NSP
Network Services Platform
Release 26.4

Security Hardening Guide

3HE-29838-AAAA-TQZZA
Issue 1
April 2026

Legal notice

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

Contents

About this document

Purpose

The *NSP Security Hardening Guide* is a reference document for increasing NSP security at the physical, OS, transport, user, and application levels.

Scope

The scope of this document is limited to describing the security hardening requirements and recommendations for an NSP deployment. In order to provide context for hardening practices, the guide also describes rigidly enforced security measures that are built into the product.

Safety information

For your safety, this document contains safety statements. Safety statements are given at points where risks of damage to personnel, equipment, and operation may exist. Failure to follow the directions in a safety statement may result in serious consequences.

Document support

Customer documentation and product support URLs:

- Documentation Center
- Technical support

How to comment

Please send your feedback to Documentation Feedback.

1 NSP host OS security

1.1 What host OS hardening measures are recommended?

1.1.1 General OS hardening measures

The following general OS hardening measures are recommended:

- Install a clean operating system environment with the minimum required packages as described in the *NSP Installation and Upgrade Guide*.
- Install the latest Recommended Patch Cluster from Red Hat (apply the patches supplied by Nokia for the NSP RHEL OS qcow2 image).
- Consult the *Host Environment Compatibility Reference for NSP and CLM* for up-to-date information about the recommended RHEL maintenance update and patch levels. Nokia supports customers applying RHEL patches or Windows patches provided by Red Hat, which include security fixes as well as functional fixes. If a patch is found to be incompatible with NSP/NFM-P, the patch may need to be removed until a solution to the incompatibility is provided by Red Hat or Nokia. Operating system patches of NSP-provided RHEL OS qcow2 images must be obtained from the NSP product group. Nokia supports only Nokia-provided RHEL OS disk images and OS patch bundles for qcow2 / OVA.
- Harden the RHEL operating system installation based on the CIS best practices described in Chapter 8, “RHEL OS security hardening”. The NSP RHEL OS qcow2 image is hardened in accordance with these supported CIS requirements only.
- Ensure that the system clocks of NSP components are always closely synchronized. The RHEL chronyd service is the mandatory time-synchronization mechanism to engage on each NSP component during deployment. For availability reasons, redundant external servers must be accessible to NSP.

  Note: An NSP station cannot serve as a time-synchronization source.
- Disable mDNS.
- Isolate NSP components with correctly configured firewalls: NSP components have no ingress or egress requirements to access the public Internet; hosts must be isolated . See “NSP Port Communications” in the *NSP Planning Guide* for information.
- Enable restricted root access to the NSP deployment, as described in 8.1.2 “Restricted root-user access” (p. 39).

1.1.2 NSP RHEL OS disk images

The Nokia-provided RHEL OS disk images are based on RHEL 8 and are available for KVM and Openstack hypervisors. An NSP RHEL OS image can be used only for the deployment of NSP software and not for the deployment of any other Nokia or third-party product.

An application that Nokia does not sanction must not be deployed on any OS instance that hosts an NSP component. Nokia reserves the right to remove any applications that are suspected of affecting NSP operation.

1.1.3 SELinux

NSP supports RHEL SELinux for enhanced system security and logging functions. See the *NSP System Administrator Guide* for information about SELinux implementation and management in NSP and the RHEL documentation for comprehensive SELinux configuration and implementation information.

All NSP system elements support SELinux in enforcing mode, except for an auxiliary database, which supports only permissive mode.

1.1.4 Sudoers file configuration

Some NSP components create rules in RHEL sudoers.d directories during installation. These rules allow NSP functions to run certain programs required for NSP operations. Rule files can be found in the /etc/sudoers.d/ directory and rule entries apply to NSP users. See 8.1 “What options do I have for RHEL sudoers configuration?” (p. 39) for more information.

2 Database security

2.1 What is recommended for database hardening for classic management?

2.1.1 Security recommendations

If an NSP deployment includes classic management, Nokia recommends the following:

- Enable IP validation on each main database. IP validation restricts the components that have database access to only those required. See the *NSP Installation and Upgrade Guide* for more information.
- Enable Oracle database error monitoring. Oracle database errors provide monitoring information that may help with troubleshooting or the detection of security violations, such as SQL injection attacks. When database error monitoring is enabled, the NFM-P raises an alarm when the Oracle software reports an error, such as an invalid SQL statement.

3 Kubernetes security

3.1 What is recommended for Kubernetes hardening?

3.1.1 Pod Security Admission (PSA) for Kubernetes cluster

Kubernetes PSA standards define different isolation levels for pods. The PSA controller is a built-in feature of Kubernetes that enforces pod security standards. Pod security restrictions are applied at the namespace level when pods are created. Pod security standards define three different policies (described below) to broadly cover the security spectrum.

Namespaces can be labelled to enforce the following pod security standards:

- **Privileged**

  A privileged policy is an unrestricted policy that provides the widest possible level of permissions. This policy allows for known privilege escalations.

- **Baseline**

  A baseline policy is a minimally restrictive policy that prevents known privilege escalations. Allows the default (minimally specified) pod configuration.

- **Restricted**

  A restricted policy is a heavily restricted policy that follows current pod hardening best practices.

NSP deploys pods in three namespaces that map to the pod security standards listed above. The namespace labels are configurable from the NSP deployer using attributes defined under the *kubernetes* section of nsp-config.yml.

Note: Most pods in the NSP cluster are deployed using the restricted pod security standard, however, certain pods require additional privileges and must be deployed with a less restrictive pod security standard.

4 Communications and network security

4.1 What should I configure for network and mediation security?

4.1.1 Network separation

Nokia strongly recommends configuring multiple NSP network interfaces to segregate different types of NSP traffic. You can segregate NSP client, mediation, and internal traffic by configuring NSP to use interfaces in separate networks for each traffic type.

The multi-interface implementation isolates different traffic types to one or more of the following networks:

- client – for GUI, OSS, REST, and browser clients, and clients such as Kafka subscribers
- mediation – for direct communication with managed NEs
- internal – for communication such as the following:
  - traffic within an NSP cluster
  - communication with NSP components and other systems; for example, the VSR-NRC, NFM-P, and NSP auxiliary databases
  - traffic related to NSP DR functions such as data replication and keepalive messaging between data centers

Using separate networks allows for additional security policies. To help secure internal services from unintended access, Nokia strongly recommends applying firewall rules to block ports reserved for internal NSP services from all traffic other than legitimate NSP clients on the internal network. For example, the NSP PostgreSQL database is an internal service between NSP components, and the northbound browser or API clients have no requirement to access the NSP PostgreSQL database. You can implement firewall rules to block the PostgreSQL port from all traffic other than legitimate NSP clients on the internal network to help secure the PostgreSQL database from unintended access.

To accommodate a deployment environment that hosts only one network, the use of multiple NSP network interfaces is optional. When NSP uses only one network for all communication, NSP client traffic shares the same network as the NE management traffic and the communication between NSP components. This type of configuration can pose a considerable security risk.

