



## **Programmability Command Reference for Cisco 8000 Series Routers**

**First Published:** 2024-05-31

**Last Modified:** 2024-05-31

### **Americas Headquarters**

Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, CA 95134-1706  
USA  
<http://www.cisco.com>  
Tel: 408 526-4000  
800 553-NETS (6387)  
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020–2024 Cisco Systems, Inc. All rights reserved.



## CONTENTS

---

### PREFACE

#### **Preface** v

Changes to This Document v

Communications, Services, and Additional Information v

---

### CHAPTER 1

#### **gRPC Commands** 1

clear gnsi path authorization counters 2

grpc 3

gnmi 5

grpc max-concurrent-streams 7

grpc certificate common-name 8

grpc tunnel 9

grpc p4rt 11

grpc p4rt interface 12

grpc p4rt location 13

gnsi load service authorization policy 14

grpc gnsi service certz ssl-profile-id 15

port (gRPC) 16

show grpc certificate 17

show gnsi service authorization policy 19

show gnsi path authorization counters 21

show gnsi path authorization policy 23

show gnsi path authorization statistics 24

show p4rt devices 26

show p4rt interfaces 28

show p4rt state 30

show p4rt stats 31

show p4rt trace 33  
show ssh server 35  
show tech-support gnsi 40  
show gnsi trace pathz 41

---

**CHAPTER 2**      **NETCONF Commands** 43  
                  netconf-yang agent 44



## Preface

---

This preface contains these sections:

- [Changes to This Document](#), on page v
- [Communications, Services, and Additional Information](#), on page v

## Changes to This Document

This table lists the technical changes made to this document since it was first released.

*Table 1: Changes to This Document*

Date	Summary
May 2024	Initial release of this document.

## Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business impact you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco Marketplace](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).

### Cisco Bug Search Tool

[Cisco Bug Search Tool](#) (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.





## gRPC Commands

---

This module describes the commands used to use the gRPC Protocol to define network operations with data models.

For detailed information about gRPC concepts, configuration tasks, and examples, see the *Use gRPC Protocol to Define Network Operations with Data Models in the Cisco 8000 Series Router* module in the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

gRPC encodes requests and responses in binary. gRPC is extensible to other content types along with Protobuf. The Protobuf binary data object in gRPC is transported over HTTP/2.

- [clear gnsi path authorization counters, on page 2](#)
- [grpc, on page 3](#)
- [gnmi, on page 5](#)
- [grpc max-concurrent-streams, on page 7](#)
- [grpc certificate common-name, on page 8](#)
- [grpc tunnel, on page 9](#)
- [grpc p4rt, on page 11](#)
- [grpc p4rt interface, on page 12](#)
- [grpc p4rt location, on page 13](#)
- [gnsi load service authorization policy, on page 14](#)
- [grpc gnsi service certz ssl-profile-id, on page 15](#)
- [port \(gRPC\), on page 16](#)
- [show grpc certificate, on page 17](#)
- [show gnsi service authorization policy, on page 19](#)
- [show gnsi path authorization counters, on page 21](#)
- [show gnsi path authorization policy, on page 23](#)
- [show gnsi path authorization statistics, on page 24](#)
- [show p4rt devices, on page 26](#)
- [show p4rt interfaces, on page 28](#)
- [show p4rt state, on page 30](#)
- [show p4rt stats, on page 31](#)
- [show p4rt trace, on page 33](#)
- [show ssh server, on page 35](#)
- [show tech-support gnsi, on page 40](#)
- [show gnsi trace pathz, on page 41](#)

# clear gnsi path authorization counters

To clear the gNSI path authorization counters, use the **clear gnsi path authorization counters** command in Global Configuration mode.

**clear gnsi path authorization counters** [ **path** *XPath* | **server-name** *server-name* ]

<b>Syntax Description</b>	<i>XPath</i>	Provide the XPath for which authorization counters can be cleared.
	<i>server-name</i>	The server's IP address from where authorization counters can be cleared.

**Command Default** Enabled, by default

**Command Modes** Global Configuration mode

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.2.11	The command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the clearing of gNSI path authorization counters on the router:

```
Router# clear gnsi path authorization counters
Router#
```



# gRPC

To configure network devices and view operational data, use the **gRPC** command in the XR Config mode. To remove the **gRPC** protocol, use the **no** form of this command.

```
gRPC { address-family | certificate-authentication | dscp | max-concurrent-streams |
max-request-per-user | max-request-total | max-streams | max-streams-per-user | no-tls | tlsv1-disable
| tls-cipher | tls-mutual | tls-trustpoint | service-layer | vrf }
```

Syntax Description		
<b>address-family</b>		Specifies the address family identifier type.
<b>certificate-authentication</b>		It enables certificate-based authentication.
<b>dscp</b>		Specifies QoS marking DSCP on transmitted gRPC.
<b>max-concurrent-streams</b>		Specifies the limit on the maximum concurrent streams per gRPC connection to be applied on the server.
<b>max-request-per-user</b>		Specifies the maximum concurrent requests per user.
<b>max-request-total</b>		Specifies the maximum concurrent requests in total.
<b>max-streams</b>		Specifies the maximum number of concurrent gRPC requests. The maximum subscription limit is 128 requests. The default is 32 requests.
<b>max-streams-per-user</b>		Specifies the maximum concurrent gRPC requests for each user. The maximum subscription limit is 128 requests. The default is 32 requests.
<b>no-tls</b>		It disable transport layer security (TLS). The TLS is enabled by default.
<b>tlsv1-disable</b>		It disable TLS version 1.0
<b>tls-cipher</b>		It enable the gRPC TLS cipher suites.
<b>tls-mutual</b>		Specifies the mutual authentication.
<b>tls-trustpoint</b>		It configure trustpoint.
<b>service-layer</b>		It enable the gRPC service layer configuration.
<b>vrf</b>		It enable server vrf.

**Command Default** None

**Command Modes** XR Config mode

Command History	Release	Modification
	Release 24.1.1	This command was introduced.

**Usage Guidelines** This command is supported on Cisco IOS XR 64-bit OS.

Task ID	Task ID	Operations
	config-services	read

The following example shows how to enable gRPC over an HTTP/2 connection:

```
Router#configure
Router(config)#grpc
Router(config-grpc)#port <port-number>
```

# gnmi

To create a gRPC listener with the default or IANA ratified gNMI port of 9339, use the **gnmi** command in Global Configuration Mode.

**gnmi port** *portnum*

<b>Syntax Description</b>	<i>portnum</i>	Specifies the server listening port for the gRPC service. • gNMI service port: default: 9339, range: 57344-57999
<b>Command Default</b>	None	
<b>Command Modes</b>	Global Configuration Mode	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	This command was introduced.

**Usage Guidelines**

Unconfiguring gNMI will disable requests on port 9339.

The allowed ports within this range are 9339 (IANA ratified port) and 57344-57999 (Linux application port range)

<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	config-services	read, write

## Examples

The following example shows how to configure gNMI as a submode under gRPC and committing this configuration would create a gRPC listener with the default or IANA ratified gNMI port of 9339.

```
Router(config-grpc) gnmi
Router(config-grpc-gnmi) commit
```

Verify the submode configuration.

```
Router#show running-config grpc
grpc
  gnmi
!
```

The **port** command under gNMI submode allows the port to be modified in the port range or IANA ratified port.

```
Router(config-grpc) #gnmi
Router(config-grpc-gnmi) #port 9339
Router(config-grpc-gnmi) #commit
```

Verify the port number.

```
Router#show running-config grpc
grpc
```

```
gnmi
  port 9339
!
```

## grpc max-concurrent-streams

To specify a limit on the number of concurrent streams per gRPC connection to be applied on the server, use the **grpc max-concurrent-streams** command in the XR Config mode. To restore the default value, use the **no** form of this command.

```
grpc max-concurrent-streams limit
```

<b>Syntax Description</b>	<b>max-concurrent-streams</b> <i>limit</i>	Specifies the limit on the number of concurrent streams per gRPC connection to be applied on the server. The range is from 1 to 128. The command default is 32.
---------------------------	--	---

<b>Command Default</b>	By default, the maximum concurrent streams per gRPC connection is 32.
------------------------	---

<b>Command Modes</b>	XR Config mode
----------------------	----------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	This command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	config-services	read, write

### Examples

The following example shows how to set the limit of the number of concurrent streams per gRPC connection to 40:

```
Router#configure
Router(config)#grpc max-concurrent-streams 40
```

## grpc certificate common-name

To allow the router (tunnel client) to dial out to a collector (tunnel server), use the **grpc** command in the XR Config mode. To remove the **gRPC** service, use the **no** form of this command.

