and the operating system that is running on the device and its version.

The device identity authentication table is separate from the Active Directory authentication table or any other local authentication table that is used for Junos OS features, or for specific third-party authentication sources. Also, unlike local user authentication tables, which are specific to an authentication source, the device identity authentication table holds device identity information for devices authenticated by different sources.

Only one authentication source, such as Active Directory, can be active at a time. A result of this requirement is that there is less demand on the system to process information.

Options **all**—Display information for all authenticated devices with entries in the table.

device-id—Display information for the authenticated device whose device ID is specified.

ip-address—Display information for the authenticated device whose IP address is specified.

brief—Display terse information for the entries in the device identity authentication table entries. You can specify **brief** as a keyword to the parameters **all** and **device-id**.

domain —Display the name of domain and information for all authenticated devices that belong to the domain. You can specify **domain** as a keyword to the parameters **all** and **device-id**.

extensive—Display**extensive** information for all of the authenticated devices for which there are table entries. It displays the domain name, the IP address of the device, the device's ID, the device category and vendor, the device type, and the operating system running on the device and its version.

Required Privilege Level view

Related Documentation

- Understanding the Device Identity Attributes and Profiles for the Integrated User Firewall Device Identity Authentication Feature on page 188
- Understanding the Device Identity Authentication Table and Its Entries on page 192

- Understanding Access Control to Network Resources Based on Device Identity Information on page 185
- authentication-source (Services User Identification Device Identity) on page 303
- source-end-user-profile on page 395

Table 41: show services user-identification device-information table Output Fields

Field Name	Field Description
Domain name	The name of the domain to which the devices belong.
NOTE: For each authenticated device, the following information is displayed when the parameter **all** is specified after **table** and it is modified by the keyword **extensive**.	
Source IP address	The IP address of the device.
Device ID	The ID assigned to the device.
Device-Groups	The groups to which the device belongs.
device-category	The kind of device. For example, the device might be a laptop. You configured this value as part of the device identity profile.
device-vendor	The maker of the device. For example, the device vendor might be Lenovo.
device-type	The device type. If this device is a laptop made by Lenovo, it might be of type thinkpad-t430.
device-os	The operating system that is running on the device. The operating system might be Windows.
device-os-version	The version of the operating system running on the device. For example, for Windows, this might be 7.1.
Location1	The location where the device is being used. The location might be specified as United States.
Referred by	The security policy that refers to the device in its **source-end-user-profile** field. The **source-end-user-profile** that you configure might pertain to a group of devices or a single device.

Sample Output

show services user-identification device-information table

```
user@host> show services user-identification device-information table all extensive
Domain: example.net
Total entries: 3
  Source IP:192.0.2.11
    Device ID: dev01
   Device-Groups: device_group01, device_group02, device_group03, device_group04,
 device_group05
    device-category: laptop
    device-vendor: lenovo
```

```
   device-type: thinkpad-t430
   device-os: windows
   device-os-version: 7.1
   Location1: us1
   Referred by: My-pf_0
 Source IP: 192.0.2.12
   Device ID: dev02
   Device-Groups: device_group06, device_group07, device_group08, device_group09,
device_group10
   device-category: laptop
   device-vendor: lenovo
   device-type: thinkpad-t430
   device-os: windows
   device-os-version: 7.1
   Location1: us1
   Referred by: My-pf_0
 Source IP: 192.0.2.14
   Device ID: dev03
   Device-Groups: device_group01, device_group02, device_group03, device_group04,
device_group05
   device-category: laptop
   device-vendor: lenovo
   device-type: thinkpad-t430
   device-os: windows
   device-os-version: 7.1
   Location1: us1
   Referred by: My-pf_0
```

show services unified-access-control counters

Syntax show services unified-access-control counters

Release Information Command introduced in Junos OS Release 12.1X44-D10.

Description Display the number of sessions allowed, denied, and terminated by the Unified Access Control (UAC) service when invoked by a firewall policy with the uac-policy action. Counts are reported for each action taken by UAC. Sessions that were allowed, denied, or terminated by other firewall policy actions are not included in these statistics.

On high-end SRX Series devices, UAC counts are grouped and displayed for each PIC on the device. On branch SRX Series devices, UAC counts are accumulated by device only. There is no PIC specification on these devices.