4.1.2 Firewall configuration

An NSP deployment has no ingress or egress requirement for access to the public Internet; each NSP host must be isolated using a properly configured firewall. Calico network policies are the recommended method for controlling traffic to an NSP cluster deployment.

NSP supports firewall deployment on all NSP host interfaces, however, firewall support among system components may vary. Components such as the NFM-P or WS-NOC that have multiple system elements may have additional firewall requirements. See the *NSP Planning Guide* and any specific component planning documentation, as required, for firewall port requirements and restrictions.

Note: Firewall deployment between the members of an NSP cluster is not supported.

Calico network policies for NSP

The NSP Kubernetes Platform uses Calico network policies to shield the NSP Kubernetes Platform from malicious attack. Calico network policies are created using the **calicoctl** command line interface, which is available on the master node of the NSP Kubernetes Platform. Due to the fact that the NSP Kubernetes platform is a closed system, Nokia recommends that network policies be configured at the interface level rather than the pod level.

Configuring Calico network policies requires the creation of host endpoints, which are resource definitions that identify the IP interfaces of VMs that host Kubernetes nodes, and global network policies (GNPs), which define the traffic filtering rules for a host endpoint. Calico resource definitions are configured using YAML files.

Typically, one host endpoint per cluster node and one GNP are created per network interface in the NSP cluster. In a single-network-interface deployment, only one host endpoint and one GNP need to be defined. In a three-network-interface deployment, a host endpoint and corresponding GNP must be defined for each of the following :

- the client network interface (for client and OSS systems)
- the internal network interface (for traffic between NSP systems)
- the mediation network interface (for communication to network elements)

See the calicoctl documentation for more information about managing Calico policies.

In the following example use-case for a single-interface deployment, the network operator wants to block all traffic coming to port 8566 of the cluster, with the exception of traffic coming from the redundant DR site. To set this configuration, the following steps are performed on both the active and standby clusters.

1. Create a host endpoint for each interface in the cluster. In “Calico host endpoint example” (p. 14), a single host endpoint is created.
2. Create a GNP for each interface in the cluster, adding all IP addresses of other clusters to the "notNets" section. See “Calico GNP example” (p. 15) for a single-interface sample configuration.
3. Apply the host endpoint and GNP. The following commands are an example:

   ```
   calicoctl apply -f a1_endpoint.yaml
   calicoctl apply -f a1_gnp.yaml
   ```

4. Verify that the policies have been applied by running the following commands:

   ```
   calicoctl get gnp -owide
   calicoctl get heps -owide
   ```

Calico host endpoint example

a1_endpoint.yaml

```
apiVersion: projectcalico.org/v3
kind: HostEndpoint
```

```
metadata:
name: node1-intf
labels:
   interface: enp1s0
spec:
interfaceName: enp1s0
node: cluster-a1
```

where

name is the name of the host endpoint

interface is the interface label

interfaceName is the interface name, which will be referenced by the GNP

node is the node name, taken from "NAME" as shown by the **kubectl get nodes -A** command

Calico GNP example

a1_gnp.yaml

```
apiVersion: projectcalico.org/v3
kind: GlobalNetworkPolicy
metadata:
  name: restrict-ssh
spec:
  selector: interface == 'enp1s0'
  applyOnForward: true
  types:
  - Ingress
  - Egress
  ingress:
  - action: Deny
    protocol: TCP
    source:
      notNets:
        - "fd11:0100:0127:0178:0000:0000:0179:0005/128"
        - "fd11:0100:0127:0178:0000:0000:0179:0003/128"
    destination:
```

```
    ports:
    - 8566
  - action: Allow
    protocol: TCP
  egress:
  - action: Allow
```

where

selector: interface matches the interface you configured in the HostEndpoint above

notNets specifies the IP addresses from the standby cluster

ports specifies the ports that allow traffic from the IP addresses specified in notNets (in this case, the standby cluster)

4.1.3 Mediation

The following is a summary of recommendations for mediation security:

- Enable secure transport protocols with CLI, NETCONF, and gRPC mediation.
- Use SCP or SFTP instead of clear file transfer equivalents such as TFTP and FTP.
- Use SNMPv3 instead of SNMPv1/v2. SNMPv3 supports authentication and encryption. SNMPv1/v2 provides no confidentiality and must be avoided.
- Verify that the mandatory RHEL chronyd service ensures that timestamps of logged activity are synchronized with other network elements. This is especially useful for precisely identifying timelines when troubleshooting an event or issue.
- Segregate the traffic between NSP and managed NEs in a separate network.

SSH

NSP supports strong SSH cryptographic algorithms by default. The default algorithms are updated as required to account for changes in the security level of specific algorithms.

SNMP

When SNMP mediation is required, SNMPv3, which supports authentication and encryption, is strongly recommended over SNMPv1/v2.

The SNMP recommendations are:

- Configure SNMPv3 to use both authentication and privacy protocols, which enables authentication and encryption features and enhances overall network security.
- Ensure that administrative credentials are properly configured with different passwords for authentication and encryption.

Lawful Intercept and data privacy

An NSP system that includes classic management can act as an optional remote controller for Lawful Intercept (LI) functions on SNMP devices that have native LI support. LI is a highly secure

function that is built into the device hardware; the NSP LI capabilities are limited to LI target specification and enabling or disabling LI on a target.

As a remote controller, NSP has no visibility of intercepted traffic; see “Lawful Intercept concepts” in the *NSP NFM-P Classic Management User Guide* for more information.

gRPC

When gRPC mediation is required, the NSP gRPC client can be configured to use two-way TLS to protect communication between NSP and the NEs; see the *NSP System Administrator Guide* for configuration information.

The gPRC recommendations are:

- Ensure that the "Secure" attribute slider is enabled when the gRPC mediation policy is created.
- Enable TLS communication between MDM and managed NEs by importing the NE’s self-signed TLS certificate into each MDM truststore. The NE certificate files must be transferred to NSP over a secure connection.

NETCONF or CLI

When NETCONF or CLI mediation is required, Telnet or SSH may be used as the transport protocol.

The NETCONF and CLI recommendations are:

- Telnet is unsecure and must be avoided. Enable SSH2 transport protocol when the NETCONF or CLI mediation policy is created.

VSR-NRC communication to the network

See the following documentation references for information about VSR-NRC communication to the network.

IP Routing Protocols

See “Unicast routing and MPLS” in the *7450 ESS, 7750 SR, 7950 XRS, and VSR Security Best Practices and Hardening Guide*.

PCEP

See “PCEP over TLS” in the *Segment Routing and PCE User Guide*.

4.2 What should I know about TLS in NSP?

4.2.1 TLS support

TLS is a cryptographic protocol for establishing encrypted communication between a client and server. NSP supports TLSv1.2 and TLSv1.3 protocols by default.

Note: Outdated TLS versions present a security risk and are disabled by default in NSP. Customers that enable older unsecure TLS protocols do so at their own risk.

