

System Management Command Reference for Cisco 8000 Series Routers

First Published: 2020-03-13

Last Modified: 2026-06-09

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

© 2020–2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface xv

Preface xv

Changes to This Document xv

Obtaining Documentation and Submitting a Service Request xv

CHAPTER 1

Bulk Content Downloader Commands 1

show bcdl 2

show bcdl consumers 4

show bcdl queues 6

show bcdl tables 7

show bcdl trace 9

CHAPTER 2

Boot Commands 11

reload 12

show epm trace boot 14

show reboot 16

CHAPTER 3

Call Home Commands 21

active 22

alert-group disable 23

call-home 24

call-home request 25

call-home send 27

call-home send alert-group inventory 28

call-home test 29

contact-email-addr 30

contact smart-licensing 31
contract-id 32
customer-id 33
destination address 34
destination message-size-limit 35
destination preferred-msg-format 36
destination transport-method 37
mail-server 38
phone-number 39
profile (call home) 40
rate-limit 41
sender 42
show call-home 43
show call-home alert-group 45
show call-home mail-server status 46
show call-home profile 47
show call-home statistics 48
show call-home trace 50
site-id 52
street-address 53
subscribe-to-alert-group inventory 54
subscribe-to-alert-group syslog 55

CHAPTER 4 Cisco Discovery Protocol Commands 57
cdp 58
cdp holdtime 59
cdp log adjacency changes 60
cdp timer 61
clear cdp counters 62
clear cdp table 63
show cdp 65
show cdp entry 67
show cdp interface 69
show cdp neighbors 71

show cdp traffic 74

CHAPTER 5 Configuration Management Commands 77

abort 79

alias 80

apply-template 83

clear comment 84

clear configuration commits 85

clear configuration inconsistency 87

clear configuration inconsistency replica 89

clear configuration sessions 90

commit 91

configuration commit auto-save 95

configure 97

description (interface) 99

do 100

end 101

end-group 103

end-template 104

exclude-group 105

exit 106

hostname 108

load 109

load commit changes 111

load configuration failed 112

load configuration removed 113

load rollback changes 114

man 115

more 118

pwd (config) 122

rollback configuration 123

root 126

save configuration 128

save configuration changes 129

save configuration commit changes 130
save configuration failed 131
save configuration merge 132
save configuration removed 133
save rollback changes 134
service cli commit-optimized enable 135
set default-afi 136
set default-safi 137
set default-vrf 138
show 140
show aliases 143
show apply-group 144
show running-configuration filter vrf 145
show configuration failed 147
show configuration changes 148
show configuration commit changes 150
show configuration commit list 154
show configuration (config) 156
show configuration failed incompatible 158
show configuration failed remove 159
show configuration failed rollback 161
show configuration failed startup 162
show configuration history 163
show configuration inconsistency replica 166
show configuration persistent 167
show configuration removed 169
show configuration rollback changes 170
show configuration running 172
show configuration running-config 173
show configuration sessions 175
show default-afi-safi-vrf 177
show history 178
template 180

CHAPTER 6 File System Commands 183

cd 184

cfs check 185

clear configuration ascii inconsistency 187

copy 188

delete 196

dir 197

mkdir 199

pwd 200

rmdir 201

show filesystem 202

CHAPTER 7 Manageability Commands 205

aaa map-to username 206

iteration 207

nvgen default-sanitize 209

show xml schema 210

streaming 211

throttle 212

xml agent 213

xml agent ssl 214

xml agent tty 215

CHAPTER 8 Network Configuration Protocol Commands 217

clear netconf-yang agent rate-limit 218

clear netconf-yang agent session 219

netconf-yang agent rate-limit 220

netconf-yang agent session 221

netconf-yang agent ssh 222

netconf-yang agent yfw idle-timeout 223

show gribi aft 224

show netconf-yang clients 226

show netconf-yang rate-limit 227

show netconf-yang statistics 228
ssh server capability netconf-xml 230
ssh server netconf port 231