**grpc certificate common-name** *WORD*

<b>Syntax Description</b>	<i>WORD</i>	Specifies the common name when certificate is generated, default: <b>ems.cisco.com</b> .
---------------------------	-------------	--

<b>Command Default</b>	None
------------------------	------

<b>Command Modes</b>	XR Config mode
----------------------	----------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	This command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	config-services	read, write

### Examples

The following example shows how to specify a common-name:

```
Router(config)#grpc
Router(config-grpc)#certificate common-name
Router(config-grpc)#commit
```

# grpc tunnel

To allow the router (tunnel client) to dial out to a collector (tunnel server), use the **grpc tunnel** command in the . To remove the **gRPC tunnel** service, use the **no** form of this command.

```
grpc tunnel { destination IP-address domain name | port port-ID | address-family ipv4 ipv6
| target address | source ipv4 virtual ipv6 virtual }
```

Syntax Description		
<b>destination</b> <i>IP-address</i> <i>domain name</i>		Specifies the gRPC tunnel destination.
<b>port</b> <i>port-ID</i>		Specifies the destination port.
<b>address-family</b> <i>ipv4</i> <i>or</i> <i>ipv6</i>		Specifies the address-family (AF) for the returned addresses from DNS. Only applicable to domain name.
<b>target</b> <i>address</i>		Specifies the target name to register the tunnel service.
<b>source</b> <i>ipv4 virtual</i> <i>or</i> <i>ipv6 virtual</i>		Specifies the virtual IP address family.

**Command Default** None

**Command Modes**

Command History	Release	Modification
	Release 7.10.1	Keywords <b>source ipv4 virtual address</b> , <b>source ipv6 virtual address</b> , <b>address-family ipv4</b> , and <b>address-family ipv6</b> were added to this command.
	Release 7.5.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read, write

## Examples

The following example shows how to set up a virtual IPv4 or IPv6 or both as source address:

```
Router(config)#grpc
Router(config-grpc)#tunnel
Router(config-grpc-tunnel)#destination 192.168.0.1 port 59500
Router(config-grpc-tunnel-dest)#target xr
Router(config-grpc-tunnel-dest)#source ipv4 virtual address
Router(config-grpc-tunnel-dest)#source ipv6 virtual address
Router(config-grpc-tunnel-dest)#source-interface MgmtEth 0/RP0/CPU0/0
```

The following example shows how to set up FQDN as gRPC tunnel destination (IPv4):

```
Router#config
Router(config)#grpc
```

```
Router(config-grpc)#tunnel  
Router(config-grpc-tunnel)#destination test.tunnel.dn port 59500  
Router(config-grpc-tunnel-dest)#address-family ipv4  
Router(config-grpc-tunnel-dest)#target xr  
Router(config-grpc-tunnel-dest)#commit
```



## grpc p4rt

To enable programming the data plane elements using Programming Protocol-independent Packet Processors (P4) Runtime API, use the **grpc p4rt** command in the . To remove the P4Runtime API, use the **no** form of this command.

**grpc p4rt**

**Syntax Description** This command has no keywords or arguments.

**Command Default** None

**Command Modes**

Command History	Release	Modification
	Release 7.10.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read, write

**Examples**

The following example shows how to enable P4Runtime service:

```
RP/0/# configure
RP/0/(config)# grpc p4rt
RP/0/(config-grpc-p4rt)# commit
```

## grpc p4rt interface

To assign unique port identifiers to configure P4Runtime programming on the router, use the **grpc p4rt interface** command in the XR Config mode. To remove the P4Runtime port identifier configuration for the interfaces, use the **no** form of this command.

```
grpc p4rt interface type location port-id port-identifier
```

Syntax Description	Parameter	Description
	<i>type</i>	Specifies the interface type. For more information, use the question mark ( ? ) online help function.
	<i>location</i>	Specifies the physical or virtual interface in <i>rack/slot/instance/port/breakout</i> or <i>rack/slot/interface/port</i> format.
	<b>port-id</b> <i>port-identifier</i>	Assigns a unique numeric identifier to each physical port on the router. The port ID is a unique 32-bit identifier. The range is 1 to 4294967039.

**Command Default** None

**Command Modes** XR Config mode

Command History	Release	Modification
	Release 7.10.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read, write

### Examples

The following example shows how to configure the interfaces HundredGigE0/0/0/24, HundredGigE0/0/0/25 and HundredGigE0/0/0/26 with port IDs 3, 6 and 7 respectively for P4Runtime:

```
RP/0/# configure
RP/0/(config)# grpc p4rt
RP/0/(config-grpc-p4rt)# location 0/0/CPU0 npu-id 0 device-id 1000000
RP/0/(config-grpc-p4rt)#location 0/0/CPU0 npu-id 1 device-id 1000001
RP/0/(config-grpc-p4rt)#location 0/1/CPU0 npu-id 2 device-id 1000002
RP/0/(config-grpc-p4rt)#location 0/1/CPU0 npu-id 3 device-id 1000011
```

## grpc p4rt location

To assign unique identifiers for each Network Processing Unit (NPU) in the system to configure P4Runtime programming on the router, use the **grpc p4rt location** command in the XR Config mode. To remove the P4Runtime device identifier configuration for the NPUs, use the **no** form of this command.

```
grpc p4rt location node-id npu-id npu-identifier device-id device-identifier
```

<b>Syntax Description</b>	<i>node-id</i>	Specifies the card location on the specified node in <i>rack/slot/module</i> notation.
	<b>npu-id</b> <i>npu-identifier</i>	Specifies the NPU identifier on the card. The <i>npu-id</i> is a unique value in the range of 0 to 7.
	<b>device-id</b> <i>device-identifier</i>	Assigns a unique device identifier to each device in the system. The <i>device-id</i> is a unique 64-bit identifier. The range is 1 to 18446744073709551615.
<b>Command Default</b>	None	
<b>Command Modes</b>	XR Config mode	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 7.10.1	This command was introduced.
<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.	
<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	config-services	read, write

### Examples

The following example shows how to configure the NPU ID and device ID for nodes 0/0/CPU0 and 0/1/CPU0:

```
RP/0/RP0/CPU0:router## configure
RP/0/RP0/CPU0:router(config)# grpc p4rt
RP/0/RP0/CPU0:router(config-grpc-p4rt)# location 0/0/CPU0 npu-id 0 device-id 1000000
RP/0/RP0/CPU0:router(config-grpc-p4rt)# location 0/0/CPU0 npu-id 1 device-id 1000001
RP/0/RP0/CPU0:router(config-grpc-p4rt)# location 0/1/CPU0 npu-id 2 device-id 1000002
RP/0/RP0/CPU0:router(config-grpc-p4rt)# location 0/1/CPU0 npu-id 3 device-id 1000011
```

# gnsi load service authorization policy

To instruct the router to load the service authorization policy file into its memory and update the policy, use the **gnsi load service authorization policy** command in Global Configuration Mode.

**gnsi load service authorization policy** *file\_path*

<b>Syntax Description</b>	<i>file-path</i> Specifies the path of the policy file.
---------------------------	---

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration Mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 7.11.1	This command was introduced.