Required Privilege Level view

Related Documentation
- Firewall User Authentication Overview on page 9

List of Sample Output show services unified-access-control counters on page 517

Output Fields Table 42 on page 516 lists the output fields for the **show services unified-access-control counters** command. Output fields are listed in the approximate order in which they appear.

Table 42: show services unified-access-control counters Output Fields

Field Name	Field Description
PIC	If applicable, the number of each PIC implementing UAC. UAC statistics are grouped by PIC.
Sessions allowed	The sessions permitted by UAC when invoked by a user role firewall policy.
Policy action	Number of sessions permitted by UAC based on the UAC policy action.
Timeout action	Number of sessions permitted by the timeout action while the SRX was disconnected from the UAC device.
Sessions denied	The sessions denied by UAC when invoked by a user role firewall policy.
Unauthenticated	Number of sessions denied by UAC because the user was not authenticated.
Policy action	Number of sessions denied by UAC based on the UAC policy action.
Policy not matched	Number of sessions denied because no UAC policy match was found.
Timeout action	Number of sessions denied by the timeout action while the SRX was disconnected from the access control device.
Sessions terminated	The sessions originally permitted that were later terminated.

Table 42: show services unified-access-control counters Output Fields *(continued)*

Field Name	Field Description
Reevaluation	Number of sessions terminated due to a change in the UAC user roles associated with the session.
Signout	Number of sessions terminated due to the user signing out.

Sample Output

show services unified-access-control counters

```
user@host> show services unified-access-control counters
PIC: fpc2.pic0
  Sessions allowed
    Policy action: 0
    Timeout action: 0
  Sessions denied
    Unauthenticated: 0
    Policy action: 0
    Policy not matched: 0
    Timeout action: 0
  Sessions terminated
    Reevaluation: 0
    Signout: 0
```

Statistics on branch devices are accumulated by device only. There is no PIC specification on these devices.

```
user@host> show services unified-access-control counters
Sessions allowed
    Policy action: 0
    Timeout action: 0
  Sessions denied
    Unauthenticated: 0
    Policy action: 0
    Policy not matched: 0
    Timeout action: 0
  Sessions terminated
    Reevaluation: 0
    Signout: 0
```

show services unified-access-control policies

Syntax show services unified-access-control policies

Release Information Command introduced in Junos OS Release 9.4.

Description Display a summary of resource access policies configured from the IC Series UAC Appliance.

Use this command when you have configured the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series appliance.

Options

- **detail**—Display a detailed view of all policies.
- **identifier *id***—Display information about a specific policy by identification number.

Required Privilege Level view

Related Documentation

- Firewall User Authentication Overview on page 9

List of Sample Output

show services unified-access-control policies on page 518
show services unified-access-control policies detail on page 518
show services unified-access-control policies identifier 1 on page 519

Sample Output

show services unified-access-control policies

```
user@host> services unified-access-control policies
Id    Resource                Action Apply        Role identifier
1     10.100.15.0/24:*        allow  selected     1113249951.100616.0
2     10.100.17.0/24:*        deny   all
```

Sample Output

show services unified-access-control policies detail

```
user@host> services unified-access-control policies detail
Identifier: 1
  Resource: 10.100.15.0/24:*
  Resource: 10.100.16.23-10.100.16.60:*
  Action: allow
  Apply: selected
  Role identifier       Role name
    1113249951.100616.0 Personal Firewall
    1112927873.881659.0 Antivirus
    1183670148.427197.0 UAC
Identifier: 2
  Resource: 10.100.17.0/24:*
  Resource: 10.100.16.23-10.100.16.60:*
  Resource: 10.100.18.0/24:*
```

```
  Action: deny
  Apply: all
```

Sample Output

show services unified-access-control policies identifier 1

```
user@host> show services unified-access-control policies identifier 1
Identifier: 1
  Resource: 10.100.15.0/24:*
  Resource: 10.100.16.23-10.100.16.60:*
  Action: allow
  Apply: selected
  Role identifier       Role name
    1113249951.100616.0 Personal Firewall
    1112927873.881659.0 Antivirus
    1183670148.427197.0 UAC
```

show services unified-access-control roles

Syntax show services unified-access-control roles

Release Information Command introduced in Junos OS Release 12.1.

Description When implementing user role firewall, display a summary of the roles that have been pushed to the SRX Series device from the access control service.

