Extreme SLX-OS
Security Hardening Guide, 20.7.3

Supporting ExtremeRouting and ExtremeSwitching
SLX 9740, SLX 9640, SLX 9540, SLX 9250, SLX 9150,
Extreme 8820, Extreme 8720, and Extreme 8520

9039535-00 Rev AB
March 2026



Copyright © 2026 Extreme Networks, Inc. All rights reserved.

Legal Notice

Extreme Networks, Inc. reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information on Extreme Networks trademarks, see: https://www.extremenetworks.com/about-extreme-networks/company/legal/trademarks

Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses. End-user license agreements and open source declarations can be found at: https://www.extremenetworks.com/support/policies/open-source-declaration/

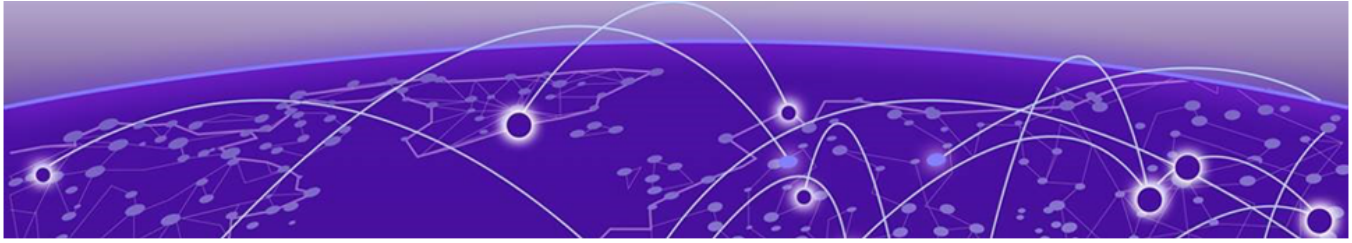


Table of Contents

Preface **5**
- Text Conventions 5
- Documentation and Training 6
  - Open Source Declarations 7
  - Training 7
- Help and Support 7
  - Subscribe to Product Announcements 8
- Send Feedback 8

About this document **9**
- What's New in this Document 9
- Supported Hardware 9

Security Hardening Guide **11**
- Security Hardening Guidance Overview 11
- Configure password policies 12
- Administrator lockout 12
- SSH Configuration 12
  - SSH ciphers 13
  - SSH MAC algorithms 13
  - SSH Key 13
  - SSH Key-exchange 14
  - SSH server timeout and login policies 14
  - Configuring SSH session re-key interval by volume and time 14
  - Re-keying by volume 14
  - Re-keying by time 15
  - Configure SSH authentication method 15
  - Disable telnet server 15
- Disable TLS versions 1.1 and earlier 15
- Enable authentication services 16
  - Enable HTTPS 16
  - Enable TLS for remote authentication services 16
  - Enable TLS for SYSLOG 17
- Disable unused remote authentication services 17
- Configure IP ACLs to block services 18
- Configure banners 19
  - Configure banners 19
- Support for RSA 4096 bit SSH hostkey 19
- Connlimit as an option for ip access-lists 20
- Version control for TLS 21
- Securing gNMI 22
- Mutual authentication support for TLS 23

To import CA of HTTPs client cert. 24
To import CA of GNMI client cert 24
Certificate expiry alert levels and period configuration 24
..... 24
User account expiry period configuration upon inactivity 25
Forcing default users password change 26
GRUB Bootloader Password Protection 27
Measured boot and Remote Attestation 28
Security Enhanced Linux (SE Linux) 28

Preface

Read the following topics to learn about:

- The meanings of text formats used in this document.
- Where you can find additional information and help.
- How to reach us with questions and comments.

Text Conventions

Unless otherwise noted, information in this document applies to all supported environments for the products in question. Exceptions, like command keywords associated with a specific software version, are identified in the text.

When a feature, function, or operation pertains to a specific hardware product, the product name is used. When features, functions, and operations are the same across an entire product family, such as Extreme Networks switches, the product is referred to as *the switch*.

Table 1: Notes and warnings

Icon	Notice type	Alerts you to...
	Tip	Helpful tips and notices for using the product
	Note	Useful information or instructions
	Important	Important features or instructions
	Caution	Risk of personal injury, system damage, or loss of data
	Warning	Risk of severe personal injury

Table 2: Text

Convention	Description
`screen displays`	This typeface indicates command syntax, or represents information as it is displayed on the screen.
The words *enter* and *type*	When you see the word *enter* in this guide, you must type something, and then press the Return or Enter key. Do not press the Return or Enter key when an instruction simply says *type*.
Key names	Key names are written in boldface, for example **Ctrl** or **Esc**. If you must press two or more keys simultaneously, the key names are linked with a plus sign (+). Example: Press **Ctrl**+**Alt**+**Del**
Words in italicized type	Italics emphasize a point or denote new terms at the place where they are defined in the text. Italics are also used when referring to publication titles.
NEW!	New information. In a PDF, this is searchable text.

Table 3: Command syntax

Convention	Description
bold text	Bold text indicates command names, keywords, and command options.
italic text	Italic text indicates variable content.
`[ ]`	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
`{ x | y | z }`	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
`x | y`	A vertical bar separates mutually exclusive elements.
`< >`	Nonprinting characters, such as passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*`[`*member*`...]`.
\	In command examples, the backslash indicates a “soft” line break. When a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Documentation and Training

Find Extreme Networks product information at the following locations:

Current Product Documentation
Release Notes
Hardware and Software Compatibility for Extreme Networks products
Extreme Optics Compatibility
Other Resources such as articles, white papers, and case studies

Open Source Declarations

Some software files have been licensed under certain open source licenses. Information is available on the Open Source Declaration page.

Training

Extreme Networks offers product training courses, both online and in person, as well as specialized certifications. For details, visit the Extreme Networks Training page.

Help and Support

If you require assistance, contact Extreme Networks using one of the following methods:

Extreme Portal

Search the GTAC (Global Technical Assistance Center) knowledge base; manage support cases and service contracts; download software; and obtain product licensing, training, and certifications.