<b>Usage Guidelines</b>	A policy file which has no specified or the policy is invalid, the default behavior will transition to the zero-policy behavior. Zero-policy allows all gRPC services to all the users if their profiles are configured.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read, write

This example shows how to activate the authorization policy test.json in the router.

```
Router(config)#gnsi load service authorization policy /disk0:/test.json
Successfully loaded policy
```

## grpc gnsi service certz ssl-profile-id

To instruct the router to load the certz.proto, use the **grpc gnsi service certz ssl-profile-id** command in Global Configuration Mode. To disable the SSL profiles configured with certz.proto, use the no form of the command.

**grpc gnsi service certz ssl-profile-id** *ssl-profile name*

<b>Syntax Description</b>	<i>ssl-profile name</i> Specifies the SSL-profile name for which certz. proto needs to be activated.
---------------------------	--

<b>Command Default</b>	None
------------------------	------

<b>Command Modes</b>	Global Configuration Mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	This command was introduced.

<b>Usage Guidelines</b>	If Certz. proto is not active, then gNOI cert.proto is taken into consideration. If neither certz.proto nor cert.proto is active, then TLS trustpoint's data is considered.
-------------------------	---

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read, write

This example shows how to activate the certz.proto in the router.

```
Router(config)#grpc gnsi service certz ssl-profile-id gNxi
Router(config)#commit
```

## port (gRPC)

To set custom ports for gNMI, gRIBI, and P4RT services within the defined range, including default IANA ports like 9339, 9340, and 9559 (respectively), use the **port** command under the service submode.

**port** *portnum*

<b>Syntax Description</b>	<i>portnum</i>	Specifies the server listening port for the gRPC service. <ul style="list-style-type: none"> <li>• gNMI service port: default: 9339, range: 57344-57999</li> <li>• gRIBI service port: default: 9340, range: 57344-57999</li> <li>• p4RT service port: default: 9559, range: 57344-57999</li> </ul>
---------------------------	----------------	---

<b>Command Default</b>	None
------------------------	------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	This command was introduced.

**Usage Guidelines** Disabling the **port** command will cause the service to use the default or IANA port.

<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	config-services	read, write

### Examples

The following example shows how to configure a port for any available gRPC service (gNMI, P4RT, gRIBI) :

#### For P4RT service:

```
Router(config-grpc) #p4rt
Router(config-grpc-p4rt) #port 9559
Router(config-grpc-p4rt) #commit
```

Verify the port number.

```
Router#show running-config grpc
grpc
  p4rt
    port 9559
!
```

# show grpc certificate

To display the active gRPC certificate management policies on the router, use the **show grpc certificate** command in EXEC mode.

## show grpc certificate

<b>Syntax Description</b>	This command has no keywords or arguments.	
<b>Command Default</b>	None	
<b>Command Modes</b>	EXEC mode	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.1.1	The command was introduced.
<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.	
<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the active gRPC certificate management policies on the router. The below-mentioned command output is truncated version.

```
Router#show grpc certificate
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number: 32 (0x20)
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: CN=localhost,O=OpenConfig,C=US
    Validity
      Not Before: Nov  8 08:49:38 2023 GMT
      Not After : Mar 22 08:49:38 2025 GMT
    Subject: CN=ems,O=OpenConfig,C=US
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public-Key: (4096 bit)
      Modulus:
        00:ea:6a:6c:25:be:9f:15:71:ce:74:89:03:ec:ef:
        0b:3b:de:58:a8:7e:28:b8:cf:b3:82:91:b4:5c:42:
        e7:d8:28:98:35:bd:35:60:a7:4e:f8:77:02:46:5f:
        27:a4:16:cf:3c:e3:24:28:69:9c:22:1e:e3:52:96:
        71:87:7c:40:0c:1f:dd:30:ea:dc:40:ca:93:00:54:
        5e:de:20:54:5b:f4:2f:9f:19:6f:71:61:28:69:3d:
        97:26:ab:e1:5f:53:3c:f1:a2:c3:14:f4:01:90:1a:
        .
        .
        .
```

```
      Exponent: 65537 (0x10001)
X509v3 extensions:
  X509v3 Key Usage: critical
    Digital Signature
  X509v3 Extended Key Usage:
    TLS Web Client Authentication, TLS Web Server Authentication
  X509v3 Authority Key Identifier:
    keyid:0A:A8:9A:6A:23:34:AE:CA:96:00:2C:F3:04:38:14:E3:D4:8D:77:BD

  X509v3 Subject Alternative Name:
    DNS, IP Address:64.103.223.56
Signature Algorithm: sha256WithRSAEncryption
b9:89:ec:60:3d:8d:7d:9c:dc:08:56:89:99:44:92:98:45:b6:
97:ba:e3:e5:f2:48:b2:44:8d:db:23:bb:a1:c0:62:79:78:18:
d7:55:f6:4a:67:5b:75:e0:c0:0b:52:51:07:36:d5:6c:c7:67:
48:86:8d:dd:70:1c:9f:7c:a1:7b:aa:a5:4e:e1:ad:cf:4c:e5:
81:db:92:cf:88:70:5a:1c:8d:de:0d:e8:b3:05:de:b9:04:4d:
23:e1:de:66:e5:08:bd:2e:31:0a:07:a6:c0:00:3a:38:2f:00:
.
.
.
```



# show gnsi service authorization policy

To display the active gRPC service authorization policies on the router, use the **show gnsi service authorization policy** command in Global Configuration mode.

## show gnsi service authorization policy

<b>Syntax Description</b>	This command has no keywords or arguments.
---------------------------	--

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 7.11.1	The command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the policy which is active on the router:

```
Router#show gnsi service authorization policy
Wed Jul 19 10:56:14.509 UTC{
  "version": "1.0",
  "created_on": 1700816204,
  "policy": {
    "name": "authz",
    "allow_rules": [
      {
        "name": "allow all gNMI for all users",
        "request": {
          "paths": [
            "*"
          ]
        },
        "source": {
          "principals": [
            "*"
          ]
        }
      }
    ],
    "deny_rules": [
      {
        "name": "deny gNMI set for oper users",
        "request": {
          "paths": [
            "/gnmi.gNMI/*"
          ]
        }
      }
    ]
  }
}
```

```
    ]
  },
  "source": {
    "principals": [
      "User1"
    ]
  }
}
]
```

# show gnsi path authorization counters

To view the gNSI path authorization counters such as number of accepted, rejected authorizations, use the **show gnsi path authorization counters** command in Global Configuration mode.

```
show gnsi path authorization counters [ path XPath | server-name server-name ]
```

<b>Syntax Description</b>	<i>XPath</i> Provide the XPath for which authorization counters can be retrieved.				
	<i>server-name</i> The server's IP address from where authorization counters can be retrieved.				
<b>Command Default</b>	Enabled, by default				
<b>Command Modes</b>	Global Configuration mode				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Release 24.2.11</td> <td>The command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Release 24.2.11	The command was introduced.
Release	Modification				
Release 24.2.11	The command was introduced.				
<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.				
<b>Task ID</b>	<table border="1"> <thead> <tr> <th>Task ID</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>config-services</td> <td>read</td> </tr> </tbody> </table>	Task ID	Operation	config-services	read
Task ID	Operation				
config-services	read				

This example displays the policy which is active on the router:

```
Router# show gnsi path authorization counters
Mon Apr 1 08:05:46.297 UTC
-----Pathz Counters Info-----

/system/config/hostname:
Rejects :                               Read                               Write
  Last :                               N/A                               N/A
Accepts :                               0                                 3
  Last :                               N/A    Mon, 01 Apr 2024 08:05:25 +0000
Total path records received 1
```

```
Router# show gnsi path authorization counters server-name 64.103.223.33
Mon Apr 1 08:33:25.194 UTC
-----Pathz Counters Info-----

/:
Rejects :                               Read                               Write
  Last :                               N/A    Mon, 01 Apr 2024 08:32:37 +0000
Accepts :                               0                                 0
  Last :                               N/A                               N/A

/system/config/hostname:
```

## show gnsi path authorization counters

```

Read                               Write
Rejects :                          0                               6
  Last :                            N/A   Mon, 01 Apr 2024 08:32:36 +0000
Accepts :                          0                               0
  Last :                            N/A                               N/A
Total path records received 2
Router#

```

```
Router# show gnsi path authorization counters path /system/config/hostname
```

```
Mon Apr 1 08:32:46.468 UTC
```

```
-----Pathz Counters Info-----
```

```
/system/config/hostname:
```

```

Read                               Write
Rejects :                          0                               6
  Last :                            N/A   Mon, 01 Apr 2024 08:32:36 +0000
Accepts :                          0                               0
  Last :                            N/A                               N/A
Total path records received 1
Router#

```

# show gnsi path authorization policy

To view the running gNSI path authorization policy on the router, use the **show gnsi path authorization policy** command in Global Configuration mode.