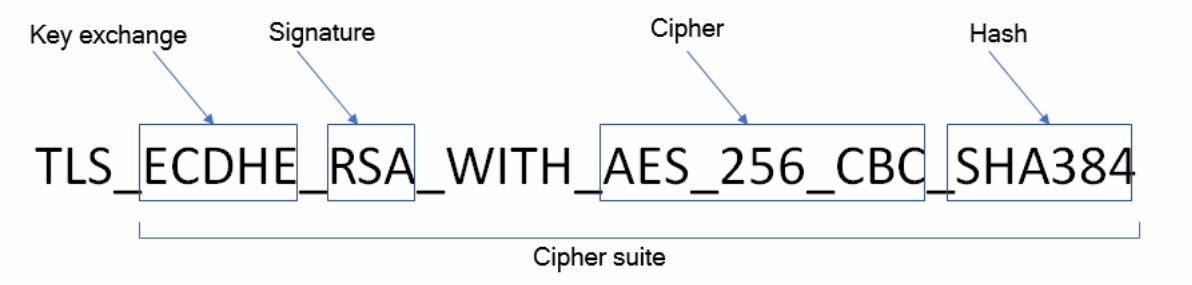
4.2.2 Cipher suites

Cipher suites define a set of cryptographic algorithms for each TLS connection. During cipher suite negotiation, both server and client must agree on the same cipher suite in order to establish a connection. A cipher suite defines the combination of key exchange, authentication, encryption, and integrity algorithms for the TLS connection.

- key exchange (asymmetric cipher); for example: ECDHE, DHE, RSA
- authentication (signature/certificate type); for example: RSA, DSA, ECDSA
- confidentiality (symmetric cipher); for example: AES_256_CBC, AES_256_GCM
- integrity (hash); for example: SHA384, SHA256, SHA

The following example displays the format of a TLS cipher suite.

Figure 4-1 TLS cipher suite example



Note: NFM-P supports the ability for administrators to customize the list of supported TLS cipher suites, however, NSP does not.

4.2.3 Perfect Forward Secrecy

Perfect Forward Secrecy (PFS) is a feature of specific key agreement protocols in which there is no link between the server's private key and each session key. Therefore, if an attacker gains access to a server private key, the attacker cannot use the private key to decrypt any archived TLS sessions. Cipher suites prefixed with "TLS_DHE" and "TLS_ECDHE" support PFS.

Note: Exercise caution when removing TLS cipher suites; incompatible cipher suites prevent NSP system access from browser and OSS clients.

4.2.4 Authenticated Encryption with Additional Data (AEAD)

In general, TLSv1.2 and TLSv1.3 ciphers compute a mac over the plaintext then the authenticated payload is encrypted. Although highly unlikely in an NSP environment, this "mac-then-encrypt" approach could open the possibility for some sophisticated padding attacks. Authenticated Encryption with Additional Data (AEAD) is a special class of ciphers that combines encryption and integrity into one operation (ie. mac-and-encrypt). These ciphers compute mac and encrypt simultaneously which can mitigate padding attacks. In NSP, AEAD cipher suites are supported with

TLSv1.2 and TLSv1.3. These cipher suites are identified with Galois Counter Mode algorithms (ie. AES-GCM) and CHACHA20_POLY1305.

4.2.5 Elliptic Curves

For cipher suites using an EC-based DHE key exchange, NSP supports standard TLS curves.

4.2.6 Diffie-Hellman parameters

For cipher suites using a non-EC based DHE key exchange, NSP supports 2048-bit DH parameters. Clients that do not support 2048-bit DH modulus cannot connect to NSP with a DHE cipher.

NSP clusters do not offer any cipher suites supporting DHE key exchange.

4.2.7 Cipher Preference

Some attacks on TLS servers can be mitigated by enforcing the server's cipher order instead of allowing the client to choose the cipher. NSP server interfaces accessible by external TLS clients are configured to enforce server-side cipher order.

4.2.8 Cipher customization

NFM-P supports the ability for administrators to customize the list of supported TLS cipher suites. The default list of cipher suites provides a balance of algorithm strength and compatibility. That said, NFM-P administrators may still chose to customize the list of supported cipher suites. See the *NSP System Administrator Guide* for more information about updating TLS versions and ciphers.

Note: Exercise caution when customizing TLS protocols and cipher suites. Incompatible cipher suites will prevent NSP/NFM-P system access from browser and OSS clients.

4.2.9 TLS certificates

NSP supports the following types of certificates:

- Kubernetes infrastructure certificates - created by the Kubernetes platform and applied to the Kubernetes registry, NSP Deployer, and NSP cluster control plane
- NSP issuer certificates - used for signing leaf certificates that are distributed to internal NSP subsystem endpoints and external-facing NSP application endpoints

  If you don’t provide an issuer key or certificate during secrets installation, then NSP creates them for you.
- NSP server certificate - also known as custom certificates, these are used by the NSP cluster gateway for external-facing NSP application endpoints

  If you don't provide an NSP server key or certificate during secrets installation, NSP uses the external issuer to sign a certificate used to secure external-facing NSP application endpoints.

NSP currently supports 2048-bit or 4096-bit RSA certificates.

Note: Certificates with the RSA key size of 4096-bits are recommended.

NSP PKI server

The NSP PKI server is a standalone utility that signs TLS certificate signing requests (CSRs) from requesting entities in an NSP system. The NSP PKI server utility is mandatory and is used to sign TLS certificates for internal services used by NSP components that reside outside of the cluster (such as NFM-P or the NSP auxiliary database). The service uses an access control list based on the NSP cluster configuration. Consequently, the service responds only to certificate requests from known addresses in the nsp-config.yml file of the local cluster.

Note: The PKI server utility is intended for NSP components only.

Certificates signed by the NSP PKI server have the following fields:

- Signature Algorithm: sha256WithRSAEncryption
- RSA key length: 2048-bit
- validity period: configurable at NSP PKI initialization
- organization name: configurable at NSP PKI initialization
- country name: configurable at NSP PKI initialization
- state or province name: configurable at NSP PKI initialization

During NSP deployment, a CSR is sent to a local PKI server, which returns a signed certificate. NSP uses a server certificate and key to seed a local copy of the PKI utility, which is active for long enough to sign the pod certificates.

Note: The PKI server is intended for use by NSP components only.

Certificate Expiry

NSP TLS certificate replacement may be required when:

- a component is added to the NSP system
- a system IP address or hostname changes
- a TLS certificate nears or reaches expiry

An NSP/NFM-P checks the expiry date of each local TLS certificate during installation and every 24 hours thereafter. If a certificate is expired or approaching expiry, NSP raises one of the following *SSLKeystore* or *SSLClientKeystore* alarms:

- Warning, if the certificate is to expire within 30 days of the current time
- Critical, if the certificate is to expire within 7 days of the current time
- Critical, if the certificate is expired

NFM-P expiry alarms are shown in NSP and the Alarm Window of the NFM-P GUI.

Configuring advance warning for certificate expiry

Administrators must carefully monitor and refresh NSP TLS certificates before expiry. If certificates expire, some NSP functions that depend on secure communication may be inoperable. For example, NSP clients may be completely unable to connect to NSP.