CHAPTER 9 Object Tracking Commands 233
action 234
delay 235
interface (track) 236
line-protocol track 237
object 238
route ipv4 239
show track 240
track 241
type line-protocol state 242
type list boolean 243
type route reachability 245
type rtr 246
vrf (track) 247

CHAPTER 10 Process Memory Management Commands 249
clear context 250
dumpcore 251
exception disk-usage-limit 254
exception filepath 255
follow 259
process 266
process core 268
process mandatory 270
show context 272
show memory 274
show memory compare 277
show memory heap 280
show processes 284

CHAPTER 11 SNMP Server Commands Chapter Map 289

clear snmp counters 292

index persistence 293

notification linkupdown 294

script snmp 295

show snmp 296

show snmp context 299

show snmp context-mapping 300

show snmp engineid 301

show snmp group 302

show snmp host 304

show snmp informs details 306

show snmp interface 308

show snmp interface notification 310

show snmp interface regular-expression 312

show snmp mib 313

show snmp mib ifmib cache 316

show snmp mib ifmib general 318

show snmp mib ifmib statsd 320

show snmp request drop summary 322

show snmp request duplicates 324

show snmp request incoming-queue detail 325

show snmp request overload stats 327

show snmp request type detail 328

show snmp request type summary 329

show snmp statistics oid group 330

show snmp statistics pdu 332

show snmp statistics poll oid all 334

Show snmp statistics poll oid nms 336

show snmp statistics slow oid 337

show snmp statistics slow oid [after/before] hh:mm:ss day mday year 339

show snmp traps details 341

show snmp users 343

show snmp view 345
snmp-server trap authentication vrf disable 346
snmp-server chassis-id 347
snmp-server community 348
snmp-server community-map 350
snmp-server contact 352
snmp-server context 353
snmp-server context mapping 354
snmp-server drop report acl 356
snmp-server drop unknown-user 357
snmp-server engineid local 358
snmp-server engineid remote 359
snmp-server entityindex persist 360
snmp-server group 361
snmp-server host 364
snmp-server ifindex persist 368
snmp-server ifmib internal cache max-duration 369
snmp-server ifmib stats cache 370
snmp-server inform 371
snmp-server interface 372
snmp-server interface subset 374
snmp-server ipv4 dscp 376
snmp-server ipv4 precedence 377
snmp-server location 379
snmp-server mibs cbqosmib persist 380
snmp-server mibs cbqosmib cache 381
snmp-server mibs eventmib congestion-control 382
snmp-server mibs eventmib packet-loss 384
snmp-server mibs sensormib cache 386
snmp-server mibs subscriber threshold access-if 387
snmp-server mibs subscriber threshold 388
snmp-server notification-log-mib 390
snmp-server packetsize 392
snmp-server queue-length 393

snmp-server script 394
snmp-server target list 395
snmp-server trap throttle-time 396
snmp-server trap-timeout 397
snmp-server traps 398
snmp-server traps bgp 404
snmp-server traps frequency synchronization 406
snmp-server traps mpls l3vpn 407
snmp-server traps ospf errors 408
snmp-server traps ospf lsa 410
snmp-server traps ospf retransmit 411
snmp-server traps ospf state-change 412
snmp-server traps ospfv3 errors 414
snmp-server traps ospfv3 state-change 415
snmp-server traps pim interface-state-change 417
snmp-server traps pim invalid-message-received 418
snmp-server traps pim neighbor-change 419
snmp-server traps pim rp-mapping-change 420
snmp-server traps rsvp 421
snmp-server traps selective-vrf-download role-change 422
snmp-server traps snmp 423
snmp-server traps subscriber session-aggregation 425
snmp-server traps syslog 426
snmp-server user 427
snmp-server view 430
snmp-server vrf 432