## show gnsi path authorization policy

<b>Syntax Description</b>	This command has no keywords or arguments.
---------------------------	--

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.2.11	The command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the gnsi path authorization policy which is active on the router:

```
Router# show gnsi path authorization policy
Mon Apr 1 04:29:37.905 UTC
version:"1" created_on:1711946719670313 policy:{rules:{user:"cafyauto"
path:{origin:"openconfig" elem:{name:"system"} elem:{name:"config"} elem:{name:"hostname"}}
action:ACTION_PERMIT mode:MODE_WRITE}}
Router#
```

## show gnsi path authorization statistics

To view the gNSI path authorization statistics on the router, use the **show gnsi path authorization statistics** command in Global Configuration mode.

### show gnsi path authorization statistics

<b>Syntax Description</b>	This command has no keywords or arguments.
---------------------------	--

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.2.11	The command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the gNSI path authorization statistics on the router:

```
Router# show gnsi path authorization statistics
Mon Apr  1 04:29:23.259 UTC
-----Pathz Info-----
Engine:

State:
  Active Policy:
    Version           : 1
    Created On (UTC)  : Wed, 09 Dec 54251401 07:58:33 +0000
  Sandbox Policy:
    Version           : N/A
    Created On (UTC)  : N/A
  Policy Rotation in Progress: False

Stats:
  Rotations in Progress Count: 0
  Policy Rotations           : 0
  Policy Rotation Errors     : 0
  Policy Upload Requests     : 0
  Policy Upload Errors       : 0
  Policy Finalize            : 0
  Policy Finalize Errors     : 0
  Probe Requests             : 0
  Probe Errors                : 0
  Get Requests                : 0
  Get Errors                  : 0
  Policy Unmarshall Errors   : 0
```

```
Sandbox Policy Errors      : 0

Counters:
No Policy Auth Requests   : 0
gNMI Path Leaves         : 0
gNMI Authorizations       : 0
gNMI Set Path Permit     : 0
gNMI Set Path Deny       : 0
gNMI Get Path Permit     : 0
gNMI Get Path Deny       : 0

Errors:
Path To String            : 0
Origin Type               : 0
Bad Mode                  : 0
Bad Action                 : 0
JSON Flatten              : 0
String To Path            : 0
Join Paths                : 0
Nil Path                  : 0
Nil SetRequest            : 0
Empty User                 : 0
Probe Internal            : 0
Path Counters:
  Increment                : 0
  Find                     : 0
  Clear                    : 0
  Walk                     : 0
```

# show p4rt devices

To view the status of P4Runtime devices, use the **show p4rt devices** command in EXEC mode.

```
show p4rt devices device-id location npu-location npu-id npu-id
```

Syntax Description	Parameter	Description
	<i>device-id</i>	Specifies the 64-bit device identifier as a decimal value in the range of 1 to 18446744073709551615.
	<b>location</b> <i>npu-location</i>	Specifies the location of the Network Processing Unit (NPU) device.
	<b>npu-id</b> <i>npu-id</i>	Specifies the unique NPU identifier in the range of 0 to 7.

**Command Default** None

**Command Modes** EXEC mode

Command History	Release	Modification
	Release 7.10.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read

This example shows how to view the status of devices configured for P4Runtime:

```
RP/0/# show p4rt devices
Wed May 17 17:11:43.670 UTC
-----P4RT Devices-----
Device Id                               : 1000000
  Node Id                               : 0/0/CPU0 (0x0)
  NPU Id                                 : 0x0
  Internal Tx State                       : 1
  Max Election Id                         : 0,0
  Shutdown Requested                      : no
  Sessions count                          : 0
  P4Info Hash Value                       : 0x0
  P4Info Ref Count                        : 0
  Protocol Stats:
    New Primary Count                     : 0
    Last Session Id                       : 0
    Successfull FwdConfig                  : 0
    Unsuccessfull FwdConfig               : 0
    Not Primary FwdConfig                  : 0
  Write Stats:
    Successfull Write                      : 0
    Unsuccessfull Write                    : 0
    Not Primary Write                      : 0
    Failed Precondition Write              : 0
```



```

        Successfull Write Entries      : 0
        Unsuccessfull Write Entries    : 0
    Read Stats:
        Successfull Read                : 0
        Unsuccessfull Read              : 0
        Failed Precondition Read        : 0
        Successfull Read Entries        : 0
    Inject Stats:
        Primary Packets                 : 0
        Primary Drops                   : 0
        Failed Precondition             : 0
        Non Primary Drops               : 0
        Bad Packet Length               : 0
        Bad Packet Metadata             : 0
    Punt Queue Stats:
        Size                            : 0
        Inserted                        : 0
        Removed                         : 0
        Full Drops                      : 0
        Drained Drops                   : 0
    Punt Stats:
        Total Primary Packets           : 0
        Primary Packet Errors           : 0

    Table Entries                      : 0

    Sessions:
        None found

    Device Id                          : 1000001
    Node Id                             : 0/1/CPU0 (0x100)
    NPU Id                              : 0x3
    Internal Tx State                   : 1
    Max Election Id                    : 0,0
    Shutdown Requested                 : no
    ----- Truncated for brevity -----

```

# show p4rt interfaces

To view the status of P4Runtime interfaces, use the **show p4rt interfaces** command in .

```
show p4rt interfaces type location
```

<b>Syntax Description</b>	<i>type</i> Specifies the interface type. For more information, use the question mark ( ? ) online help function.				
	<i>location</i> Specifies the physical or virtual interface in <i>rack/slot/instance/port/breakout</i> or <i>rack/slot/interface/port</i> format.				
<b>Command Default</b>	None				
<b>Command Modes</b>	XR EXEC mode				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Release 7.10.1</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Release 7.10.1	This command was introduced.
Release	Modification				
Release 7.10.1	This command was introduced.				
<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.				
<b>Task ID</b>	<table border="1"> <thead> <tr> <th>Task ID</th> <th>Operations</th> </tr> </thead> <tbody> <tr> <td>config-services</td> <td>read</td> </tr> </tbody> </table>	Task ID	Operations	config-services	read
Task ID	Operations				
config-services	read				

This example shows how to view the status of interfaces configured for P4Runtime:

```
RP/0/# show p4rt interfaces
Wed May 17 16:53:01.459 UTC
-----P4RT Interface-----
Interface Name      : HundredGigE0/0/0/24
  Handle            : 0x250
  P4RT Port-id      : 3
  Node-id           : 0/0/CPU0 (0x0)
  NPU-id            : 0x0
  FSM State         : SPIO_ATTACHED
  RefCnt            : 3
  Flags             : 0xd

Interface Name      : HundredGigE0/0/0/25
  Handle            : 0x258
  P4RT Port-id      : 6
  Node-id           : 0/0/CPU0 (0x0)
  NPU-id            : 0x1
  FSM State         : SPIO_ATTACHED
  RefCnt            : 3
  Flags             : 0xd

Interface Name      : HundredGigE0/0/0/26
  Handle            : 0x260
  P4RT Port-id      : 7
  Node-id           : 0/0/CPU0 (0x0)
  NPU-id            : 0x1
```

```
FSM State      : SPIO_ATTACHED
RefCnt         : 3
Flags          : 0xd
```

## show p4rt state

To view the global state of P4Runtime gRPC service configured on the router, use the **show p4rt state** command in .