Using the NSP and NFM-P GUI, administrators can configure alarm policies (referred to in NSP as “e-mail policies” and the NFM-P as “alarm e-mail policies”) to ensure that advance warning emails are sent for certificate expiry events. For example, an NSP policy containing an advanced filter for “*Alarm Name equals SSLKeystoreCertificateExpiring*” sends an email to the recipient list when NSP raises the initial warning alarm for TLS certificate expiry.

See the *NSP Network and Service Assurance Guide* or *NSP System Administrator Guide* for more information about configuring alarm e-mail policies, depending on platform installation type.

4.3 How do certificate signatures work?

4.3.1 Self-signing certificate bundles

NSP supports self-signing for customer-created certificate bundles to verify that installed certificates come from trusted sources. After you have created a key pair and loaded the secret file into Kubernetes, you can use the certificate-bundle-builder tool to create signed certificate bundles. See the *NSP System Administrator Guide* for more information about certificate administration.

5 NSP user security

5.1 How does NSP user authentication work?

5.1.1 Basic elements

The NSP user authentication mechanism, OAuth 2.0, provides the following single sign-on (SSO) functions:

- local and remote user authentication
- configurable login protection

The *NSP System Administrator Guide* describes NSP SSO functions and parameters.

5.1.2 OAuth 2.0

OAuth 2.0 is an implementation of the Keycloak open-source identity and access-management solution that employs the standard OAuth 2.0 protocol. OAuth 2.0 maintains a local user database, supports remote users, and can temporary or permanently lock out users to prevent brute-force or random-guess attacks.

Migrating from CAS to OAuth 2.0

In NSP Release 24.4, user management changed from NFM-P/CAS to OAuth 2.0. When upgrading from an older release that uses NFM-P/CAS to a current release that uses OAuth 2.0, you must import local users and user groups provisioned in the NFM-P to NSP. Remote users do not need to be imported, as they are created automatically when they log in.

Following the upgrade, NFM-P user security must be used only to manage XML API users. Users and system security in NSP must be used to manage all other user types.

See the *NSP System Administrator Guide* for more information about importing users and groups from NFM-P.

5.1.3 Local user authentication

NSP local user authentication uses robust password encryption and has rules that govern the password change and complexity requirements.

Password encryption

A one-way cryptographic hash is applied to all NSP user passwords stored in the local database. The encryption protects against an accidental or intentional database disclosure, as the password cannot be decrypted. To further mitigate against password attacks, a randomized salt is added to each user password before the one-way cryptographic hash is applied.

Password complexity

User password complexity rules are configurable; the following are the default rules, which state that a password must:

- be at least ten characters
- not be the same as the previous three passwords
- include at least one of the following special characters:
  ( ) ? ~ ! @ # $ % & * _ +
- include at least one lowercase character
- include at least one uppercase character
- include at least one digit
- not be the username
- not equal the email address

Password changes

One administrator account is created by default during NSP system installation. During the initial administrator login using the default password, the user is prompted to change the password. The creation of additional local users includes an option to force the user to change the password during the initial login.

Note: Nokia recommends that you enable the initial password-change option.

5.1.4 Remote user authentication

NSP supports LDAP/S, RADIUS, and TACACS+ remote user access.

When RADIUS remote authentication is configured, multi-factor authentication (MFA) can be used for added protection. NSP supports MFA through the RADIUS access-challenge packet from the RADIUS server. When NSP receives the access-challenge from the RADIUS server, the user is prompted for one-time verification code.

See the *NSP System Administrator Guide* for information about configuring remote users.

Note: If LDAP is used for remote access, it is strongly recommended that you use LDAPS to secure the LDAP communication.

Note: If RADIUS is used for remote access, it is strongly recommended that you configure MFA via the access-challenge from the RADIUS server.

5.1.5 Login protection

You cannot enable both temporary and permanent user lockouts; if a user lockout is to be enforced, only one mechanism can be active at any time.

Note: Temporary user lockout is enabled by default.

Note: Nokia recommends deploying NSP with brute-force protection enabled and the parameters configured in accordance with your security policy.

5.1.6 Session controls

To enhance security, an idle session timeout and token lifespan can be applied. The timeout and lifespan apply to NSP SSO and REST sessions:

Note: Functions that require near-real-time event updates communicate continuously with NSP, so do not time out from inactivity.

You are encouraged to assess the number of concurrent sessions that your deployment requires and to set the maximum allowed number to the lowest value that meets the requirement.

5.2 What is NSP User Access Control?

5.2.1 User groups and roles

NSP User Access Control (UAC) is an optional mechanism that enables an NSP administrator to define user access rights to NSP functions and data. When UAC is disabled, NSP user access is unrestricted.

NSP enables you to define resource groups that contain NSP data-model objects; for example, equipment and service components. NSP also enables the creation of user groups and roles. A role, which is assigned to a user group, can be given read, write, or execute permissions to specific functions and resource groups. Consequently, an NSP user has the access defined in a role to only the resources in the associated resource groups.

A UAC resource group can include objects such as the following:

- service — IP links, services, service sites, service endpoints, service bindings
- equipment — chassis, LAGs, devices
- KPI — NE and service performance indicators
- Analytics — analytics resources

5.3 What is login banner customization?

5.3.1 Customizing the NSP login banner

A login banner warns or advises a user during a login attempt of the acceptable system use policies. In general, such a banner must warn unauthorized users not to proceed and must display a clear statement about system logging and monitoring to detect unauthorized use or access. It is recommended that you configure an NSP login banner in accordance with your security policy.

An NSP administrator can create a customized security statement that is presented to the NSP operator at each NSP login attempt. An administrator can optionally force operator acknowledgement of the security statement.

The *NSP System Administrator Guide* describes how to enable and configure the NSP login banner, which is disabled by default.

6 NE security in NFM-P

6.1 What encryption options do I have in NFM-P?

6.1.1 MACsec pre-shared keys

The NFM-P may optionally be used to configure MACsec pre-shared keys (PSKs) for encrypted switch-to-switch connectivity between supported NEs. Each security association in the NE contains a Security Association Key (SAK) where the cryptographic operations used to encrypt the datapath PDUs. The SAK is the secret key used by a security association (SA) to encrypt the channel.

A PSK can be created by NSP. Each PSK is configured with two fields:

- Connectivity Association Key Name
- Connectivity Association Key (CAK) value

The NFM-P can be configured with scheduled hitless re-keying of PSKs.

The NFM-P supports two sources for keying material:

1. local: PSK is generated locally
2. hardware security module (HSM): PSK is generated by a supported HSM; see the *NSP NFM-P Network Element Compatibility Guide* for a list of supported HSMs

Before an HSM can be used for key management, the HSM must be added to the NFM-P configuration. See the *NSP System Administrator Guide* for more information.

Note: The NFM-P does not store CAKs generated by an HSM

Note: For increased security, Nokia recommends scheduling periodic re-keying of PSKs.