Required Privilege Level view

Related Documentation
- *Security Basics Guide for Security Devices*
- Firewall User Authentication Overview on page 9

List of Sample Output show services unified-access-control roles on page 520

Output Fields Table 43 on page 520 lists the output fields for the **show services unified-access-control roles** command. Output fields are listed in the approximate order in which they appear.

Table 43: show services unified-access-control roles Output Fields

Field Name	Field Description
Name	Name of the user role.
Identifier	Unique identifier associated with the specified user role.
Total	Total number of user roles specified in the table.

Sample Output

show services unified-access-control roles

```
user@host> show services unified-access-control roles
Name                                            Identifier
Users                                           0000000001.000005.0
admin-1                                         1420298444.225667.0
Total: 2
```

show services unified-access-control status

Syntax show services unified-access-control status

Release Information Command introduced in Junos OS Release 9.4.

Description Display the status of the connection between the SRX Series device and the IC Series UAC Appliance as well as statistics to help debug connections to the IC Series appliance.

Use this command when you have configured the SRX Series device to act as a Junos OS Enforcer in a Unified Access Control (UAC) deployment. When deployed as a Junos OS Enforcer, the SRX Series device enforces the policies that are defined on the UAC's IC Series appliance.

Required Privilege Level view

Related Documentation
- Firewall User Authentication Overview on page 9

List of Sample Output show services unified-access-control status on page 521

Sample Output

show services unified-access-control status

```
user@host> show services unified-access-control status
Host           Address         Port   Interface     State
dev106vm26     10.64.11.106    11123  ge-0/0/0.0    connected
dev107vm26     10.64.11.106    11123  ge-0/0/0.0    closed
```

show services user-identification active-directory-access active-directory-authentication-table

Syntax

show services user-identification active-directory-access
active-directory-authentication-table
(all | group *name* | ip-address *ip-address* | user *name*
<domain *name*> <node (***node-id*** | all | local | primary)> <brief | extensive>

Release Information

Command introduced in Junos OS Release 12.1X47-D10.

Description

Display information about all entries in the Active Directory authentication table used in the integrated user firewall feature, or for a specific group, IP address or user.

Options

- **all**—Summary of the authentication entry information.
- **group *group-name***—Display the entries from the authentication table for the specified group.
- **ip-address *ip-address***—Display the entries from the authentication table for the specified IP address.
- **user *name***—Display the entries from the authentication table for the specified username.
- **domain *name***—(Optional) Display the summary, group, or user entries for the specified domain.
- **node**—(Optional) For chassis cluster configurations, display the summary, IP address, or user entries for a specific node.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.
- **brief | extensive**—Display the specified level of output (the default is brief).

Required Privilege Level

view

Related Documentation

- clear services user-identification active-directory-access on page 456
- show services user-identification active-directory-access domain-controller status on page 526

List of Sample Output

show services user-identification active-directory-access active-directory-authentication-table ip-address <ip-address> on page 523
show services user-identification active-directory-access active-directory-authentication-table all on page 524
show services user-identification active-directory-access active-directory-authentication-table all domain on page 524

show services user-identification active-directory-access active-directory-authentication-table all extensive on page 524

Output Fields Table 44 on page 523 lists the output fields for the **show services user-identification active-directory-access active-directory-authentication-table all extensive** command.

Table 44: show services user-identification active-directory-access active-directory-authentication-table all extensive Output Fields

Field Name	Field Description
Source IP	IP address for user who is logged in through the domain controller.
Username	ID of the user who is logged in through the domain controller.
Groups	Groups to which the user is associated in the domain controller.
State	States include the following: Pending—This IP address is being probed. Initial—The authentication entry is only received from the WMIC daemon, not pushed to the Packet Forwarding Engine. Valid—The authentication entry is pushed to the Packet Forwarding Engine. Invalid—The PC probe failed.
Access start date	Date that the authentication entry was created.
Access start time	Time that the authentication entry was created.
Age time	Number of minutes after which the authentication entry will time out.