CHAPTER 12 **Software Package Management Commands** 435

install commit 436
show install active 437
show install available 440
show install cached 441
show install committed 443
show install fixes 445

show install history 447
show install log 449
show install request 450
show install rollback 452

CHAPTER 13 Terminal Services Commands 453
absolute-timeout 454
access-class 455
autocommand 457
banner exec 459
banner login 461
banner motd 463
clear line 465
clear line vty 466
disconnect-character 467
escape-character 468
exec-timeout 469
length 470
line 471
send 472
session-timeout 473
show line 474
show terminal 477
show cli submode-exit 479
show users 480
terminal length 481
terminal width 482
timestamp disable 483
transport input 484
transport output 485
transport preferred 486
vty-pool 487
cli interactive syntax check 489
cli whitespace completion 490

CHAPTER 14 Utility Commands 491

utility cut 492
utility date 496
utility date set 498
utility fgrep 499
utility find 501
utility less 503
utility mv 505
utility sort 506
utility tail 509
utility uniq 511
utility wc 513
utility bc 515
utility df 519
utility du 523
utility egrep 526
utility head 529
utility which 531
utility xargs 533

CHAPTER 15 Zero Touch Provisioning Commands 535

ztp breakout 536
ztp bootz 537
ztp bootz-server 538
ztp clean 539
ztp initiate 540
ztp terminate 542



Preface

- Preface, on page xv
- Changes to This Document, on page xv
- Obtaining Documentation and Submitting a Service Request, on page xv

Preface

This guide consists of information about the system management commands in Cisco IOS XR Software.

The *System Management Command Reference for Cisco 8000 Series Routers* preface contains these sections:

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
September 2025	Republished for Release 25.3.1
June 2025	Republished for Release 25.2.1
March 2025	Republished for Release 25.1.1
December 2019	Initial release of this document.

Obtaining Documentation and Submitting a Service Request



Bulk Content Downloader Commands

- show bcdl, on page 2
- show bcdl consumers, on page 4
- show bcdl queues, on page 6
- show bcdl tables, on page 7
- show bcdl trace, on page 9

show bcdl

To display Bulk Content Downloader (BCDL) information, use the **show bcdl** command in EXEC mode.

show bcdl [*group_name*]

Syntax Description

group_name	(Optional) Displays information for a specific BCDL group.

Command Default

No default behavior or values

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read

The following example shows sample output from the **show bcdl** command:

```
RP/0/RSP0/CPU0:router# show bcdl ipv4_rib

Sun May 31 06:56:12.093 PST
grp ipv4_rib, gid 2040, sg cnt 1, agent jid 124, node 0/RSP0/CPU0, pulse 105,
new mbr 0
sg  lwg fd csmr hdlr-act dnld-act susp wait-lck seq pulse-tot pulse-out
 0 2043 11    4       no       no   no        0 113       103         0
```

Table 2: show bcdl Field Descriptions

Field	Description
group	Type of download and the Group Services Protocol (GSP) group name.
gid	Heavyweight group (HWG) in the GSP. This is the group that a consumer initially joins. It is used by the BCDL agent to send control updates.
sg count	Number of subgroups for this particular download type.
agent jid	Job identifier of the BCDL agent. The JID is numerical identifier for a particular process and remains the same across process restarts.
node	Node, expressed in the *rackslotmodule* notation, in which the agent is running.

Field	Description
pulse	Pulse code used by the producer to pulse the BCDL agent.
new mbr	Number of new consumers that have not yet been assigned a subgroup.
sg	Subgroups number.
lwg	Lightweight group in GSP. This is a type of child group of the HWG. The BCDL agent tells the consumers to join this group to receive data.
fd	The connection handle between the producer and the BCDL agent.
csmr	Number of consumers.
hdlr-act	Specifies if there is a download in progress.
dnld-act	Indicates whether the convergence flag has been sent or not.
susp	Indicates whether the download is suspended due to the queue filling up.
wait-lck	If nonzero, some thread is waiting for other thread to take control of this subgroup.
seq	Sequence number of the last message sent on this subgroup.
pulse-tot	Total number of pulses sent by the producer to the BCDL agent.
pulse-out	Total number of outstanding pulses that have not yet been processed by the BCDL agent.