**show p4rt state**

**Syntax Description** This command has no keywords or arguments.

**Command Default** None

**Command Modes** XR EXEC mode

Command History	Release	Modification
	Release 7.10.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read

This example shows how to view the global state of P4Runtime service configured on the router:

```
RP/0/# show p4rt state
Wed May 17 17:24:56.802 UTC
-----P4RT state-----
Global:
  Thread cernno           : Success
  State                   : CONFIGURED
  Configured              : Yes

Interface Manager:
  Connected               : Yes

SPIO:
  Initialized              : Yes
  Thread cernno           : Success
  Thread running          : Yes
  Thread asked to stop    : No
  Resync in Progress      : No

NETIO:
  Connected               : Yes

LPTS:
  Client cernno           : Success
```

## show p4rt stats

To view the P4Runtime statistics, use the **show p4rt stats** command in .

**show p4rt stats**

**Syntax Description** This command has no keywords or arguments.

**Command Default** None

**Command Modes** XR EXEC mode

Command History	Release	Modification
	Release 7.10.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	config-services	read

This example shows how to view the global state of P4Runtime services configured on the router:

```
RP/0/# show p4rt stats
Wed May 17 17:34:14.611 UTC
-----P4RT stats-----
Global:
  Ifname Objects           : 3
  ID Objects               : 3
  IfHandle Objects        : 3
  Stale Interface Objects  : 0
Inject Stats:
  Added to Internal Queue  : 0
  Internal Queue Full Drops : 0

SPIO:
  Interface Attach OK      : 3
  Interface Attach Error   : 0
  Interface Resync OK      : 0
  Interface Resync Error   : 0
Punt Stats:
  Packets                  : 0
  Added to Device Queue    : 0
  Ifhandle Errors          : 0
  Egress Ifhandle Lookup Errors: 0
  Egress Ifhandle Errors   : 0
  Packet Len Errors        : 0
  Bad Punt Reason Errors   : 0
  Packet Buf Errors        : 0
  Bad Device Errors        : 0
  Device Queue Full Drops  : 0
Inject Stats:
  SPIO Errors              : 0
```

```
        SPIO Delivered           : 0

NETIO:
  Inject Stats:
    Bad Packet Len Errors       : 0
    Packet Buffer Memory Error   : 0
    Bad IP Packet Error         : 0
    Pak API Error               : 0
    Netio Send Error            : 0
    Netio Down Error            : 0
    Netio Delivered             : 0

LPTS:
  Write:
    Attempts                     : 0
    Errors                       : 0
    Entries:
      Attempts                   : 0
      Errors                     : 0
      Skipped (gRPC Parse)       : 0
      Opcode Errors              : 0
      Punt type Errors           : 0
      Not Supported Punt type    : 0
      LPTS Client Errors         : 0
      LPTS Client Success        : 0
  Read:
    Attempts                     : 0
    Errors                       : 0
    Entries:
      Destination Errors         : 0
      Node_id Errors             : 0
      Npu_id Errors              : 0
      Attribute Errors           : 0
      Read                       : 0
```

# show p4rt trace

To view the trace information of P4Runtime configuration, use the **show p4rt trace** command in .

```
show p4rt trace { all | lib }
```

<b>Syntax Description</b>	<b>all</b> Displays trace data for all P4Runtime library.				
	<b>lib</b> Displays trace data for general P4Runtime library.				
<b>Command Default</b>	None				
<b>Command Modes</b>	XR EXEC mode				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Release 7.10.1</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Release 7.10.1	This command was introduced.
Release	Modification				
Release 7.10.1	This command was introduced.				
<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.				
<b>Task ID</b>	<table border="1"> <thead> <tr> <th>Task ID</th> <th>Operations</th> </tr> </thead> <tbody> <tr> <td>config-services</td> <td>read</td> </tr> </tbody> </table>	Task ID	Operations	config-services	read
Task ID	Operations				
config-services	read				

This example shows how to view the trace information for P4Runtime configuration:

```
RP/0/# show p4rt trace all
Wed May 17 17:40:28.774 UTC
111 wrapping entries (6528 possible, 896 allocated, 0 filtered, 111 total)
May 17 15:08:47.499 p4rt/lib_slow 0/RP0/CPU0 t18073 Code(224) Thread Init: 'Slow Trace
Started'
May 17 15:08:47.499 p4rt/lib_slow 0/RP0/CPU0 t18073 Code(249) Thread Init: Parent 'thread
Barrier WAITING'
May 17 15:08:47.502 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(218) Thread Init: 'p4rt thread
EVMGR ok'
May 17 15:08:47.502 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(219) Thread Init: 'Role pulse
handler attached'
May 17 15:08:47.502 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(256) Role changed to: 'ACTIVE'
May 17 15:08:47.502 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(226) Thread Init: 'p4rt thread
debug ok'
May 17 15:08:47.502 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(240) DB: 'DB Initialized ok'
May 17 15:08:47.512 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(232) EDM Init: 'EDM sysdb reg ok'
May 17 15:08:47.512 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(233) EDM Init: 'EDM conn id ok'
May 17 15:08:47.512 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(356) OC EDM: OC EDM connect
May 17 15:08:47.522 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(359) OC EDM: Interface EDM
registration successful
May 17 15:08:47.522 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(360) OC EDM: Platform EDM register
May 17 15:08:47.529 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(361) OC EDM: Platform EDM
registration successful
May 17 15:08:47.529 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(238) OC EDM: Conn Success
May 17 15:08:47.532 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(269) LPTS: 'LPTS client init OK'
May 17 15:08:47.532 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(257) Event: 'Client Connections
Init'
```

```
May 17 15:08:47.535 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(236) IfMgr: 'IM callback registered'  
May 17 15:08:47.535 p4rt/lib_event 0/RP0/CPU0 t18092 IfMgr: Code(4) - 'Connection UP'  
May 17 15:08:47.535 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(238) IfMgr: 'Conn Success'  
May 17 15:08:47.535 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(243) SPIO: 'spio Mutex ok'  
May 17 15:08:47.535 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(244) SPIO: 'spio thread EVMGR ok'  
May 17 15:08:47.535 p4rt/lib_slow 0/RP0/CPU0 t18092 Code(227) Thread Init: 'Stop pulse  
handler attached'  
----- Truncated for brevity -----
```



## show ssh server

To view the SSH server configuration and the host-certs, use the **show ssh server** command in the EXEC mode.

```
show ssh server { vrf vrf-name | configuration | | gnsi configuration | | authorized_keys user
user-name | authorized_principals user user-name | ca_keys | host_keys | host-certs }
```

Syntax Description		
<b>vrf</b> <i>vrf-name</i>		Displays all the active configurations on the router for a given VRF
<b>gnsi</b>		Displays all the finalized configurations on the router for a given VRF.
<b>authorized_keys user</b> <i>user-name</i>		Displays a user's public keys for authentication.
<b>authorized_principals user</b> <i>user-name</i>		Displays the list of accepted principal names for a user's certificate authentication.
<b>ca_keys</b> <i>user-name</i>		Displays the trusted certificate authorities' public keys for user authentication.
<b>host_keys</b>		Displays various SSH private host keys (rsa, ecda, dsa, ed25519) if set up.
<b>host-certs</b>		Displays different public host certificates that match with the private host keys shown in <b>host_keys</b> .