The Hub

A forum for Extreme Networks customers to connect with one another, answer questions, and share ideas and feedback. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.

Call GTAC

For immediate support: (800) 998 2408 (toll-free in U.S. and Canada) or 1 (408) 579 2800. For the support phone number in your country, visit www.extremenetworks.com/support/contact.

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number, or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any actions already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

Subscribe to Product Announcements

You can subscribe to email notifications for product and software release announcements, Field Notices, and Vulnerability Notices.

1. Go to The Hub.
2. In the list of categories, expand the **Product Announcements** list.
3. Select a product for which you would like to receive notifications.
4. Select **Subscribe**.
5. To select additional products, return to the **Product Announcements** list and repeat steps 3 and 4.

You can modify your product selections or unsubscribe at any time.

Send Feedback

The User Enablement team at Extreme Networks has made every effort to ensure that this document is accurate, complete, and easy to use. We strive to improve our documentation to help you in your work, so we want to hear from you. We welcome all feedback, but we especially want to know about:

- Content errors, or confusing or conflicting information.
- Improvements that would help you find relevant information.
- Broken links or usability issues.

To send feedback, email us at Product-Documentation@extremenetworks.com.

Provide as much detail as possible including the publication title, topic heading, and page number (if applicable), along with your comments and suggestions for improvement.

About this document

What's New in this Document on page 9
Supported Hardware on page 9

What's New in this Document

This document is released with the SLX-OS 20.7.3 software release.

The following table includes descriptions of changes made to this document for the SLX-OS 20.7.3 software release.

Table 4: Changes made in Rev AB of this document.

Feature	Description	Described in
SSH Configuration	Updated this topic with the latest recommended best practices for configuring keys, algorithms, and ciphers for enhancing security.	SSH Configuration on page 12

Table 5: Changes made in Rev AA of this document.

Feature	Description	Described in
TLS	TLS v 1.1 is no longer supported from SLX-OS 20.7.3 onwards. Removed references to TLS v 1.1 from this topic.	Version control for TLS on page 21
Banner Configuration	• Added that MOTD will also be applicable for SSH sessions. • Added that a single line MOTD string must be terminated with a new line character.	Configure banners on page 19

For additional information, refer to the *Extreme SLX-OS Release Notes* for this version.

Supported Hardware

SLX-OS 20.7.3 supports the following hardware platforms.

- Extreme 8820
- Extreme 8720

- Extreme 8520
- ExtremeSwitching SLX 9540
- ExtremeSwitching SLX 9250
- ExtremeSwitching SLX 9150
- ExtremeRouting SLX 9740
- ExtremeRouting SLX 9640

Note

All configurations and software features that are applicable to SLX 9150 and SLX 9250 devices are also applicable for the Extreme 8520 and Extreme 8720 devices respectively.

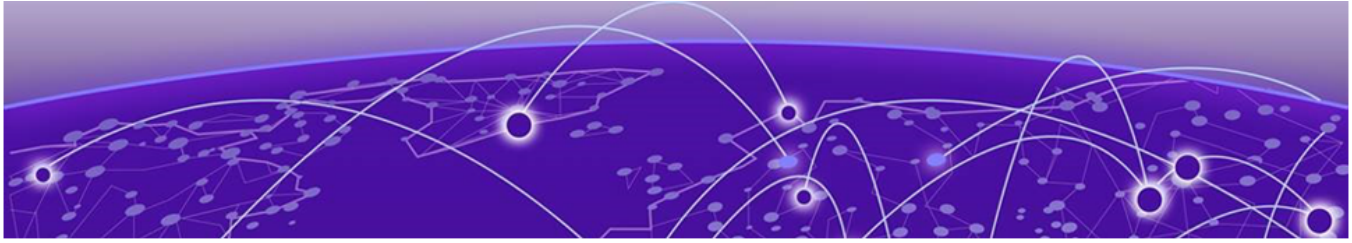
All configurations and software features that are applicable to SLX 9740 devices are also applicable for the Extreme 8820 devices.

The "Measured Boot with Remote Attestation" feature is only applicable to the Extreme 8520, Extreme 8720, and Extreme 8820 devices. It is not supported on the SLX 9150 and SLX 9250 devices.

Note

Although many software and hardware configurations are tested and supported for this release, documenting all possible configurations and scenarios is beyond this document's scope.

For information about other releases, see the documentation for those releases.



Security Hardening Guide

Security Hardening Guidance Overview on page 11
Configure password policies on page 12
Administrator lockout on page 12
SSH Configuration on page 12
Disable TLS versions 1.1 and earlier on page 15
Enable authentication services on page 16
Disable unused remote authentication services on page 17
Configure IP ACLs to block services on page 18
Configure banners on page 19
Support for RSA 4096 bit SSH hostkey on page 19
Connlimit as an option for ip access-lists on page 20
Version control for TLS on page 21
Securing gNMI on page 22
Mutual authentication support for TLS on page 23
Certificate expiry alert levels and period configuration on page 24
User account expiry period configuration upon inactivity on page 25
Forcing default users password change on page 26
GRUB Bootloader Password Protection on page 27
Measured boot and Remote Attestation on page 28
Security Enhanced Linux (SE Linux) on page 28

Security Hardening Guidance Overview

Device hardening steps are performed by a user with administrative privileges. Specific hardening actions may or may not be appropriate for a given environment and must be considered in the context of the overall security policy and existing physical and procedural controls.

The NetworkOS device management functions are isolated through authentication. Once administrators login with specific credentials, their access is limited to commands for which they have privileges with role-based permissions. Additionally, network management communication paths are protected against modification and disclosure using SSHv2. The audit channel to an external Syslog server is protected using TLS encapsulation.

Configure password policies

The minimum password strength and configurable attributes are recommended that includes minimum length, character sets, with the number of retries when logging in. This record details on how long an account can be locked out when the maximum number of login failures is observed.