show bcdl consumers

To display Bulk Content Downloader (BCDL) consumer information, use the **show bcdl consumers** command in

EXEC

mode

show bcdl consumers [*group_name*]

Syntax Description

group_name	(Optional) Displays information for a specific BCDL group.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read

The following example shows sample output using the **show bcdl consumers** command:

```
RP/0/RSP0/CPU0:router# show bcdl consumers ipv4_rib

Sun May 31 06:17:38.209 PST
group ipv4_rib, gsp gid 2040, 4 consumers, agent jid 124, node 0/RSP0/CPU0
(expected 4 consumers to reply, received 4 replies)
      pid       node asg csg  lwg sus  messages       bytes  errs name
   323727 0/RSP0/CPU0   0   0 2043   N       113       54196     0 fib_mgr
   110686   0/6/CPU0   0   0 2043   N       111       54140     0 fib_mgr
   110686   0/4/CPU0   0   0 2043   N       112       54168     0 fib_mgr
   110686   0/1/CPU0   0   0 2043   N       111       54140     0 fib_mgr
```

This table describes the significant fields shown in the display that are not described in Table 2: show bcdl Field Descriptions, on page 2.

Table 3: show bcdl consumers Field Descriptions

Field	Description
PID	Process identifier.

Field	Description
node	Consumer node, expressed in the *rack/slot* notation.
asg	Subgroup to which the BCDL agent thinks this consumer belongs.
csg	Subgroup to which the consumer thinks it belongs.
messages	Number of messages processed by this particular consumer.
bytes	Bytes processed by this particular consumer.
errors	Errors encountered by the consumer. This field indicates the number of times the connection was reset.
name	Name of the consumer process.

show bcdl queues

To display the Bulk Content Downloader (BCDL) queue information, use the **show bcdl queues** command in

EXEC

mode.

show bcdl queues [*group_name*]

Syntax Description

group_name	(Optional) Displays information for a specific BCDL group.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read

The following example shows sample output from the **show bcdl queues** command:

```
RP/0/RSP0/CPU0:router# show bcdl queues ipv4_rib

Sun May 31 07:13:28.665 PST
group ipv4_rib, gsp gid 2040, 4 consumers, agent jid 124, node 0/RSP0/CPU0
(expected 4 consumers to reply, received 4 replies)
      pid       node asg csg  lwg sus msgs_in_q  bytes_in_q  errs name
   323727 0/RSP0/CPU0   0   0 2043   N         0           0     0 fib_mgr
   110686   0/6/CPU0   0   0 2043   N         0           0     0 fib_mgr
   110686   0/1/CPU0   0   0 2043   N         0           0     0 fib_mgr
   110686   0/4/CPU0   0   0 2043   N         0           0     0 fib_mgr
```

Table 2: show bcdl Field Descriptions, on page 2 and Table 3: show bcdl consumers Field Descriptions, on page 4 describe the significant fields shown in the display.

show bcdl tables

To display Bulk Content Downloader (BCDL) table information, use the **show bcdl tables** command in

EXEC

mode.

show bcdl tables [*group_name*]

Syntax Description

group_name	Displays information for a specific BCDL group.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read