6.1.2 Network group encryption

The NFM-P may optionally be used to deploy network group encryption (NGE) attributes to NEs. The NFM-P uses SNMP to deploy general NGE attributes to NEs and uses SSH2 sessions to configure the key values. You can use an existing SSH2 user account on each NE, or, to facilitate the tracking of key value configuration activity, you can use the User NGE account. The NFM-P creates the account on each participating NGE NE and uses the account only for creating and updating key values. The NFM-P user activity log records all NGE configuration activity.

Note: To facilitate the tracking of key value configuration activity, use the "User NGE" account on each NE.

Note: For increased security, Nokia recommends using a scheduled task for the regular and automatic replacement of the keys in the key group.

6.1.3 FIPS

The NFM-P supports the United States Federal Information Processing Standards (FIPS) for NE management and client communication. See the *NSP Installation and Upgrade Guide* for information about enabling FIPS.

7 Network security in the NSP UI

7.1 What is OS Security for NEs?

7.1.1 NE management using anti-theft mode

Anti-theft mode can be enabled on compatible devices. Enabling anti-theft mode ensures that if the device is stolen, it cannot be reconfigured and reused.

When anti-theft mode is enabled, after each reboot the device is locked, requiring the OS security password before allowing access to the configuration commands, preventing the device from being used in a new network deployment.

See the NE documentation for information about configuring an OS security password on the device.

7.2 What is an OS security policy in NSP?

7.2.1 OS security password management

The use of an OS security policy provides the ability for the NSP to continue to manage a model-driven NE when anti-theft mode is enabled. The OS security policy defines the password the NSP uses to unlock the NE for NSP management when the NE is locked by anti-theft protection.

The OS security password can be configured using CLI on the NE, or configured using NSP and pushed to the NE.

If a password is configured on the NE prior to discovery, the password configured in the OS security policy in the NSP UI must match the password configured on the NE to discover the NE.

If the NE is locked at the time it is discovered by the NSP, successful discovery unlocks the NE.

Caution: NSP 25.4 or later artifacts with adaptation for anti-theft must be installed in the NSP deployment before anti-theft is enabled on the NEs. Management of classic NEs in anti-theft mode is not supported in NSP.

7.2.2 Applying the policy to the NE

An OS security policy can be included with a unified discovery rule. The anti-theft password provided by the OS security policy is applied to all compatible NEs that are discovered by the unified discovery rule. If the discovery rule includes devices that are not compatible with anti-theft or do not have anti-theft enabled, the OS security policy has no effect on the NE.

When anti-theft is enabled on an NE the NE immediately locks, requiring the password to unlock and continue to manage the NE. Therefore, an OS security policy must be associated to the discovery rule before anti-theft mode is enabled on the NE.

Click on an OS security policy in the **Network Security**, **OS Security Policies** view to see the discovery rules that use the policy and the included NEs, that is, the NEs for which the password in the policy applies.

See 7.3 “Pathway: manage OS security” (p. 34) for the high-level process.

7.2.3 Troubleshooting

If an anti-theft error has occurred, check the MDM server and MDM Tomcat logs for details on the root cause.

Table 7-1, “OS Security and Anti-theft troubleshooting” (p. 29) describes some failure scenarios and provides troubleshooting information.

Table 7-1 OS Security and Anti-theft troubleshooting

Problem	Relevant message in logs	Alarm	Solution
In rare cases, after an NE reboot caused by OS security password or anti-theft operations, a mismatch of the anti-theft or OS security status may occur between NE and the NSP.	—	—	Re-discover the NE in force mode using the RESTCONF API. See Device Administration and Mediation RESTCONF APIs on the Network Developer Portal
When assigning an OS security policy to a discovery rule, if there are already NEs managed that support anti-theft, updating the OS security policy fails.	`Can not change the os policy while managed nodes support Anti Theft` in RESTCONF response or mdm-tomcat log	—	1. Unmanage all NEs associated with the discovery rule that support anti-theft 2. Associate an OS security policy with the discovery rule. 3. Manage the NEs again. If the discovery rule already has an OS security policy associated with it and you want to attach a new OS security policy, you need to delete the discovery rule, create a new one and add the new OS security policy and NEs. The OS security policy cannot be changed if one is already attached to the discovery rule.
If the NE is locked, deploying a device configuration template to the NE fails.	—	—	Verify that NEs are unlocked, reachable, and have the correct OS security policy associated before updating any configuration.
Problems encountered when unlocking the NE			

Table 7-1 OS Security and Anti-theft troubleshooting (continued)

Problem	Relevant message in logs	Alarm	Solution
Password mismatch	`Failed to process RPC request to NE: Operation failed - incorrect password entered` in mdm-server log	Error while unlocking anti-theft mode: password is incorrect or Netconf Error. Check MDM server logs for details.	• If the operation failed because of a network connectivity issue, unlock the NE from NSP after the networking issue is fixed; see “How do I unlock an NE?” in the *NSP Device Management Guide*. • If the operation failed on all NEs associated to the same OS security policy: 1. Update the current OS security password with the "Change password only" option on the security policy edit form; see 7.5 “How do I edit or delete an OS Security policy?” (p. 37) 2. Unlock the NE; see “How do I unlock an NE?” in the *NSP Device Management Guide*. • If the operation failed on subset of NEs associated with the same os-security policy, update the OS security password on the affected NEs using CLI.
No OS security password is attached to the unified discovery rule	`No OS policy attached to the discovery rule` in mdm-tomcat or mdm-server logs	No OS policy attached to the discovery rule. Attach the OS policy and force rediscover.	Attach the OS policy and perform a force rediscover operation in the API.
An OS security password is attached to the unified discovery rule, but can’t be found by NSP	—	—	Attach the OS policy and perform a force rediscover operation in the Device Administration API.
Problems encountered when pushing a password			

Table 7-1 OS Security and Anti-theft troubleshooting (continued)

Problem	Relevant message in logs	Alarm	Solution
The NE is not associated with an OS security policy	`Can not find the password to be pushed to node ID; Please make sure the node is associated with a security policy` in RESTCONF `OS password is not configured on node ID; Please configure OS password first` in mdm-tomcat log	—	Associate an OS security policy to the discovery rule the NE belongs to.
Password specified in RPC is the same as the password on the NE.	`"The new password (from the os policy) is the same as what is currently configured on node` from RESTCONF	—	1. Update the OS security password using NSP, see 7.5 “How do I edit or delete an OS Security policy?” (p. 37) 2. Trigger a password push; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*.
Failed to push new password when the previous push failed.	`Push os password is in progress or failed on node` from RESTCONF	—	• If the previous push failure was caused by network connectivity issues, fix the issue and trigger password push again; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*. • If the previous push failure was caused by mismatch of CLI OS security password, update the mismatched password on the NE and trigger password push again.
Failed to push new password because the previous push is still in progress.	`Push os password is in progress or failed on node` from RESTCONF	—	Wait until the current password push process is complete before triggering password push operation again.