An example password policy configuration:

```
device(config)# password-attributes min-length 8
device(config)# password-attributes max-retry 4
device(config)# password-attributes max-lockout-duration 5000
device(config)# password-attributes character-restriction upper 1
device(config)# password-attributes character-restriction lower 2
device(config)# password-attributes character-restriction numeric 1
device(config)# password-attributes character-restriction special-char 1
```

The default password encryption policy is Encryption Level 10, which utilizes salted SHA512 for password storage.

Refer to the topic *Password Policies* in the *Extreme SLX-OS Security Configuration Guide* for the current version for further details.

Administrator lockout

By default, the administrator is not locked out of the device even after `max-retry` failures. To lock the administrator out, execute the below command:

```
device(config)# password-attributes admin-lockout
```

When the administrator is locked-out, the device allows access for the administrator after the value set for `max-lockout-duration` has elapsed.

Note
The administrator logins over the serial port/console, which is never locked out and can login over the network again only if the admin-lockout password attribute is disabled.

To allow the administrator to login over the network and disable administrator lockout execute the below commands:

```
device# configure terminal

device(config)# no password-attributes admin-lockout
```

Refer to the topic *Password Policies* in the Extreme SLX-OS Security Configuration Guide for the current version for further details.

SSH Configuration

SSH provides a large number of ciphers and MAC algorithms for use. Extreme recommends a sub-set of these ciphers and MAC algorithms.

The following are the recommended SSH ciphers, MAC Algorithms, SSH Key, SSH Key Exchange Algorithms and other configuration parameters for enhanced security.

For more information on all the ciphers, algorithms, and keys available for use with SSH, see the topic *Feature support for SSH* in the Extreme SLX-OS Management Configuration Guide for the current version.

SSH ciphers

The following ciphers are recommended for the SSH clients and SSH servers:

- `aes128-ctr`
- `aes192-ctr`
- `aes256-ctr`
- `aes128-gcm@openssh.com`
- `aes256-gcm@openssh.com`

Refer to the topic *Configure SSH Ciphers* in the Extreme SLX-OS Security Configuration Guide for the current version for specific guidance on configuring these SSH ciphers.

SSH MAC algorithms

The following MAC algorithm is recommended for SSH clients and SSH servers:

- `umac-128-etm@openssh.com`

Refer to the topic *Configure SSH MAC* in the Extreme SLX-OS Security Configuration Guide for the current version for specific guidance on configuring these SSH MAC algorithms.

SSH Key

The following is the recommended SSH Key:

- `rsa 4096`



Important

Remove the default DSA and ECDSA keys before applying the recommended SSH Key.

```
SLX (config)# no ssh server key dsa
                SLX (config)# no ssh server key ecdsa
                SLX (config)# ssh server key rsa 4096
                    SLX (config)#
```

SSH Key-exchange

The following MAC algorithms are recommended for SSH Key-exchange on SSH clients and SSH servers:

- `curve25519-sha256`
- `curve25519-sha256@libssh.org`
- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group16-sha512`
- `diffie-hellman-group18-sha512`

Refer to the topic *Configure SSH Key* in the Extreme SLX-OS Security Configuration Guide for the current version, for specific guidance on configuring SSH Key-exchange algorithms.

SSH server timeout and login policies

Enter the `ssh server max-idle-timeout` command to set the timeout value for SSH connections to the server. This setting affects `ssh` connections to the server including the `netconf` sessions.

```
device(config)# ssh server max-idle-timeout 20
```

Enter the **sshserver max-auth-tries** command to set the number of login attempts

```
device(config)# ssh server max-auth-tries 2
```

Enter the **sshserver max-login-timeout** command to set the login timeout. Set the value to an appropriate timeout period in the administrator's environment.

```
device(config)# ssh server max-login-timeout 30
```

Configuring SSH session re-key interval by volume and time

The SSH servers can trigger re-keying once a certain time interval is reached or data traffic reaches a specified volume. During re-keying, a set of key exchange messages are transferred between the SSH clients and the server, changing the key used for the session security.

Re-keying by volume

The **re-key-volume** option cannot exceed a value equal to `1024` MB. The default value is `1024` MB. The range of the re-key volume configured using the **ssh-server** command is `512` to `1024` MB.

```
device(config)# ssh server rekey-volume ?
                Possible completions:
                    <DECIMAL> <512-4095> Megabytes
```

Re-keying by time

The SSH re-key can also be configured based on time. The default value is `3600` seconds. The following command is used to specify the time.

```
device(config)# ssh server rekey-interval ?
            Possible completions:
                <DECIMAL> <900-3600> Seconds
```

Configure SSH authentication method

The SSH provides public key and password authentication methods, including support for `X.509 v3` certificates.

To use SSH public-key authentication, enter the **`certutil import sshkey directory`** *pubkey-directory* **file** *filename* **protocol SCP host** *remote-ip* **user** *user-account password password* command to import the public key.

```
device# certutil import sshkey user admin host 10.70.4.106
            directory /users/home40/bmeenaks/.ssh file id_rsa.pub login fvt
            Password: ***********

            2012/11/14-10:28:58, [SEC-3050], 75,, INFO, VDX,
            Event: sshutil, Status: success, Info: Imported SSH public key from
10.70.4.106 for
                user 'admin'.
```

To support password less SSH authentication, externally generated key pairs using `RSA-2048`.

Refer to the topic *Secure Shell* in the Extreme SLX-OS Security Configuration Guide for the current version for further guidance on configuring SSH authentication method.