Sample Output

show services user-identification active-directory-access active-directory-authentication-table ip-address <ip-address>

```
user@host> show services user-identification active-directory-access
active-directory-authentication-table ip-address 192.0.2.3
Domain: ad02.net
 Source-ip: 192.0.2.3
  Username: user1
  Groups:group1
  State: Valid
  Source: wmic
  Access start date: 2014-03-10
  Access start time: 13:59:56
  Age time: 1437
```

Sample Output

show services user-identification active-directory-access active-directory-authentication-table all

```
user@host> show services user-identification active-directory-access
active-directory-authentication-table all
Domain: www.engineering-example.net
Total count: 2
Source IP       Username      Groups          State
 10.1.1.2         u2           r1, r3, r4      initial
 10.1.1.3         u3           r5, r6, r4      pending

Domain: www.hr-example.net
Total count: 2
Source IP       Username      Groups          State
 10.1.1.5         u4           r1, r3, r4      initial
 10.1.1.6         u5           r5, r6, r4      pending
```

Sample Output

show services user-identification active-directory-access active-directory-authentication-table all domain

```
user@host> show services user-identification active-directory-access
active-directory-authentication-table all domain www.mycompany-example.com
Domain: www.mycompany-example.com
Total count: 2
Source IP       Username      Groups          State
 10.1.1.2         u2          r1, r3, r4      initial
 10.1.1.3         u3          r5, r6, r4      pending
```

Sample Output

show services user-identification active-directory-access active-directory-authentication-table all extensive

```
user@host> show services user-identification active-directory-access
active-directory-authentication-table all extensive
Domain: www.mycompany-example.com
Total entries: 2

Source IP: 10.1.1.2
Username: u2
Groups: r1, r3, r4
State: inital
Access start date: 2013-05-22
Access start time: 10:56:58
Age time: 20 min

Source IP: 10.1.1.3
Username: u3
Groups: r5, r6, r4
State: pending
Access start date: 2013-05-22
Access start time: 10:56:58
Age time: 20 min

Domain: www.hr-example.net
Total entries: 2

Source IP: 10.1.1.2
Username: u2
```

```
Groups: r1, r3, r4
State: inital
Access start date: 2013-05-22
Access start time: 10:56:58
Age time: 20 min

Source IP: 10.1.1.3
Username: u3
Groups: r5, r6, r4
State: pending
Access start date: 2013-05-22
Access start time: 10:56:58
Age time: 20
```

show services user-identification active-directory-access domain-controller status

Syntax show services user-identification active-directory-access domain-controller status <domain *name*> <node (***node-id*** | all | local | primary)> <brief | extensive>

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Display status information for the Active Directory domain controllers configured for the integrated user firewall feature.

Options

- **domain *name***—(Optional) Display the status of the domain controllers for a specific domain.
- **node**—(Optional) For chassis cluster configurations, display the status of the domain controllers for a specific node.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.
- **brief | extensive**—Display the specified level of output (the default is brief).

Required Privilege Level view

Related Documentation

- active-directory-access on page 294
- show services user-identification active-directory-access active-directory-authentication-table on page 522

List of Sample Output

show services user-identification active-directory-access domain-controller status on page 527
show services user-identification active-directory-access domain-controller status brief domain on page 527
show services user-identification active-directory-access domain-controller status extensive domain on page 527

Output Fields Table 45 on page 526 lists the output fields for the **show services user-identification active-directory-access domain-controller status** command.

Table 45: show services user-identification active-directory-access domain-controller Output Fields

Field Name	Field Description
Domain controller	Domain controller name.
Address	IP address of the domain controller.

Table 45: show services user-identification active-directory-access domain-controller Output Fields *(continued)*

Field Name	Field Description
Status	Connection status of the domain controller: connected or disconnected.
Reason	Reason for a disconnected status: network issue, authentication failed, or host unreachable.

Sample Output

show services user-identification active-directory-access domain-controller status

Displays brief information for domain controllers in all configured domains.

```
user@host> show services user-identification active-directory-access domain-controller status
Domain: example-domain-controller.com
    Domain controller       Address         Status
    DC1                   203.0.113.51      Connected
    DC2                   203.0.113.12      Connected
    DC3                   203.0.113.6       Connected
    DC4                   203.0.113.11      Disconnected
    DC5                   203.0.113.7       Disconnected

Domain: example-domain
    Domain controller       Address          Status
    example-domain10          10.1.1.1       Disconnected
    example-domain20          10.2.2.2       Disconnected
    example-domain30          10.3.3.3       Disconnected
```