Table 7-1 OS Security and Anti-theft troubleshooting (continued)

Problem	Relevant message in logs	Alarm	Solution
Password mismatch	`Operation failed - current-password incorrect` from mdm-server log	Error while pushing OS password : current password is incorrect or Netconf Error. Check MDM server logs for details.	• If the operation failed on all NEs associated to the same os-security policy: 1. Update the current OS security password with the "Change password only" option on the security policy edit form; see 7.5 “How do I edit or delete an OS Security policy?” (p. 37) 2. Update the policy again with the “Change password and create a scheduled operation to push it to the NEs” option with a new OS security password. • If the operation failed on subset of NEs associated with the same OS security policy: 1. Update the OS security password on the affected NEs using CLI. 2. Trigger a password push again; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*.
Problems encountered when updating anti-theft mode			
Push anti-theft mode to NE without an OS security password configured on the NE	`OS password is not configured on node` *`ID`* `Please Configure os password first` from RESTCONF	—	1. Trigger a password push; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*. 2. Turn on anti-theft mode; see “How do I enable or disable anti-theft mode?” in the *NSP Device Management Guide*.

Table 7-1 OS Security and Anti-theft troubleshooting (continued)

Problem	Relevant message in logs	Alarm	Solution
Failed to enable or disable anti-theft due to password mismatch	`Operation failed - incorrect password entered` in mdm-server log	Error while pushing Anti Theft mode : current password is incorrect or Netconf Error. Check MDM server logs for details.	• If the operation failed on all NEs associated to the same os-security policy: 1. Update the current OS security password with the "Change password only" option on the security policy edit form; see 7.5 “How do I edit or delete an OS Security policy?” (p. 37) 2. Turn on anti-theft mode; see “How do I enable or disable anti-theft mode?” in the *NSP Device Management Guide*. • If the operation failed on a subset of NEs associated with the same OS security policy: 1. Update the OS security password on the failed NEs using CLI. 2. Turn on anti-theft mode; see “How do I enable or disable anti-theft mode?” in the *NSP Device Management Guide*.
Anti-theft mode is already set to the mode specified in RPC.	`Anti-theft mode is already set to` *`mode`* `on node` *`ID`* from RESTCONF	—	—

7.3 Pathway: manage OS security

7.3.1 OS security task overview

The following is a generic flow of the high-level tasks that are typically used to configure and manage OS security using the NSP. As appropriate, review the pathway associated with each task for detailed instructions.

Note: If the anti-theft status of the NE is locked at the time of discovery by NSP, successful discovery unlocks the NE.

7.3.2 Stages

1

Are the NEs already configured with an OS security password or anti-theft mode?

- If no, see “OS security not configured on NEs” (p. 34)
- If yes, see “OS security already configured on NEs” (p. 35)

OS security not configured on NEs

2

Create an OS security policy, see 7.4 “How do I create an OS security policy?” (p. 36).

Verify that the password in the policy matches the password configured on the NEs.

3

Include the policy in a unified discovery rule; see “How do I discover devices?” in the *NSP Device Management Guide*.

4

Discover the devices as needed; see “How do I discover devices?” in the *NSP Device Management Guide*

5

Push the OS password to devices; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*.

6

Enable anti-theft mode; see “How do I enable or disable anti-theft mode?” in the *NSP Device Management Guide*.

OS security already configured on NEs

7

Record the OS password that has been configured on the NEs.

8

Create an OS security policy, see 7.4 “How do I create an OS security policy?” (p. 36).

Verify that the password in the policy matches the password configured on the NEs.

9

Include the policy in a unified discovery rule; see “How do I discover devices?” in the *NSP Device Management Guide*.

10

Discover the devices as needed; see “How do I discover devices?” in the *NSP Device Management Guide*

Post-discovery options

11

Edit the OS security policy to update the password as needed, see 7.5 “How do I edit or delete an OS Security policy?” (p. 37).

12

Push the OS password to devices; see “How do I push the OS password to NEs?” in the *NSP Device Management Guide*.

13

Enable anti-theft mode; see “How do I enable or disable anti-theft mode?” in the *NSP Device Management Guide*.

7.4 How do I create an OS security policy?

7.4.1 Purpose

CAUTION

Communication problems

If the OS security password in NSP does not match the password configured on the NE, communication with devices will be affected.

Verify that the password matches the configuration on the NE.

Use this procedure to configure communication with NEs in anti-theft mode. This procedure requires an administrator role.

7.4.2 Steps

1

Open **Network Security**, **OS Security Policies**.

The system displays the list of configured policies.

2

Click **+ OS Security Policy**.

3

In the form that opens, configure a policy name and optional description. In the **Password** field, enter the password configured on the NE.

4

Click **Create**. The policy is added to the list.

5

To apply the policy to NEs, add it to a discovery rule; see “How do I discover devices?” and “How do I edit or delete a discovery rule?” in the *NSP Device Management Guide*.

END OF STEPS

7.5 How do I edit or delete an OS Security policy?

7.5.1 Purpose

CAUTION

Communication problems

If an OS security password is edited when it is in use by a discovery rule, communication with devices may be affected.

If the password has been changed on the NE, verify that the updated credentials match the configuration on the NE.

Use this procedure to update the description or password in an OS Security policy, or to delete the policy. You can update the password only, to resolve a password mismatch, or create an operation to push the updated password to the NEs. See the *NSP Device Management Guide* for more information about operations.

This procedure requires an administrator role.

7.5.2 Steps

1

Open **Network Security**, **OS Security Policies**.

The system displays the list of configured policies.

Choose a policy and click ⋮ (Table row actions), **Edit**.

To update the password to resolve a password mismatch

2

To resolve a mismatch between the password on the NE and the OS security policy, perform the following.

1. Enable the **Change password** toggle.

   Change password options are displayed.
2. Choose **Change password only**.
3. Enter and confirm the new password.

   Ensure that the new password matches the new password on the NEs.
4. Click **Update**.

To update a password and push it to the NEs

3

To change the password and push the new password to NEs, perform the following.

1. Enable the **Change password** toggle.

   Change password options are displayed.
2. Choose **Change password and create a scheduled operation to push it to the NEs**.
3. Enter the current password and click **Validate** to confirm.
4. Click **Schedule**.

   The Create Operation form opens. Proceed to Step 4

4

To schedule a password push to the NEs, perform the following in the Create Operation form:

1. Enter a name in the Operation Name field.
2. In the Operation Inputs panel, choose Enabled or Disabled from the Anti-theft Mode drop-down.
3. In the **View/Edit Schedule** panel, configure the timing of the operation:
   - To start the operation immediately, choose **Run Immediately**.
   - To schedule the operation to start at a later time, choose **Schedule a date and time** and configure the scheduling options.

To delete an OS security policy

5

To delete an OS Security policy, choose a policy , click ⋮ (Table row actions), **Delete**, and confirm.

A policy cannot be deleted if it is in use by a discovery rule.

END OF STEPS

8 RHEL OS security hardening

8.1 What options do I have for RHEL sudoers configuration?

8.1.1 Default configuration mapping

The following table provides the default mapping between NSP components, sudoers files, and users.