Disable telnet server

Enter the **telnet server shutdown** command in global configuration mode to disable the Telnet server.

```
device(config)# telnet server shutdown
```

Disable TLS versions 1.1 and earlier

To disable support for TLS versions 1.1 and earlier, do the following:

1. SSH to the system and acquire a root shell:

   ```
   SLX# start shell
   Entering Linux shell for the user: admin
   [admin@SLX]# su -
   Password:
   [root@SLX]#
   ```

2. Edit the `Apache webserver` config located at '`/fabos/webtools/bin/web.conf.0`' and replace the line that contains the '`SSLProtocol`' variable with the following:

   ```
   SSLProtocol -all +TLSv1.2
   ```

3. Grep the process table to look for `httpd` processes and kill the lowest numbered one (first in the list).

```
# ps axuww | grep httpd
nobody    5046  0.0  0.0  88956  4220 ?        S    20:32   0:00
/usr/sbin/httpd.0 -DSSL -f /fabos/webtools/bin/httpd.conf.0
root     24164  0.0  0.0  88688  6360 ?        Ss   01:59   0:14
/usr/sbin/httpd.0 -DSSL -f /fabos/webtools/bin/httpd.conf.0
nobody   29385  0.0  0.0  88956  4220 ?        S    19:22   0:00
/usr/sbin/httpd.0 -DSSL -f /fabos/webtools/bin/httpd.conf.0

# kill 5046
```

   In the above example, the *httpd* process with the lowest PID number is killed.

4. Restart `Apache` by manually executing the following command:

```
# /usr/sbin/httpd.0 -DSSL -f /fabos/webtools/bin/httpd.conf.0
```

   After this step, `SLX-OS` will run `Apache` with TLS version 1.1 and below disabled.



Note

The **`httpd.conf.0`** file includes the **`web.conf.0`** file automatically and there is no persistent change across reboots. However, this will be fixed in future SLX-OS release.

Enable authentication services

Enable HTTPS

Refer to the topic *HTTPS Certificate* in the *Extreme SLX-OS Security Configuration Guide* for the current version for specific guidance on installing certificates and enabling HTTPS.

Enable TLS for remote authentication services

RADIUS over TLS and LDAP over TLS are supported.

Refer to the topic *RADIUS Server Authentication* in the *Extreme SLX-OS Security Configuration Guide* for the current version for specific guidance on configuring RADIUS over TLS.

Refer to the topic *Lightweight Directory Access Protocol* in the *Extreme SLX-OS Security Configuration Guide* for the current version for specific guidance on configuring LDAP over TLS.

Enable TLS for SYSLOG

To enable secure logging using the `syslog` server, complete the following steps.

1. Enter the **crypto import syslogca** command in privileged EXEC mode to import the syslog CA certificate.

```
device# crypto import syslogca rbridge-id 1
    protocol SCP host 10.2.2.101 directory /home/certs/ file
    chainCA02.cert.pem user admin password <password>
```

   The CA certificate imported must be generated using `RSA-2048` with `SHA-256`.

2. Enter the **logging syslog-server** *ip-address* command in global configuration mode to configure the syslog server.

```
device(config)# logging syslog-server 10.20.238.120
    secure port 1999
```

The device enforces certificate validation during import and TLS server certificate validation occurs during the TLS handshake according to the following rules:

- Certificate validation and the certificate path validation support a minimum path length of three certificates.
- The certificate path must terminate with a trusted CA certificate.
- The certificate path should be validated by verifying the presence of the `basic Constraints`extension and that the CA flagis setto TRUE for all CA certificates.
- The revocation status of the certificate should be validated.
- For `SYSLOG`, the device currently requires that an IP address must be used for Common Name (CN) and Subject Alternative Name (SAN).
- The `extendedKeyUsage` field should be validated according to the following rules:
  - Certificates used for trusted updates and executable code integrity verification should have the Code Signing purpose `(id- kp 3 with OID 1.3.6.1.5.5.7.3.3)` in the `extended Key Usage`field.
  - Server certificates presented for TLS should have the Server Authentication purpose `(id-kp 1 with OID 1.3.6.1.5.5.7.3.1)` in the `extended Key Usage` field.
  - Client certificates presented for TLS should have the Client Authentication purpose (`id-kp 2 with OID 1.3.6.1.5.5.7.3.2`) in the `extended Key Usage` field.
  - OCSP certificates presented for OCSP responses should have the OCSP Signing purpose (`id-kp 9 withOID 1.3.6.1.5.5.7.3.9`) in the `extended Key Usage` field.
  - A certificate should only be treated as a CA certificate if the `basic Constraints` extension is present and the CA flag is set to TRUE.

Disable unused remote authentication services

To disable unused remote authentication services, do the following:

Enter the **no tacacs-server** command to remove any TACACS + server configuration.

```
device(config)# no tacacs-server <host>
```

Enter the **no radius-server** command to remove any RADIUS server configuration.

```
device(config)# no radius-server <host>
```

Enter the **no ldap-server** command to remove any LDAP server configuration.

```
device(config)# no ldap-server <host>
```

Configure IP ACLs to block services

Use IP ACLs to block Telnet, HTTP, and Extreme internal ports `7110, 7710, 8008, 9110,` and `9710` for `IPv4` and `IPv6` address families.

If SSH access is required, use the **seq permit** command to allow access on port 22.

If remote access is required, such as through SCP or LDAP, use the **seq permit** commands to allow UDP and TCP traffic on ports `1024` through `65535`.

Configure IP ACLs usingthe **ip access-list** command and use the **ip access-group** command to apply the rules to the management interface.

This example shows the configuration of an extended ACL that blocks some ports used for the various services.

```
device(config)# ip access-list extended ccextACL
device(config-ip-ext)# seq 1 deny tcp any any eq 23
device(config-ip-ext)#seq 2 deny tcp any any eq 80
device(config-ip-ext)#seq 5 deny tcp any any eq 7110
device(config-ip-ext)#seq 6 deny tcp any any eq 7710
device(config-ip-ext)#seq 7 deny tcp any any eq 8008
device(config-ip-ext)#seq 8 deny tcp any any eq 9110
device(config-ip-ext)#seq 9 deny tcp any any eq 9710
device(config-ip-ext)#seq 11 permit tcp any any range 1024 65535
device(config-ip-ext)#seq 12 permit udp any any range 1024 65535
device(config-ip-ext)#seq 13 permit tcp any any eq 22
device(config-ip-ext)#seq 14 permit tcp any any eq 830
device(config-ip-ext)#exit
```

This configuration step shows how to apply the above ACL to the IPv4 address family.

```
device(config)# interface management 1/0
device(config-Management-1/0)# ip access-group ccextACL in
```

This configuration step shows how to apply the above ACL to the IPv6 address family.

```
device(config)# interface management 1/0
device(config-Management-1/0)# ipv6 access-group ccextACL6 in
```

Refer to the topic *ACLs* in the *Extreme SLX-OS Security Configuration Guide* for the current version for further details.