The following example shows sample output using the **show bcdl tables** command:

```
RP/0/RSP0/CPU0:router# show bcdl tables ipv4_rib

Sun May 31 07:19:29.878 PST
grp ipv4_rib, gid 2040, sg cnt 1, agent jid 124, node 0/RSP0/CPU0, pulse 105, ne
w mbr 0
  sg  lwg fd csmr hdlr-act dnld-act susp wait-lck   seq pulse-tot pulse-out
   0 2043 11    4       no       no   no        0   113       103         0
sgs: 1, table_cnt: 1, table_mid_cnt: 4, buf size: 100
Showing table info for 1 subgroups
 sg 0: has 1 tables (messages: 0, bytes: 0)
  table 0xe0000000: 4 members, dnld act: 0, messages: 113, bytes: 54196
   cnsmr 0: pid 323727 on node 0/RSP0/CPU0
   cnsmr 1: pid 110686 on node 0/6/CPU0
   cnsmr 2: pid 110686 on node 0/1/CPU0
   cnsmr 3: pid 110686 on node 0/4/CPU0
```

The significant fields shown in the display that are not described in show bcdl consumers, on page 4 or show bcdl queues, on page 6 are described in this table.

Table 4: show bcdl tables Field Descriptions

Field	Description
sgs	Number of subgroups.
table_cnt	Number of tables in this subgroup.
sg	Specific subgroup for which information is provided.
has 1 tables	Number of tables in this subgroup.
messages	Messages sent that are not associated with a particular table ID.
bytes	Bytes sent that are not associated with a particular table ID.
table	Specific table ID for which information is provided.
members	Number of consumers associated with this table.
dnld act	Indicates whether or not the convergence flag has been sent.
messages	Number of messages sent for a particular table.
bytes	Number of bytes sent for a particular table.
cnsmr 0: pid 419725 on node 0/RP0/CPU0	Process ID and node information for each consumer in the specified table.

show bcdl trace

To display Bulk Content Downloader (BCDL) trace information, use the **show bcdl trace** command in

EXEC

mode.

show bcdl trace [*group_name*] [**event**] [**timing**] [**grpsnd**] [**wrapping** | **unique**] [**hexdump**] [**last** *n*] [**reverse**] [**stats**] [**tailf**] [**verbose**] [**file** *filename* **original location** *node-id* | **location** {*node-id* | **all**}]

Syntax Description

group_name	(Optional) Displays information for a specific BCDL group.
event	(Optional) Displays event trace entries.
timing	(Optional) Displays timing trace entries.
grpsnd	(Optional) Displays group send trace entries.
wrapping	(Optional) Displays wrapping entries.
unique	(Optional) Displays unique entries only, along with the count of the number of times this entry appears.
hexdump	(Optional) Displays traces in hexadecimal format.
last *n*	(Optional) Displays the last *n* number of traces only.
reverse	(Optional) Displays the most recent traces first.
stats	(Optional) Displays execution path statistics.
tailf	(Optional) Displays new traces as they are added.
verbose	(Optional) Displays additional internal debugging information.
file *filename* **original location** *node-id*	(Optional) Specifies a filename and original location of the file to display.
location {*node-id* | **all**}	Specifies the RP node for which to display the execution path monitoring information. The *node-id* argument is expressed in the *rack/slot* notation. The **all** keyword specifies all RP nodes.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read

The following example shows sample output using the **show bcdl trace** command:

```
RP/0/RSP0/CPU0:router# show bcdl trace ipv4_rib location 0/1/cpu0

Sun May 31 08:21:07.933 PST
143 wrapping entries (4096 possible, 0 filtered, 143 total)
May 21 15:14:55.790 bcdl/c/ipv4_rib 0/1/CPU0 t4  LE
  bcdl_join_internal: timer_create ret 0, id is 9
May 21 15:14:56.890 bcdl/c/ipv4_rib 0/1/CPU0 t7  LE
  bcdl_join_internal: group_lookup bcdl_ipv4_rib
  returned gid 2040
May 21 15:14:56.966 bcdl/c/ipv4_rib 0/1/CPU0 t7  LE
  join hwg 2040 returns 0
May 21 15:14:56.978 bcdl/c/ipv4_rib 0/1/CPU0 t7  LE
  bcdl_join_internal: joined group bcdl_ipv4_rib,
  member count 5
May 21 15:14:58.969 bcdl/c/ipv4_rib 0/1/CPU0 t7  LE
  rcv gsp mtype 3: connection init sg 1 cur_seq 0
  lwg_gid 2056 table tag 0x00000000 resend state yes
May 21 15:14:58.969 bcdl/c/ipv4_rib 0/1/CPU0 t7  LE
  pc ring high water 0 -> 1, 0 bytes
May 21 15:14:58.969 bcdl/c/ipv4_rib 0/1/CPU0 t4  LE
  c_h deliver msg_id 7 connection init, table event
  0 table tag 0x00000000
May 21 15:14:58.969 bcdl/c/ipv4_rib 0/1/CPU0 t4  LE
  conn init, seq 64206 -> 0, sg 65534 -> 1, gid 2040,
  lwg gid -1 -> 2056
 ...
```