**Command Default** None

**Command Modes** EXEC mode

Command History	Release	Modification
	Release 24.2.11	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

### Example

The following example shows the gNSI configurations on the router:

```
Router# config
Router(config)# grpc port 57888
Router(config)# grpc no-tls
Router(config)# commit
Router(config)# end
```

To view the gNSI configuration on the router, use the **show ssh server gnsi configuration** command.

```

Router# show ssh server gnsi configuration
Wed May 1 14:45:29.008 UTC
-----
AuthorizedKeysFile /etc/cisssh/authorized_list/%u/authorized_keys
AuthorizedPrincipalsFile /etc/cisssh/authorized_list/%u/authorized_principals
HostCertificate /etc/cisssh/host_certs/ecdsa-sha2-nistp256-cert.pub
HostCertificate /etc/cisssh/host_certs/ecdsa-sha2-nistp521-cert.pub
HostCertificate /etc/cisssh/host_certs/ed25519-cert.pub
-----

```

The following example shows the VRF configurations on the router:

```

Router# config
Router(config)# ssh server vrf default
Router(config)# commit
Router(config)# end

```

To view the server VRF configuration on the router, use the **show ssh server vrf** command.

```

Router# show ssh server vrf default configuration
-----
UsePAM yes
HostKeyAlgorithms
x509v3-ssh-rsa,ssh-rsa-cert-v01@openssh.com,ssh-ed25519-cert-v01@openssh.com,
ecdsa-sha2-nistp521-cert-v01@openssh.com,ecdsa-sha2-nistp256-cert-v01@openssh.com,
ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,ssh-ed25519,rsa-sha2-512,
rsa-sha2-256,ssh-rsa,ssh-dss
PermitRootLogin yes
MaxAuthTries 20
MaxSessions 16
RekeyLimit 1024M 60m
Subsystem sftp /pkg/bin/sftp-server
MACs hmac-sha2-512,hmac-sha2-256,hmac-sha1
LoginGraceTime 30
ClientAliveInterval 60
AllowTcpForwarding no
MaxStartups 150
LogLevel DEBUG
IPQoS 0x40
HostKey /pkg/ecdsa-sha2-nistp256
HostKey /pkg/ecdsa-sha2-nistp384
HostKey /pkg/ecdsa-sha2-nistp521
HostKey /pkg/ed25519
HostKey /pkg/rsa
HostKey /pkg/dsa
HostKey /pkg/x509v3-ssh-rsa
HostKey /pkg/ssh-rsa-cert-v01
AcceptedAlgorithms x509v3-ssh-rsa,x509v3-ecdsa-sha2-nistp256,x509v3-ecdsa-sha2-nistp384,
x509v3-ecdsa-sha2-nistp521,x509v3-ssh-dss,ssh-rsa,ssh-rsa-cert-v01@openssh.com,
rsa-sha2-256-cert-v01@openssh.com,rsa-sha2-512-cert-v01@openssh.com,
ecdsa-sha2-nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,
ecdsa-sha2-nistp521-cert-v01@openssh.com,ssh-dss-cert-v01@openssh.com,ssh-ed25519-cert-v01@openssh.com,
ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,ssh-ed25519,rsa-sha2-256,rsa-sha2-512,ssh-rsa
Port 22
PidFile /var/run/sshd_default.pid

```

To view the server host-keys on the router, use the **show ssh server host-keys** command.

```

Router# show ssh server host-keys
Wed May 1 14:39:36.746 UTC
-----
Key label: the_default
Type      : ED25519
Data      : ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILMXlhKk4HixCE/HGwKgbGwL7ecm0fze7ZsQQIjW

```

```
xxxx@vxr-slurm-146.xxxx.com
```

```
Key label: the_default
```

```
Type      : ECDSA General Curve Nistp256
```

```
Degree    : 256
```

```
Data      : ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBA9mwzn5O1+
oV5m6Zdo3Mqmc6IjkkxrCbt+E/vhK67/B8mEaGE05JfFcJ7zHp905HsiLm0mYijS4zQCZNYRMcvNk=
xxxx@vxr-slurm-146.xxxx.com
```

```
Key label: the_default
```

```
Type      : ECDSA General Curve Nistp521
```

```
Degree    : 521
```

```
Data      : ecdsa-sha2-nistp521
```

```
AAAAE2VjZHNhLXNoYTItbmlzdHA1MjEAAAIAAAAAAAAAAAAAIbmlzdHA1MjEAAACFBABjiqUtIXeBafO
sur6xhCaX0865nf6GpOgIQc/DzBNC1AJTtqZfQ14FMHPTkixAsHz/7OVSh70tMgk4VzCHH+EmpAB5zIrz7fSzJFXSs9DjQw
75DxtOsjb/mcovLnHU2wfsiDD7qOjhyznL/VlAkKRq60aFK9w4r0qWW5L/infNDoDfvg==
xxxx@vxr-slurm-146.xxxx.com
```

```
-----
Router#
```