Configure banners

Configure banners

Configure banners

The commands below are used to configure banner messages. The banner messages are used to provide information to the user when the device is accessed. There are three commands that can be used to setup banner messages as explained below:

- **banner incoming** : Sets the incoming banner message. The message is seen on the console when a user accesses the device.
- **banner motd** : Sets the message of the day banner. The message is displayed when the device receives a login request and before the Telnet or SSH session is established.
- **banner login** : Sets the switch banner and the message is displayed after the user is authenticated.

Banner length is from `1 – 2048` characters, which can be issued as a single line of text, or in multiline mode by pressing `esc m`.

Note

When your banner message is short and is displayed within one line, terminate the message with the **\n** new line character within double quotes (" ").

Each of these commands can be invoked from the `CLI` in Privileged EXEC Mode as shown below:

```
SLX # configure terminal
SLX (config)# banner incoming <message>
SLX (config)# no banner incoming
SLX (config)# banner motd <message>
SLX (config)# no banner motd
SLX (config)# banner login <message>
SLX (config)# no banner login
```

Support for RSA 4096 bit SSH hostkey

SLX-OS provides support for the strongest of various RSA hostkeys for SSH, which is `4096` bits. Now you can configure `SSH RSA` hostkey `4096` bit using below option.

```
SLX(config)# ssh server key rsa ?
Possible completions:
[2048]
1024 1024 bits RSA key
2048 2048 bits RSA key [default]
4096 4096 bits RSA key
SLX(config)# ssh server key rsa 4096
```

The default **RSA** hostkey for **SSH** when the above hostkey is not configured, is **2048** bits.

SLX-OS supports the following **SSH** server hostkey algorithms **RSA**, **ECDSA P256** and **DSA** to be configured. For maximum security, it is recommended to use either **ECDSA** or **RSA**(minimum **2048** bits) as the hostkeys. **DSA** and **RSA 1024** are both insecure with **1024** bit key length.

SLX-OS's `OpenSSh` server sends **ECDSA** key as the *hostkey* if available, since *ECDSA* is the strongest available algorithm on SLX-OS.

On a LINUX client, the SSH client receives the following message to accept the hostkey sent by SLX-OS.

```
The authenticity of host 10.24.12.129 (10.24.12.129) can't be established.
ECDSA key fingerprint is SHA256:LlgBLdBedpJ1AU6GWa4OYjtye6JM4CfR8i8k2SwGOfw.
Are you sure you want to continue connecting yes/no ?
```

If you remove `ECDSA` hostkey configured from `SSH` server key `CLI` , then the `OpenSSH` server in SLX-OS negotiates **RSA** hostkey based on the bit length, which you configured using **ssh server key rsa** CLI.

The default being `2048` bits. Hence, you need to explicitly configure **ssh** server key **RSA 4096** to use the **RSA 4096** bit hostkey and remove **ECDSA** if it does not consider, so that the server sends **RSA 4096** as the hostkey.

Connlimit as an option for ip access-lists

Management ACL configurations provide `connlimit` as an option to restrict the number of connections from a given host, using a specific protocol or application port.

For example, the below configuration, when applied to the management port restricts the number of tcp connections to the `SLX` mgmt. IP from the given client IP to N, where N can range between *`1-65536`*.

```
SLX(config)#ip access-list extended check
SLX(conf-ipacl-ext)# permit tcp host <client IP> host <SLX mgmtIP> connlimit <N>
```

The protocol option in the above example is specified as **udp**, which restricts **udp** connections. Similarly, to restrict connections for both **tcp** and **udp** use **ip** as the protocol name in the command.

Use the **any** option to filter connections on specified protocols from any IP address.

```
SLX(conf-ipacl-ext)# permit ip any any connlimit N
SLX(conf-ipacl-ext)# permit tcp any any connlimit N
```

Note
Only incoming connections on management port can be restricted using the `connlimit` option.

When you include application ports to the access-list with *connlimit* option, you can restrict the number of connections using a specific application protocols and allow rest of the traffic.

The following example command restricts number of SSH connections, which uses `port 22` from the specified client IP.

```
SLX(conf-ipacl-ext)# permit tcp host <client IP> host <SLX mgmtIP> eq 22 connlimit N.
```

Restricting `connlimits` to application protocols can be a highly useful when mitigating `DDOS` attacks, as it prevents access from malicious clients.

For example, the `HTTP/HTTPs` service in SLX has a restriction of allowing maximum of 30 parallel REST connections to SLX-OS.

A DDOS attack scenario occurs when a buggy client, which does not close its sockets but monitor SLX-OS heart beat, but sends periodic unauthorized REST requests to SLX-OS at a rapid rate. This exhausts the 30 connections rapidly and denies other operational REST client access to SLX-OS till the socket state transitions cleans up these orphaned connections.

As shown below, `connlimit` can be applied to prevent such a `DDOS` attack by malicious clients by restricting the number of simultaneous connections. If the maximum operational REST client connections is known, it can be configured to restrict access to SLX-OS.

In the following example, *<client IP 1>* is the operational client and *<client IP 2>* is the heartbeat monitor. The example shows the configuration of restricting their concurrent connections to a maximum of 10 each.

```
SLX(conf-ipacl-ext)# permit tcp host <client IP 1> host <SLX mgmtIP> eq https connlimit 10
SLX(conf-ipacl-ext)# permit tcp host <client IP 2> host <SLX mgmtIP> eq https connlimit 10
```

To use the `connlimit` options with an access-list, it may be required that the administrator first configures a `permit ip` rule with `any any` option to allow other traffic without disruption. This is because of the order of IP table rules that are added by default in SLX-OS.