Sample Output

show services user-identification active-directory-access domain-controller status brief domain

```
user@host> show services user-identification active-directory-access domain-controller status
brief domain example-domain-controller.com
Domain: example-domain-controller.com
    Domain controller       Address           Status
    DC1                   203.0.113.51        Connected
    DC2                   203.0.113.12        Connected
    DC3                   203.0.113.6         Connected
    DC4                   203.0.113.11        Disconnected
    DC5                   203.0.113.7         Disconnected
```

Sample Output

show services user-identification active-directory-access domain-controller status extensive domain

```
user@host> show services user-identification active-directory-access domain-controller status
extensive domain example-domain
Domain: example-domain
  Domain controller: example-domain10
    Address: 10.1.1.1
    Status: Disconnected
    Reason: Network issue
  Domain controller: example-domain20
```

```
  Address: 10.2.2.2
  Status: Disconnected
  Reason: Authentication failed
Domain controller: example-domain30
  Address: 10.3.3.3
  Status: Disconnected
  Reason: Host unreachable
```

show services user-identification active-directory-access statistics

Syntax show services user-identification active-directory-access statistics (ip-user-mapping | ip-user-probe | user-group-mapping) <domain *name*>

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Display statistics about IP address-to-user mapping, user-to-group mapping, and IP user probes used for the integrated user firewall feature. If two domains are configured, output is provided per domain.

Options

- **ip-user-mapping**—Number of total queries and failed queries to the event log on the domain controller for address-to-user mappings. Includes additional information, such as the log scan interval and the timestamp of the last event read.
- **ip-user-probe**—Number of total PC probes and failed probes.
- **user-group-mapping**—Number of total queries and failed queries to the LDAP server for user-to-group mappings
- **domain *name***—(Optional) Display the statistics for the specified domain.

Required Privilege Level view

Related Documentation

- clear services user-identification active-directory-access on page 456
- ip-user-mapping on page 350
- request services user-identification active-directory-access ip-user-probe on page 460
- user-group-mapping on page 428

List of Sample Output show services user-identification active-directory-access statistics ip-user-mapping on page 530
show services user-identification active-directory-access statistics ip-user-probe on page 531
show services user-identification active-directory-access statistics user-group-mapping on page 531

Output Fields Table 46 on page 529 lists the output fields for the **show services user-identification active-directory-access statistics ip-user-mapping** command.

Table 46: show services user-identification active-directory-access statistics ip-user-mapping Output Fields

Field Name	Field Description
Host	IP address of the domain controller.
Initial event log timespan	When the feature is first deployed, the number of previous hours for which the event log on the domain controller is read. A one means the last hour of the event log is read.

Table 46: show services user-identification active-directory-access statistics ip-user-mapping Output Fields *(continued)*

Field Name	Field Description
Eventlog scan interval	Number of seconds between event log scans.
Total log query number	Count of the queries on the event log.
Failed log query number	Count of the failed queries on the event log.
Log read number	Count of the times the event log was read.
Latest timestamp	Year:month:date:hours:minutes:seconds is the timestamp taken from the event log.

Table 47 on page 530 lists the output fields for the **show services user-identification active-directory-access statistics ip-user-probe** command.

Table 47: show services user-identification active-directory-access statistics ip-user-probe Output Fields

Field Name	Field Description
Total user probe number	Count of the probes of IP addresses to get IP address-to-user mappings.
Failed user probe number	Count of failed probe attempts.

Table 48 on page 530 lists the output fields for the **show services user-identification active-directory-access statistics user-group-mapping** command.

Table 48: show services user-identification active-directory-access statistics user-group-mapping Output Fields

Field Name	Field Description
Host	IP address and port being queried.
Total query number	Count of queries.
Failed query number	Count of failed query attempts.

Sample Output

show services user-identification active-directory-access statistics ip-user-mapping

```
user@host> show services user-identification active-directory-access statistics ip-user-mapping
Domain: example-domain1.com
     Host: 192.0.2.192
     Initial event log timespan : 1
     Eventlog scan interval : 60
     Total log query number : 240
     Failed log query number : 0
```

```
     Log read number : 838
     Latest timestamp :2013-10-11:15:11:54
     Host: 192.0.2.50
     Initial event log timespan : 1
     Eventlog scan interval : 60
     Total log query number : 273
     Failed log query number : 0
     Log read number : 2012
     Latest timestamp :2013-10-11:15:11:23
Domain: example-domain2.com
     Host: 192.0.2.39
     Initial event log timespan : 1
     Eventlog scan interval : 10
     Total log query number : 1596
     Failed log query number : 0
     Log read number : 6691
     Latest timestamp :2013-10-11:15:25:03
     Host: 192.0.2.1
     Initial event log timespan : 1
     Eventlog scan interval : 10
     Total log query number : 2628
     Failed log query number : 0
     Log read number : 114953
     Latest timestamp :2013-10-11:15:24:01
```