To view the host certificates on the router, use the **show ssh server host-certs** command.

```
Router# show ssh server host-certs
```

```
Wed May 1 13:56:21.596 UTC
```

```
-----
Type : ecdsa-sha2-nistp521-cert
```

```
Data : ecdsa-sha2-nistp521-cert-v01@openssh.com AAAAKGVjZHNhLXNoYTItbmlzdHA1MjEt
Y2VydC12MDFab3B1bnNzaC5jb20AAAAgKjWh4uPFNKIr4uZV5maPUoOfyys/ncTyMpBbQZX+7KMAAAAI
bmlzdHA1MjEAAACFBABjiqUtIXeBafOsur6xhCaX0865nf6GpOgIQc/DzBNC1AJTtqZfQ14FMHPTk
ixAsHz/7OVSh70tMgk4VzCHH+EmpAB5zIrz7fSzJFXSs9DjQw75DxtOsjb/mcovLnHU2wfsiDD7qOjhyznL/
VlAkKRq60aFK9w4r0qWW5L/infNDoDfvgAAAAAAAAAAAAAAAAAAgAAAVjaXNjbwAAAAAAAAAAAAZQxHgAAAAB
nll0eAAAAAAAAAAAAAAAAAAABFwAAAAdzc2gtcnNhAAAAAwEAAQAAAQEA26xFTM/0hz1cDKmg6q17s81k+
UqOqEm6FUytpKw/apd4cBFNxBGWO5BaitQjTWSDLik9+rxmBF+vpBh4fScT64WDFHUx0OX9URAD14cyK21
z1KUP7L607ypurZDqmsLuNHYH+nQgwCBJKQzd6/Ph2iuYxY5xhDCG8FzSrxyoM1tHrL7gCey9fd08+Jl
dTMAAdqp8SvCvJjJcKuj0GJ68ut3pI4j0xZCTIMvQQ6ZmW5JgemN7xJLMUN4ZzJjGT1o1dKq5kMEVP8pOk8
yLIQkOyRcmuN1BW126D/W58dYXdY5z/OcYwZTBQ1SSIE+Lwb20RktJfVqrYn1aNaq/f38KdyYVQAAARQAAAAA
cnNhLXNoYTItNTEyAAABAic35ctjmPfoB3RRc3bD9gvHRzKzIO5mGbHxeH06qrNFyDxjP/A02Qyd1lRU1qjeH/
REAI38/RhUInEj75Iwi+f349xZx0bGacULZHMJWPYy2cGgx3e4WLF43Z3Zu09xSNzVCCUea71d21JhJGUAMWG1
ak86RLbObvAeSyYCCUG+jdNDBq7dfiaej05DvY33RRszfEf/4Cy6X8GyzyB/V0bmjrC1lUkb56JNsCNyweWCB
je2da5BwqxSbQUaLkD97Lad1Jjjeo8A/qxRMMVm71e9AAmlhtKtlUusqEawW1KmeZ4rbUkyTOJ3NaxdW/gEs4
uuAh58oweCaZyasv3ay0= lavms@vxr-slurm-146.cisco.com
```

```
Type : ecdsa-sha2-nistp256-cert
```

```
Data : ecdsa-sha2-nistp256-cert-v01@openssh.com AAAAKGVjZHNhLXNoYTItbmlzdHAyNTYtY2Vyd
C12MDFab3B1bnNzaC5jb20AAAAgQDMsG2AcMkoXfaK9SGTtyuJ65sd0GuR7037ikt6Yo9IAAAAAbmlzdHAyNTYAAABBA9
mwzn5O1+oV5m6Zdo3Mqmc6IjkkxrCbt+E/vhK67/B8mEaGE05JfFcJ7zHp905HsiLm0mYijS4zQCZNYRMcvNk
AAAAAAAAAAAAAAAAAAAAIAAAAFY21zY28AAAAJAAAAABWnppc2NvAAAAAAAAAAD//////////wAAAAAAAAAAAAAAAAAAAA
cAAAAHc3NoLXJzYQAAAAMBAAEAAAGBAOI0hEHzx1mQXR84w/IoKLOSfq/XI0aFqHdQ4ysQu3nTxIQeqRjtdVslQM2OZf+
iExpMl4ElZ9Y1p01BbrMynRhSywx+vtfypBIONfqi/z+jj3uea9i8tf7XF4311t1ze/SuwG9koUb+UI/MhsjL4AUefc9
u4qqY1+OVjKvZe4OfSzQglbNAQWHzhngs1pTjEeYAM5w3zv1DN4SJKPaA41/cRYLj29LJOMhD8NuAtfPkJxjU55Ja/
cISsfQdQrSTX1+2cF13vnVYL6JIqjBR9vX36fuKurlZLFx95y7D71Rab0Nh8D1kbqM8H94LLOd850XfDC/ygOjthkh
MrKipBwX9NnHOE3pwXR7RLaVXNqso04rQCJm1tiQ6ujTFGbtBhvvh+v+uTgHIIcsnJ3ZPIjrsI4KoqaIWPsoKhhBzq
JGcM1Jcs6Dqfkt16P/AUOCgo2ssUwaXIRG6sn9plipd27Pq0JvTrIcPdNce0hpr9jAWNpx9UGHeGGFXdXKwqSQh
wAAAZQAAAMcnNhLXNoYTItNTEyAAABgLLXLFmLOFZUU01hb/c10F8NEe95I865wZ1GKPGF08so5B0yeufjchHuAGkVCC
aO6IYW6jmnfROF48kgmZ03Ri83fIs0McnK9Q0/zb6t2AcWGI/cZtzm2WxQJ0C9SzsIXMGvAK+JnG2CG8Ca7Pa25hCLyhm
Rt22ysGkYCAws1buFI+1AAhnIgoUkBPUIA9kwIBTPT9dn5vezcmyfJfTgsa/X7mSm6sfvrFprz6R4Zv6AtRqi6GkWA
g47UXPmo71AjSIBgzryN1VpHm0UveWAIzu6zOLCCTiKTFqciTaIEbV0aZ5e0g72uB7T6RLhvhyhwaiZ3hqfGaiFqiTzO
omScKzM5+XTOWgW4stT5n8PqTxYXH3okHALNH29ne8JcnFm9xWgK8Ru9YxftRqDO9sb2Z5XtSZEuBr9bUCLfmeZ4ZeY
ptRxm5tXYMhAevqrRovtMcRMysZCLqYANwEh+6n0J/xgkoHFEFY2G0W0gc+a9/Ag1QoQvqyDocYa42N9NLEg==
xxxxx@vxr-slurm-146.cisco.com
```

```
Type : ed25519-cert
```

```
Data : ssh-ed25519-cert-v01@openssh.com AAAAIHNzaC1lZDI1NTE5LWN1cnQtZjAxQG9wZW5zc2guY29t
AAAAIAHRCCkOCw1xUoTS9LsmH05SeyxMolxYumXSaHygo9fFAAAAILMk1hKk4HixCE/HGwKkGkGwGLAT7ecm0fze7sQQIJ
```

```
wAAAAAAAAAAAAAAAAAAAAAAABWNpc2NvAAAAAAAAAAB11DE2AAAAAGfWWjYAAAAAAAAAAAAAAAAAAAAEAXAAAB3NzaC1yc2EAAAADAQABAAQDbrEVMz/SHOVmQaDqrXuzYWT5So6oSboVTK2krD9o93hwEU3EZY7kFqJNCNNZIMuKt36vGYEX6+kGHh9JxPrhYMUdTHQ5f1RFoOXhzIrbXPUPQ/svrTvKm6tkOqawu40dgf6dCDAIEkpdN3r8+HaK5jFjnGEMiBwXNKvHKgyW0esvuAJ7L1907z4mVlMwAOqnxIK8mMlWq6PQYnry63ekgjIPTFkJMgy9BDpmZZImB6Y3vEksxQ3hnMmMZPWlUOSrmQwRU/yk6tZKUHcQ7JFYa42UFbXboP9bnx1hd1jnP85xhZlMFDVJIGt4vBu3RGS019WqtiFVo2r9/fwoPJhVAAABFAAAAAAaxyc2Etc2hhMi01MTIAAAEAFZeQNRf3YT9K+/Zqkh17fnh+TIT2GYPktlVmyZ364EQ9igkKeOTuvqg/TNct3BBsdRMAPShxOWr+qcvkU+Amk3u5oP3TbWKvqMA91T3t/ZP3Mo+C+7ONe2zcvC9Rj2JgMn0tcVFI464vNENyqUcs2AAs/hppiCwdyXbm4kQKxkaxIukonW7E9PuBkv939L4K1VtVen4S0nTRVPX0tFXO73dIW+BhjDec9NSE/+tJY0SsuvqlL80QV73K/gHv6cJ2QaNinMSBg84Eu/SghQJO+092ocZSWQe4MiEg4Cgz/KjJhg4I4yyLbBNA176aAt7k4VTh183QZFLDMU1a4UuT5g==xxxxx@vvr-slurm-146.cisco.com
```

To view the certificate authority keys on the router, use the **show ssh server ca-keys** command.

```
Router# show ssh server ca-keys
Wed May 1 15:06:21.094 UTC
-----
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5umMS5Xc74mUxfRIMJLkawJk/BzRc1t+
/1EbD8G+eIMrwrTZ5c60mI/B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2K2jFL7qzS9+Q+vv31+
+fHvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf
f35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+WUkwsorhDz5Y2e4PUfWfV5AsgbegZmnPrXkqGb7Kmf2L1uJgcyxZT+
HZilvSY5gP7FawbkEYTOmgWJEv3f sabgupta@bg1-ads-4100
-----
Router#
```

To view the authorized keys for a user on the router, use the **server authorized-keys user** command.

```
Router# show ssh server authorized-keys user user1
Wed May 1 14:29:48.644 UTC
-----
pty ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5umMS5Xc74mUxfRIMJLkawJk/BzRc1t+/1EbD8G+eIMrwrTZ5c60mI/
B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2K2jFL7qzS9+Q+vv31+fHvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/
WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+WUkwsorhDz5Y2e4P
UfWfV5AsgbegZmnPrXkqGb7Kmf2L1uJgcyxZT+HZilvSY5gP7FawbkEYTOmgWJEv3f rsavalue
from="192.0.2.1,192.0.2.22,192.0.2.33" ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5um
MS5Xc74mUxfRIMJLkawJk/BzRc1t+/1EbD8G+eIMrwrTZ5c60mI/B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2
K2jFL7qzS9+Q+vv31+fHvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf
35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+WUkwsorhDz5Y2e4PUfWfV5AsgbegZmnPrXkqGb7Kmf2L1uJgcyxZT+HZilvSY5gP7FawbkEYTOmgWJEv3f
rsavalue
expiry-time="20241001" ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5umMS5Xc74mUxfRIMJLkawJk/BzRc1t+/
1EbD8G+eIMrwrTZ5c60mI/B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2K2jFL7qzS9+Q+vv31+fHvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/
WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+WUkwsorhDz5Y2e4PUfWfV5Asgbeg
ZmnPrXkqGb7Kmf2L1uJgcyxZT+HZilvSY5gP7FawbkEYTOmgWJEv3f rsavalue
expiry-time="20241001" ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5umMS5Xc74mUxfRIMJLkawJk/BzRc1t+/
1EbD8G+eIMrwrTZ5c60mI/B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2K2jFL7qzS9+Q+vv31+f
HvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+
WUkwsorhDz5Y2e4PUfWfV5AsgbegZmnPrXkqGb7Kmf2L1uJgcyxZT+HZilvSY5gP7FawbkEYTOmgWJEv3f rsavalue
from="abcd" ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC83fcxKGF2i5umMS5Xc74mUxfRIMJLkawJk/BzRc1t+/1EbD8G+eIMrwrTZ5c60mI/
B0Cy1hzgAKKW5KXouBPDEvyIn3BBmYlqzHzenj1RXZYmU1S0lqcB2K2jFL7qzS9+Q+vv31+fHvDRMWW5sJnsdPatdY8X1ZodNlUqwa6C/
WcQ4b2FkEp4FctmrJfXv81Mbe+KqiPA1+fjXWH7douS7FDUj2bNetf35gcxcDptbLS8oCGvJ4fQCB9kkGpKBe20a+WUkwsorhDz5Y2e4PUfWfV5AsgbegZmn
PrXkqGb7Kmf2L1uJgcyxZT+HZilvSY5gP7FawbkEYTOmgWJEv3f rsavalue
-----
```

To view the list of principals (identities) that are authorized for SSH access, use the **show ssh server authorized-principals user** command.

```
Router# show ssh server authorized-principals user user1
Wed May 1 14:37:37.933 UTC
-----
pty cisco
from="192.0.2.1,192.0.2.22,192.0.2.32" lab
expiry-time="20241001" one
```

-----

# show tech-support gnsi

To collect diagnostic information of gNSI on the router, use the **show tech-support gnsi** command in Global Configuration mode.