To understand IP tables order and how to use `connlimit` effectively and its limitations, refer to the topic *ACL* in the *Extreme SLX-OS Security Configuration Guide* for the current version.



Note

Administrators, do not use `connlimit` with generic protocols like `tcp`, `udp` and `ip` with the `any any` option, unless you are very familiar with management connections to the SLX-OS to prevent possible disruption of traffic that is not intended to be restricted.

Version control for TLS

Administrator can configure the minimum TLS protocol version to be used by SLX-OS manageability applications that use TLS either as a client or server. SLX-OS provides separate TLS version control options for TLS clients and servers. The applications that act as TLS clients in SLX are `SYSLOG`, `RADIUS` and `LDAP`.

The TLS servers of SLX-OS management plane are HTTPs and secure GNMI. The control knobs are like below.

```
ssl-profile-server)# tls min-version ?
Possible completions:
```

```
    <1.2|1.3> specify TLS version

SLX(mgmt-sec-ssl-profile-server)# tls min-version 1.2
SLX(mgmt-sec-ssl-profile-server)# exit
SLX(mgmt-security)# ssl-profile ?
Possible completions:
client       management security ssl profile client for tls configuration
server       management security ssl profile server for tls configuration

SLX(mgmt-security)# ssl-profile client
SLX(mgmt-sec-ssl-profile-client)# tls ?
Possible completions:
min-version       min version to be supported by client

SLX(mgmt-sec-ssl-profile-client)# tls min-version ?

Possible completions:
    <1.2|1.3> specify TLS version
SLX(mgmt-sec-ssl-profile-client)# tls min-version 1.3
SLX(mgmt-sec-ssl-profile-client)# end
SLX#
```

Setting the minimum version to TLS v1.3 for the client profile forces TLS clients to send only TLS v1.3 version in its client hello packet as TLS v1.3 is the max supported TLS version in SLX-OS. In case the server negotiates a lesser secure version, the SLX-OS breaks the handshake upon receiving the server hello.

The below example is of the audit log that appears when the handshake is broken where the **`show logging audit`** command output indicates the insecure version that was negotiated by the server.

```
63 AUDIT, 2025/05/17-16:25:24 (GMT), [SEC-3111], INFO, SECURITY,
NONE/root/NONE/None/CLI,, SLX, Event: TLS SESSION, TLS handshake,
Info: server version 1.2 is lesser than client min-version 1.3 TLS handshake failed.
```

Similarly, setting the minimum version to `TLS v1.3` for the server profile forces TLS servers in SLX-OS to break the handshake upon receiving a client `hello` with less secure TLS version.

The below example is of the audit log that appears when the handshake is broken where the **`show logging audit`** command output indicates the insecure version that was sent by the client.

```
63 AUDIT, 2025/05/17-16:25:24 (GMT), [SEC-3111], INFO, SECURITY,
NONE/root/NONE/None/CLI,, SLX, Event: TLS SESSION, TLS handshake,
Info: client version 1.2 is lesser than server min-version 1.3 TLS handshake failed.
```

Securing gNMI

gRPC Network Management Interface (gNMI) is a network management protocol that supports modification and retrieval of configuration, as well as control and transmission telemetry streams from a network element to a data collection system.

SLX-OS supports using TLS to protect gNMI traffic. The following configuration example enables using TLS to protect gNMI traffic.

```
SLX(config-gNMI-server)# secure-port <port number>
```

Here the port number can vary from `1024 to 49151`. When this configuration is applied, gNMI runs over TLS using the configured port and client get connected to this port to make a TLS connection.

The administrator must use a gNMI client that has TLS support and configure it for the same.

Note
Removing the above configuration makes `gNMI` to switch to the default non-secure mode, and will listen on the non-secure default port `9339`.

On a SLX-OS device, which is the gNMI server, the gNMI server certificate and the private key signing it can be imported to the switch using `pkcs12` format just like when the HTTPs certificate and key are imported.

The following command option is provided for the same, where the certificate and the key is encrypted into `pkcs12` format file on a trusted external server and imported from that server.

```
SLX# crypto ca import-pkcs type pkcs12 cert-type gNMI-server directory
    <dir-name> file <file-name> host <host-name/ip> protocol <SCP|FTP>
    user <server-username> password <server-password> pkcs-passphrase
    <pkcs export password>
```

Mutual authentication support for TLS

SLX-OS servers are enabled with the ability to present, receive and validate client certificates to either authenticate themselves or authenticate a remote client presenting its certificate to the SLX-OS server.

Since the TLS clients on SLX-OS are **`syslog`**, **`RADIUS`**, and **`LDAP`** the two command options that are provided to import the `pkcs12` format of the client certificates and the private key signing it.

```
SLX# crypto ca import-pkcs type pkcs12
    cert-type <ldap-client/radiusclient/syslog-client>
    directory <dir-name> file <file-name> host <host-name/ip>
    protocol <SCP|FTP> user <server username> password <server-password>
    pkcs-passphrase <pkcs export password>
```

When the client certificate is imported to SLX via above command for each of the services, connecting to their servers via secure port that sends client certificates to the server only if the server requests for client certificate. Enabling the server to send client certificate request is external server configuration and not in the scope of the current document.

The TLS servers in SLX-OS are HTTPs and secure gNMI. These servers must authenticate their external clients when the latter present their client certificates. For this, the TLS servers of SLX-OS must send Client Certificate Request in the TLS handshake.

To validate the incoming client certificate against a trusted authority during the TLS handshake, a CA certificate for the Client Certificate must be imported to SLX-OS. The CA can be imported for both HTTPs and secure gNMI using the below command.