NSP component	Sudoer files	User
NFM-P main or auxiliary server	nfmp-main, nspos-sudo	nsp
NFM-P main database	nfmp-main-db	oracle
NSP auxiliary database	nspos-auxdb, nspos-auxdbproxy	samauxdb
CLM	clm-sudo, nspos-sudo	nsp

8.1.2 Restricted root-user access

If you employ a special sudoers configuration, privileged users that you create can execute only specific NSP management and deployment commands. If a user other than the privileged non-root user attempts to execute a restricted command, the command fails.

You can also prevent remote root-user access to the stations in an NSP deployment by designating a specific privileged user for remote access.

You can restrict root-user access on the NSP deployer host and cluster VMs; NSP auxiliary database; and on NFM-P main server, auxiliary server, and main database stations.

Note: Client delegate servers do not support restricted root access.

The root user performs the initial OS and VM setup for an NSP deployment and creates the alternative users with restricted access. The root user is not required for NSP deployment operations afterward.

Restricted root-user access:

- assigns sudo privileges for only the required commands per user
- ensures that any configuration or control actions are traceable to a specific user

See “Restricting root-user system access” in the *NSP Installation and Upgrade Guide* for more information.

9 Data privacy summary

9.1 How do NSP network and user data privacy work?

9.1.1 Purpose

This appendix summarizes how NSP treats private data that is collected, processed, or retained, such as:

- user authentication data
- NE data
- subscriber data
- email notification policy data

See 9.1.2 “NSP data privacy” (p. 41) or 9.1.3 “NFM-P data privacy” (p. 43) for specific summary information.

9.1.2 NSP data privacy

The following table lists and describes, by category, how NSP treats network and user data.

Table 9-1 NSP treatment of private data

Data category	Description and treatment
Local user data (local authentication)	
Type of data	• username and password • email • IP address
Purpose	• authentication of local NSP users • user email addresses (optional) to send notifications for specific events; for example, alarms or account status • IP address provides accountability of individual product access.
Storage	• local database • logs
Retention	Data is retained in the database until an authorized user deletes it. Log retention time can vary based on log file size and the number of log backups.
Processing	Local user data is processed for the stated purpose.
Access	Authorized users

Table 9-1 NSP treatment of private data (continued)

Data category	Description and treatment
Safeguards	• Additional local users must be created by an authorized user. • Database access is restricted to authorized users. • TLS secures data in transit. • Passwords for local users are hashed before they are stored. • Log file access is restricted to authorized users.
Comments	Local authentication is performed using a local database of users and a local security scheme.
NE data	
Type of data	• username and password • IP address
Purpose	• NE authentication • NE IP address for NE discovery and access
Storage	• local database • logs
Retention	Data is retained in the database until an authorized user deletes it. Log retention can vary based on the log file size and number of log backups.
Processing	NE data is processed for the stated purpose.
Access	Authorized users
Safeguards	• NEs are configured by authorized users. • Database access is restricted to authorized users. • Secure transit option is available. • Passwords for NE users are encrypted before being stored. • Log file access is restricted to authorized users.
Subscriber data	
Type of data	• MAC address • IP address
Purpose	• statistics • SLA support • troubleshooting
Storage	• local database • logs
Retention	Data is retained in the database until an authorized user deletes it. Log retention can vary based on the log file size and number of log backups. Retention period for statistics can be configured.
Processing	Subscriber data is processed for the stated purpose.

Table 9-1 NSP treatment of private data (continued)

Data category	Description and treatment
Access	Authorized users
Safeguards	• NEs are configured by authorized users. • Database access is restricted to authorized users. • Log file access is restricted to authorized users.
E-mail notification policy data	
Type of data	• username and password • email address (sender) • email address (recipient)
Purpose	• username, password, and sender's email address are used for SMTP configuration • recipient email addresses are required to create email notification policies in supported functions
Storage	• local database
Retention	Data is retained in the database until an authorized user deletes it. By default, SMTP server and email notification policies are not configured.
Processing	SMTP server configuration and email notification policies are processed for the stated purpose.
Access	Authorized users
Safeguards	• SMTP configuration and email policies are configured by authorized users. • Database access is restricted to authorized users. • The password for SMTP configuration is encrypted before being stored.

9.1.3 NFM-P data privacy

The following table lists and describes, by category, how the NFM-P treats network and user data.

Table 9-2 NFM-P treatment of private data

Category	Description
Local user data (local authentication)	
Type of data	• username and password • email • IP address
Purpose	• authentication of local NSP users • user email addresses (optional) to send notifications for specific events; for example, alarms or account status • IP address provides accountability of individual product access.

Table 9-2 NFM-P treatment of private data (continued)

Category	Description
Storage	• local database • logs
Retention	Data is retained in the database until an authorized user deletes it. Log retention time can vary based on log file size and the number of log backups.
Processing	Local user data is processed for the stated purpose.
Access	Authorized users
Safeguards	• Additional local users must be created by an authorized user. • Database access is restricted to authorized users. • TLS secures data in transit. • Passwords for local users are hashed before they are stored. • Log file access is restricted to authorized users.
Comments	Local authentication is performed using a local database of users and a local security scheme.
Customer profile data	
Type of data	• name • email • address • phone
Purpose	Data may be used by an authorized user for associating customers to configured services.
Storage	Local database
Retention	Data is retained in the database until an authorized user deletes it.
Processing	Customer profile data is processed for the stated purpose.
Access	Authorized users
Safeguards	• Customer profile must be created by an authorized user. • Database access is restricted to authorized users.
NE data	
Type of data	• username and password • IP address
Purpose	• NE authentication • NE IP address for NE discovery and access
Storage	• local database • logs Note that NE backups that are stored on the NFM-P server may contain data that is not stored in the NFM-P database. Data contained in the NE backup files are dependent upon the NE type and version; therefore the privacy statements for the individual NEs must be consulted.

Table 9-2 NFM-P treatment of private data (continued)

Category	Description
Retention	Data is retained in the database until an authorized user deletes it. Log retention can vary based on the log file size and number of log backups.
Processing	NE data is processed for the stated purpose.
Access	Authorized users
Safeguards	• NEs are configured by authorized users. • Database access is restricted to authorized users. • Secure transit option is available. • Passwords for NE users are encrypted before being stored. • Log file access is restricted to authorized users.
Subscriber data	
Type of data	• MAC address • IP address • International Mobile Subscriber Identity (IMSI) • International Mobile Station Equipment Identity (IMEI) • Mobile Station International Subscriber Directory Number (MSISDN) • Access Point Name (APN)
Purpose	• statistics • SLA support • troubleshooting • Analytics • UE or network node performance information
Storage	• main database • NSP auxiliary database • logs • auxiliary servers (optional): statistics
Retention	Data is retained in the database until an authorized user deletes it. Log retention can vary based on the log file size and number of log backups. Retention period for auxiliary servers can be configured.
Processing	Subscriber data is processed for the stated purpose.
Access	Authorized users
Safeguards	• NEs are configured by authorized users. • Database access is restricted to authorized users. • Secure transit option is available. • File access is restricted to authorized users. • Log file access is restricted to authorized users.