## show tech-support gnsi

<b>Syntax Description</b>	This command has no keywords or arguments.
---------------------------	--

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.2.11	The command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the gNSI tech-support:

```
Router# show tech-support gnsi
Mon Apr  1 06:55:51.482 UTC
++ Show tech start time: 2024-Apr-01.065551.UTC ++
Mon Apr  1 06:55:52 UTC 2024 Waiting for gathering to complete
...
Mon Apr  1 06:56:01 UTC 2024 Compressing show tech output
Show tech output available at Router#:
/harddisk:/showtech/showtech-mtb_sf2-gnsi-2024-Apr-01.065551.UTC.tgz
++ Show tech end time: 2024-Apr-01.065601.UTC ++
```

**show tech-support gnsi** command places the collected diagnostic information in a file, example **Router#:** /harddisk:/showtech/showtech-mtb\_sf2-gnsi-2024-Apr-01.065551.

# show gnsi trace pathz

To trace the configured gNSI policy on the router, use the **show gnsi trace pathz** command in Global Configuration mode.

## show gnsi trace pathz

<b>Syntax Description</b>	This command has no keywords or arguments.
---------------------------	--

<b>Command Default</b>	Enabled, by default
------------------------	---------------------

<b>Command Modes</b>	Global Configuration mode
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 24.2.11	The command was introduced.

<b>Usage Guidelines</b>	No specific guidelines impact the use of this command.
-------------------------	--

<b>Task ID</b>	<b>Task ID</b>	<b>Operation</b>
	config-services	read

This example displays the gNSI trace data on the router:

```
Router# show gnsi trace pathz all
Mon Apr 1 04:31:26.689 UTC
61 wrapping entries (21760 possible, 512 allocated, 0 filtered, 61 total)
Apr 1 04:07:09.681 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code(178) 'Trying to load policy'
'/mnt/rdsfs/ems/gnsi/pathz_policy.txt'
Apr 1 04:07:09.685 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code(173) 'Set Sandbox policy'
'1(54251382-02-18 11:34:58 +0000 UTC)'
Apr 1 04:07:09.685 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code(179) 'Set Policy from'
'/mnt/rdsfs/ems/gnsi/pathz_policy.txt'
Apr 1 04:07:09.685 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code(249) 'Pathz Policy Clearing
Counters' ' '
Apr 1 04:07:09.685 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code (79): 'Engine Initialized'
Apr 1 04:08:05.761 gnsi/pathz 0/RP0/CPU0 t11794 Pathz: Code(63) 'Pathz.Get()'
'5.38.4.111:52126'
Apr 1 04:08:05.761 gnsi/pathz_err 0/RP0/CPU0 t11794 Pathz ERROR: Code (65): 'Nil Policy'
Apr 1 04:08:05.788 gnsi/pathz 0/RP0/CPU0 t11480 Pathz: Code(63) 'Pathz.Get()'
'5.38.4.111:52126'
Apr 1 04:08:05.788 gnsi/pathz 0/RP0/CPU0 t11480 Pathz: Code(176) 'Get'
'POLICY_INSTANCE_ACTIVE 1(1711946094752098)'
Apr 1 04:08:05.791 gnsi/pathz_deny 0/RP0/CPU0 t11481 Pathz DENY: Code(235) 'Upd/Rep Denied
path' 'cafyauto@/system/config/hostname,|1,1711946094752098'
Apr 1 04:08:05.808 gnsi/pathz_deny 0/RP0/CPU0 t11383 Pathz DENY: Code(234) 'Del Denied
path' 'cafyauto@/system/config/hostname,|1,1711946094752098'
Apr 1 04:08:05.821 gnsi/pathz_deny 0/RP0/CPU0 t11480 Pathz DENY: Code(235) 'Upd/Rep Denied
path' 'cafyauto@/system/config/hostname,|1,1711946094752098'
Apr 1 04:08:07.348 gnsi/pathz_deny 0/RP0/CPU0 t11383 Pathz DENY: Code(235) 'Upd/Rep Denied
path' 'cafyauto@/lldp/config/enabled,|1,1711946094752098'
```

## show gnsi trace pathz

```
Apr 1 04:08:08.205 gnsi/pathz 0/RP0/CPU0 t11383 Pathz: Code(63) 'Pathz.Get()'
'5.38.4.111:52126'
Apr 1 04:08:08.205 gnsi/pathz_err 0/RP0/CPU0 t11383 Pathz ERROR: Code (65): 'Nil Policy'
Apr 1 04:08:08.221 gnsi/pathz 0/RP0/CPU0 t11480 Pathz: Code(63) 'Pathz.Get()'
'5.38.4.111:52126'
Apr 1 04:08:08.221 gnsi/pathz 0/RP0/CPU0 t11480 Pathz: Code(176) 'Get'
'POLICY_INSTANCE_ACTIVE 1(1711946094752098)'
Apr 1 04:08:08.238 gnsi/pathz_deny 0/RP0/CPU0 t11481 Pathz DENY: Code(235) 'Upd/Rep Denied
path' 'cafyauto@/system/config/hostname,|1,1711946094752098'
Apr 1 04:08:08.281 gnsi/pathz_deny 0/RP0/CPU0 t11480 Pathz DENY: Code(234) 'Del Denied
path' 'cafyauto@/system/config/hostname,|1,1711946094752098'
Router#
```





## NETCONF Commands

---

This module describes the commands used to use the NETCONF configuration protocol to define network operations with data models.

For detailed information about gRPC concepts, configuration tasks, and examples, see the *Use NETCONF Protocol to Define Network Operations with Data Models* in the *Cisco 8000 Series Router* module in the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

NETCONF uses an Extensible Markup Language (XML)-based data encoding for the configuration data, as well as protocol messages.

- [netconf-yang agent, on page 44](#)

# netconf-yang agent

To set the netconf-yang agent details such as netconf-yang version and netconf-yang session limits, use the **netconf-yang agent** command in the Global Configuration mode.

To remove the configured session version and limits, use the **no** form of the command.

```
netconf-yang agent { netconf1.0 | session { limit value } }
```

Syntax Description	limit <i>value</i>	Sets the maximum count for concurrent netconf-yang sessions. The range is 1–1024.
	session	Netconf -yang-agent session. You can provide the session settings, such as absolute, idle session time, number of sessions using this command.
	netconf1.0	Provides NETCONF version 1.0 support as per the RFC-4741 and RFC-4742.

**Command Default** By default, no limits are set

**Command Modes** Global Configuration mode

Command History	Release	Modification
	Release 5.3.1	This command was introduced.

**Usage Guidelines** No specific guidelines impact the use of this command.

Task ID	Task ID	Operation
	config-services	read, write

## Example

This command shows how to use the **netconf-yang agent** command:

```
Router# config
Router(config)# netconf-yang agent netconf1.0
Router(config)# netconf-yang agent session limit 10
```

```
/*limit value sets the maximum count for concurrent netconf-yang sessions. The range is
from 1 to 128.*/
Router# end
```