To import CA of HTTPs client cert.

```
SLX#crypto import httpsclientca
    directory <dir-name> file <file-name> host <host-name/ip>
    protocol <SCP|FTP> user <server-username> password
    <server-password>
```

To import CA of GNMI client cert.

```
SLX#crypto import gnmiclientca directory
    <dir-name> file <file-name> host <host-name/ip> protocol
    <SCP|FTP> user <server-username>
    password<server-password>
```

Importing the CA by using the above commands acts as a control knob for turning on mutual authentication and enables these services to request client certificate from clients during TLS handshake. When the client certificate is requested, clients must present their client certificates issued by the imported CA.



Note
Removing the imported CA's using the **no** form of the above commands disables mutual authentication for the respective services.

Enabling mutual authentication enhances security by preventing a *Man-In-The-Middle* attack from imposing clients, which fail to identify themselves to SLX-OS or to establish SLX-OS as a trusted client to TLS servers seeking client authentication.

Certificate expiry alert levels and period configuration

Expiry of TLS certificates can cause disruptions. A mechanism is provide to alert administrators to the expiry by generating an alarm before the certificate expiry.

Alarms with 4 levels of severity can be generated and can be configured individually. These 4 levels are *Critical*, *major*, *Minor*, and *Info*. Each of these severity alarms can be configured to activate a set number of days before the TLS certificate expires.

The following example configuration shows how to configure the various alarms.

```
SLX(config)# crypto cert expiry-level info period 50
SLX(config)# crypto cert expiry-level minor period 30
SLX(config)# crypto cert expiry-level major period 10
SLX(config)# crypto cert expiry-level critical period 5
```

The *expiry-level* configuration allows period to be specified in the range between `1 to 90` in number of days. When configured, periodically, once in 24 hours, the expiry date of all TLS certificates present in SLX-OS are checked. When the number of days remaining for expiry of a certain certificate matches the period configured, an alert is issued with the severity indicating the level configured in the *expiry-level* field.

The alert is issued in the form of a RASLOG. On SLX-OS, SNMP trap severity levels can be set. Upon setting the SNMP trap severity level to warning, the generated RASLOGs will also issue an SNMP trap.

The RASLOGs and SNMP traps carries detail about the expired certificate like the serial number of the certificate, its subject, and other details. It also displays that this certificate expires within these number days.

The following is an example when info level is configured.

RASLOG

```
2022/05/13-00:00:02, [SEC-3136], 87,, WARNING, SLX, Event: cert expiry,
    Alert-level:INFO, Certificate Details=[subject=
    /C=IN/ST=KA/L=BAN/O=HCL/OU=Engg/CN=10.24.12.129/emailAddress=test@test.com
    issuer= /C=IN/ST=KA/L=BAN/O=HCL/OU=Engg/CN=root serial=4098] will expire in 44 days.
```

SNMP Trap

```
05:35:32.203670 IP 10.24.12.129.50000 > ldap.testsqa.com.SNMPtrap:
    C="cm2" V2Trap(452) system.sysUpTime.0=81400
    S:1.1.4.1.0=E:1588.2.1.1.1.0.4 S:18.1.3.0=10.24.12.129
    E:1588.2.1.1.1.8.5.1.1.87=87 E:1588.2.1.1.1.8.5.1.2.87="2022/05/13-
    00:00:02" E:1588.2.1.1.1.8.5.1.3.87=3 E:1588.2.1.1.1.8.5.1.4.87=1
    E:1588.2.1.1.1.8.5.1.5.87="SEC-3136 Event: cert expiry , Alertlevel:
    INFO, Certificate Details=[subject=
    /C=IN/ST=KA/L=BAN/O=HCL/OU=Engg/CN=10.24.12.129/emailAddress=test@test.com
    issuer= /C=IN/ST=KA/L=BAN/O=HCL/OU=Engg/CN=root
    serial=4098] will expire in 44 days."
```

In case administrator has configured multiple levels or all 4 levels, then many alerts are issued indicating the particular severity level when the period remaining for expiry matches the configured period against each level.

In case a certificate is already expired a raslog with **`Error as severity level`** is sent continuously sent every 24 hours until the specific certificate is changed.

This RASLOG is sent irrespective of the expiry level configuration. Having the above configurations provides reminders to administrators to change the certificate and prevents a service from non-functional due to TLS handshake failure resulting from certificate expiry.

User account expiry period configuration upon inactivity

Keeping track of inactive *user* accounts and locking them from accessing the system enhances the security of the whole network. SLX-OS provides a configuration that enables administrators to set a period after which inactive user accounts are locked out of the system. *root* and *default admin* accounts are never locked out.

A locked account must be explicitly unlocked by the device's administrator.

Inactive account configuration is done for each user account when that account is created. Use the **`username`** command and its **`acct-inactivity-expiry-period`** option to set the duration after which the account becomes inactive. To generate a warning to

the user, that the account will become inactive, use the **username** command and the **acct-inactivity-warning-period** option.

If the user logs in within the period set in the *acct-inactivity-expiry-period* configuration, the count is reset to zero (0).

The following example shows the configuration for creating the user account *test*, its inactivity period is set to 40 days, and the warning period is set to 20 days.

```
SLX(config)#username test acct-inactivity-expiry-period 40
                    acct-inactivity-warning-period 20 password xyz@12345 role admin
```

The range for inactivity expiry period for an account can be configured between 1 to 180 days and the range for inactivity warning period can be between 1 to 120 days.

Note
These configurations cannot be set for the *root* and *default admin* accounts. However, this configuration can be set for the *default user account* or any other account.

The user account expiry warning RASLOG is generated only once when the user does not login for the specified inactivity warning period.

Once the inactive user account expires after the specified inactivity expiry period, an error RASLOG indicating the expiry is sent every 24 hours. If the user configures SNMP trap severity level to warning, a *SNMP* trap will also be generated with these RASLOGs.