Table 9-2 NFM-P treatment of private data (continued)

Category	Description
E-mail notification policies	
Type of data	• username and password • email address (sender) • email address (recipient)
Purpose	• Username, password, and sender's email address are used for SMTP configuration. • Recipient email addresses are required to create email notification policies in supported functions.
Storage	• local database
Retention	Data is retained in the database until an authorized user deletes it. By default, SMTP server and email notification policies are not configured.
Processing	SMTP server configuration and email notification policies are processed for the stated purpose.
Access	Authorized users
Safeguards	• SMTP configuration and email policies are configured by authorized users. • Database access is restricted to authorized users. • Password for SMTP configuration is encrypted before being stored.

10 NSP interface cryptography

10.1 What algorithms are supported per NSP component?

10.1.1 TLS server algorithms for client networks

Component	Protocol	Ciphersuites	Signatures	Groups
NSP Web Server	TLSv1.3	TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256	rsa_pss_rsae_sha256 rsa_pss_rsae_sha384 rsa_pss_rsae_sha512	secp256r1 secp384r1 secp521r1 x25519 x448
	TLSv1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_GCM_SHA384 TLS_RSA_WITH_AES_128_GCM_SHA256 TLS_RSA_WITH_AES_256_CBC_SHA256 TLS_RSA_WITH_AES_128_CBC_SHA256 TLS_RSA_WITH_AES_256_CBC_SHA TLS_RSA_WITH_AES_128_CBC_SHA	rsa_pss_rsae_sha256 rsa_pss_rsae_sha384 rsa_pss_rsae_sha512 rsa_pkcs1_sha256 rsa_pkcs1_sha384 rsa_pkcs1_sha512 rsa_pkcs1_sha224	

Component	Protocol	Ciphersuites	Signatures	Groups
NFM-P Web Server for Java Client connection (tcp/8444)	TLSv1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 TLS_DHE_RSA_WITH_AES_256_CBC_SHA TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 TLS_DHE_RSA_WITH_AES_128_CBC_SHA TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	rsa_pkcs1_sha1 rsa_pkcs1_sha224 rsa_pkcs1_sha256 rsa_pkcs1_sha384 rsa_pkcs1_sha512 rsa_pss_rsae_sha256 rsa_pss_rsae_sha384 rsa_pss_rsae_sha512	secp256r1 secp384r1 secp521r1 x25519 x448

10.1.2 TLS client algorithms for mediation networks

Supported TLS client algorithms for gRPC mediation policies.

Protocol	Ciphersuites	Signatures	Groups
TLSv1.3	TLS_AES_128_GCM_SHA256 TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256	ecdsa_secp256r1_sha256 rsa_pss_rsae_sha256 rsa_pkcs1_sha256 ecdsa_secp384r1_sha384 rsa_pss_rsae_sha384 rsa_pkcs1_sha384 rsa_pss_rsae_sha512 rsa_pkcs1_sha512 rsa_pkcs1_sha1	x25519 secp256r1 secp384r1 secp521r1
TLSv1.2	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA		
	TLS_RSA_WITH_AES_128_GCM_SHA256 TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA		N/A

10.1.3 SSH client algorithms for mediation networks

Supported SSH client algorithms for CLI and NETCONF mediation policies.

Mediation type	Crypto function	Default algorithms (non-FIPS)	Default algorithms (FIPS)	Configurable	Configuration procedure
Classic	Key exchange	curve25519-sha256 curve25519-sha256@libssh.org curve448-sha512 ecdh-sha2-nistp521 ecdh-sha2-nistp384 ecdh-sha2-nistp256 diffie-hellman-group-exchange-sha256 diffie-hellman-group18-sha512 diffie-hellman-group17-sha512 diffie-hellman-group16-sha512 diffie-hellman-group15-sha512 diffie-hellman-group14-sha256	ecdh-sha2-nistp521 ecdh-sha2-nistp384 ecdh-sha2-nistp256 diffie-hellman-group-exchange-sha256 diffie-hellman-group18-sha512 diffie-hellman-group17-sha512 diffie-hellman-group16-sha512 diffie-hellman-group15-sha512 diffie-hellman-group14-sha256	Yes	*NSP System Administrator Guide*
	Host key	ecdsa-sha2-nistp256-cert-v01@openssh.com ecdsa-sha2-nistp384-cert-v01@openssh.com ecdsa-sha2-nistp521-cert-v01@openssh.com ssh-ed25519-cert-v01@openssh.com rsa-sha2-512-cert-v01@openssh.com rsa-sha2-256-cert-v01@openssh.com ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 ssh-ed25519 rsa-sha2-512 rsa-sha2-256 ssh-rsa	ecdsa-sha2-nistp256-cert-v01@openssh.com ecdsa-sha2-nistp384-cert-v01@openssh.com ecdsa-sha2-nistp521-cert-v01@openssh.com ssh-ed25519-cert-v01@openssh.com rsa-sha2-512-cert-v01@openssh.com rsa-sha2-256-cert-v01@openssh.com ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 ssh-ed25519 rsa-sha2-512 rsa-sha2-256 ssh-rsa	Yes	*NSP System Administrator Guide*

Mediation type	Crypto function	Default algorithms (non-FIPS)	Default algorithms (FIPS)	Configurable	Configuration procedure
	Cipher	chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com aes128-cbc aes192-cbc aes256-cbc	chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com aes128-cbc aes192-cbc aes256-cbc	Yes	*NSP System Administrator Guide*
	MAC	hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha1-etm@openssh.com hmac-sha2-256 hmac-sha2-512 hmac-sha1	hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha1-etm@openssh.com hmac-sha2-256 hmac-sha2-512 hmac-sha1	Yes	*NSP System Administrator Guide*
Model-driven	Key exchange	ecdh-sha2-nistp521 ecdh-sha2-nistp384 ecdh-sha2-nistp256 diffie-hellman-group-exchange-sha256 diffie-hellman-group-exchange-sha1 diffie-hellman-group14-sha1 diffie-hellman-group14-sha256 diffie-hellman-group15-sha512 diffie-hellman-group16-sha512 diffie-hellman-group17-sha512 diffie-hellman-group18-sha512 diffie-hellman-group1-sha1	N/A	No	N/A

Mediation type	Crypto function	Default algorithms (non-FIPS)	Default algorithms (FIPS)	Configurable	Configuration procedure
	Host key	ecdsa-sha2-nistp256-cert-v01@openssh.com ecdsa-sha2-nistp384-cert-v01@openssh.com ecdsa-sha2-nistp521-cert-v01@openssh.com ssh-ed25519-cert-v01@openssh.com rsa-sha2-512-cert-v01@openssh.com rsa-sha2-256-cert-v01@openssh.com ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 ssh-ed25519 rsa-sha2-512 rsa-sha2-256 ssh-rsa	N/A	No	N/A
	Cipher	chacha20-poly1305@openssh.com aes256-ctr aes192-ctr aes128-ctr aes256-gcm@openssh.com aes128-gcm@openssh.com aes256-cbc aes192-cbc aes128-cbc	N/A	No	N/A
	MAC	hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha1-etm@openssh.com hmac-sha2-256 hmac-sha2-512 hmac-sha1	N/A	No	N/A