Sample Output

show services user-identification active-directory-access statistics ip-user-probe

```
user@host> show services user-identification active-directory-access statistics ip-user-probe
Domain: example-domain3.com
Total user probe number : 176116
Failed user probe number : 916
Domain: example-domain3.com
Total user probe number : 17632
Failed user probe number : 342
```

Sample Output

show services user-identification active-directory-access statistics user-group-mapping

```
user@host> show services user-identification active-directory-access statistics
user-group-mapping
Domain: example-domain3.com
Host: 192.0.2.1 Port  389
Total query number : 176116
Failed query number : 916
Domain: example-domain3.com
Host: 192.0.2.5 Port  389
Total query number : 8965
```

show services user-identification active-directory-access user-group-mapping

Syntax show services user-identification active-directory-access user-group-mapping (group *name* | status | user *name*) domain *name*

Release Information Command introduced in Junos OS Release 12.1X47-D10.

Description Display user-to-group mapping information used in the integrated user firewall feature. Note that the LDAP server is often part of the domain controller.

Options

- **group *group-name***—Display the users mapped to the specified group.
- **status**—Display the status of the last query to the LDAP server for user-group mapping.
- **user *name***—Display the groups for the specified username.
- **domain *name***—(Optional) Display the group, status, or user information for the specified domain.

Required Privilege Level view

Related Documentation

- LDAP Functionality in Integrated User Firewall on page 150
- user-group-mapping on page 428

List of Sample Output

show services user-identification active-directory-access user-group-mapping group domain on page 533
show services user-identification active-directory-access user-group-mapping status on page 533
show services user-identification active-directory-access user-group-mapping user on page 534

Output Fields Table 38 on page 510 lists the output fields for the **show services user-identification active-directory-access user-group-mapping group** command.

Table 49: show services user-identification active-directory-access user-group-mapping group Output Fields

Field Name	Field Description
Domain	Domain of the specified group.
Users	Usernames mapped to the specified group.

Table 39 on page 511 lists the output fields for the **show services user-identification active-directory-access user-group-mapping status** command.

Table 50: show services user-identification active-directory-access user-group-mapping status Output Fields

Field Name	Field Description
Domain	Domain for which the status is displayed.
LDAP server	IP address of the LDAP server.
Port	Port number on the LDAP server.
Last-query-status	Status of the last query from the SRX Series device.
Last-query-time	Year-month-date:hour:minutes:seconds when the SRX device last queried the LDAP server.

Table 40 on page 511 lists the output fields for the **show services user-identification active-directory-access user-group-mapping user** command.

Table 51: show services user-identification active-directory-access user-group-mapping user Output Fields

Field Name	Field Description
Domain controller	Domain controller about which the user information is displayed.
Groups	Groups to which the user belongs.
Referenced by policy	Groups to which the user belongs and that are referenced by a firewall policy.

Sample Output

show services user-identification active-directory-access user-group-mapping group domain

```
user@host> show services user-identification active-directory-access user-group-mapping group
finance domain www.apac-acme.net
show services user-identification active-directory-access user-group-mapping group
 finance-group
Domain: example-domain.net
Users: user1, user2
Domain: example2.domain.net
Users: user3
```

Sample Output

show services user-identification active-directory-access user-group-mapping status

```
user@host> show services user-identification active-directory-access user-group-mapping status
Domain: example-domain.net
LDAP server    Port      Last-query-status    Last-query-time
192.0.2.87      389       Query success        2014-02-07:15:50:52

Domain: example2.domain.net
LDAP server    Port      Last-query-status    Last-query-time
192.0.2.144     389       Idle                 0
```

Sample Output

show services user-identification active-directory-access user-group-mapping user

```
user@host> show services user-identification active-directory-access user-group-mapping user
user1
Domain example-domain.net
Groups: Dev, NAT, SBU
Referenced by policy: SBU
Domain: example2.domain.net
Groups: HR, USA
```