RASLOG Example

```
2021/03/04-09:50:00, [SEC-3138], 3445,, WARNING, SLX, Event: user
    inactivity warning USER test will expire in 25 days.
    2021/03/15-09:51:49, [SEC-3139], 3448,, ERROR, SLX, Event: user
    expired USER test expired 12 days ago.
```

SNMP Trap Example

```
03:27:30.135220 IP 10.24.15.197.50000 > ldap.testsqa.com.SNMPtrap:
    C="cm1" Trap(276) E:1588.2.1.1.1 10.24.15.197 enterpriseSpecific s=4
    365800 S:18.1.3.0=10.24.15.197 E:1588.2.1.1.1.8.5.1.1.1918=1918
    E:1588.2.1.1.1.8.5.1.2.1918="2020/12/26-02:53:09"
    E:1588.2.1.1.1.8.5.1.3.1918=3 E:1588.2.1.1.1.8.5.1.4.1918=1
    E:1588.2.1.1.1.8.5.1.5.1918="SEC-3138 Event: user inactivity warning, USER user will
expire in 2 days."
03:27:30.313334 IP 10.24.15.197.50000 > ldap.testsqa.com.SNMPtrap:
    C="cm1" Trap(246) E:1588.2.1.1.1 10.24.15.197 enterpriseSpecific s=4
    365800 S:18.1.3.0=10.24.15.197 E:1588.2.1.1.1.8.5.1.1.1919=1919
    E:1588.2.1.1.1.8.5.1.2.1919="2020/12/26-02:53:09"
    E:1588.2.1.1.1.8.5.1.3.1919=2 E:1588.2.1.1.1.8.5.1.4.1919=1
    E:1588.2.1.1.1.8.5.1.5.1919="SEC-3139 Event: user expired USER Extuser expired 3 days
ago."
```

Forcing default users password change

Forcing the default SLX-OS users, *root*, *admin*, and *user* to change their password at first login significantly enhances the security of the SLX-OS device.

The default password for default admin and default user are `default config options` and the default password for `root user` is present in the factory settings of the device.

Note
The user is not allowed to login without changing the password upon first login for these users when this configuration is present.

The configuration to enable default user password change is as below.

```
SLX(config)# password-attributes force-default-password-change
```

Forcing users to change their password is a well know security practice to reduce the surface of attack on a network. SLX-OS provides administrators the ability to force the uses to change their passwords periodically. Use the **password-attributes max-password-age** command to configure the number of days after which a SLX-OS user is forced to change their password.

The following example shows the configuration of password expiry for SLX-OS accounts.

```
SLX(config)# password-attributes max-password-age 100
```

In the above example, global password age for all users other than root is set to `100` days. Password expiry age can be specified in the range `0 to 999` days where `0` disables password aging. When disabled, SLX-OS passwords do not expire.

By having this configuration, on expiry of this configured period, a SLX-OS user, after login, is prompted to change their password.

GRUB Bootloader Password Protection

When there are no protections to access the `GRUB` boot loader, any user with access to the console during boot may interrupt the boot sequence to enter `GRUB` without authenticating. Securing `GRUB` to prevent unauthorized access prevents configuration changes at that level.

The following command configures GRUB Bootloader Password.

```
SLX(config-grub)# username root password fabos345
```

When a user with access to the console during boot interrupts the boot sequence to modify `GRUB`, the system prompts for and validate a credentials before entering `GRUB` menu entries other than default.

The `GRUB` credentials will be asked when accessing GRUB and only when the feature is enabled by using the above CLI.

Measured boot and Remote Attestation

Exploiting an embedded network device by planting the malware in one or more components of boot process is a type of security attack, which can go unnoticed as the malware may behave just like normal firmware.

Measured boot feature supports measuring the boot components and selected (custom) files during run time. Remote Attestation feature authenticates the hardware and software components (i.e., measurements from measured boot) to remote attestation server.

The following command is used to enable measured boot feature in SLX-OS device,

```
SLX# measured-boot enable
```

Note
The device must be rebooted for the above CLI to take effect.

To support Remote attestation, user must setup (Keylime) registrar server, which is not in the scope of this document (please refer online Keylime server installation guide).

The following commands are used to configure Keylime agent that runs on the SLX-OS device:

```
SLX(config)# remote-attestation
SLX(config-remote-attestation)# registrar-server <registrar-ipaddress>
```

To start Keylime agent on SLX-OS device, execute the below command.

```
SLX(config-remote-attestation)# agent-enable
```

Note
Refer Keylime server guides to start remote attestation using *keylime-tenant* utility.

Security Enhanced Linux (SE Linux)

Security-Enhanced Linux (SE Linux) is a Linux Kernel Module that enhances the security of SLXOS's underlying Linux OS. SE Linux works by providing security policies for access control at the operating system level. Support for Mandatory Access Control (MAC) is also available for use.

Security policies are a set of rules that implement access control restrictions for applications, processes, and files on the SLXOS's operating system. These rules are used by SE Linux to enhance security by preventing bypass of application security mechanism and enable containing the potential damages due to malicious or misbehaving applications.

Support for SE Linux is introduced in SLXOS version 20.4.1. As a part of this, MAC policy support for *SSHD* and *HTTPD* modules and their dependencies are added.

SE Linux has three modes of operation:

- In the *Disabled* mode, the operating system does not implement SE Linux policy and also does not label any persistent objects such as files. Not marking these persistent objects makes it harder to implement SE Linux in the future.
- In the *Permissive* mode, the operating system implements the SE Linux policy fully. All policy enforcement activities are logged. However, the policy is not enforced.
- In the *Enforcing* mode, the operating system implements the SE Linux policy completely including denying access, and activity logging.

SE Linux *Permissive* mode is enabled by default and cannot be changed.



Note

The last *1000 error log* entries will be saved in the `INFRA.txt` file within the support save logs.



Note

This feature is enabled on all platforms of SLXOS.

Use the **`show selinux status`**command to verify the current SE Linux status.

```
SLX # show selinux status
SE Linux status:                 enabled
SE Linuxfs mount:                /sys/fs/selinux
SE Linux root directory:         /etc/selinux
Loaded policy name:              mls
Current mode:                    permissive
Mode from config file:           enforcing
Policy MLS status:               enabled
Policy deny_unknown status:      allowed
Memory protection checking:      actual (secure)
Max kernel policy version:       31
```