Boot Commands

- reload, on page 12
- show epm trace boot, on page 14
- show reboot, on page 16

reload

To reloads the route processor (RP), use the **reload** command in XR EXEC mode.

reload

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Releases	Modifications
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **reload** command to cause the RP to reload the Cisco IOS XR software according to the configuration register setting (for example, 0x0 to enter ROMMON mode and 0x2 to reload the RP to EXEC mode). If a standby RP is in the ready redundancy state, the **reload** command also causes the router to fail over to the standby RP. Use the **show redundancy** command in EXEC mode to display the status of the standby RP.

When the **reload** command is used and a switchover occurs, the running (active) software configuration is automatically maintained during switchover.

Caution If a standby RP is not installed or is not in the ready state, then the router experiences a loss of service while the active RP is reloading Cisco IOS XR software. To view the status of the standby RP, issue the **show redundancy** command in EXEC mode.

If you use the **reload** command and there is no available standby node, you are prompted to continue with the reload:

```
Router# reload

Standby card not present or not Ready for failover. Proceed?[confirm]y
```

Task ID

Task ID	Operations
root-lr	execute

The following example shows how to reload the active RP. If a standby RP is in the ready state, then the router fails over to the standby RP. If the standby RP is not installed or is not in the ready state, then the router enters ROMMON mode and routing operations stop.

```
Router# reload

  Updating Commit Database.  Please wait...[OK]
```

```
Proceed with reload? [confirm] y

PCI0 device[7]: Vendor ID 0x10ee
PCI0 device[7]: Device ID 0x300e
PCI1 device[7]: Device ID 0x1100
PCI1 device[7]: Vendor ID 0x1013
PCI1 device[8]: Device ID 0x649
PCI1 device[8]: Vendor ID 0x1095
PCI1 device[9]: Device ID 0x5618
PCI1 device[9]: Vendor ID 0x14e4
PCI1 device[10]: Device ID 0x5618
PCI1 device[10]: Vendor ID 0x14e4
System Bootstrap, Version 1.15(20040120:002852) ,
Copyright (c) 1994-2004 by cisco Systems, Inc.
Board type is 0x100000 (1048576)
Enabling watchdog
Broadcom 5618 #0 Found on PCI
Broadcom 5618 #1 Found on PCI
No. of BCM 56xx switches found 2 .
BCM Switch #0 initialisation complete.
BCM Switch #1 initialisation complete
G4(7450-SMP-GT64260_A) platform with 2048 Mb of main memory

rommon B1 >
```

show epm trace boot

To display execution path monitoring traces, use the **show epm trace boot** command in administration EXEC mode.

show epm trace boot [**hexdump**] [**last** *n*] [**reverse**] [**stats**] [**tailf**] [**unique**][**verbose**] [**wrapping**][**file** *filename* **original**] [**location** {*node-id* | **all**}]

Syntax Description

hexdump	(Optional) Displays traces in hexadecimal format.
last *n*	(Optional) Displays the last *n* number of traces only.
reverse	(Optional) Displays the most recent traces first.
stats	(Optional) Displays execution path statistics.
tailf	(Optional) Displays new traces as they are added.
unique	(Optional) Displays unique entries only, along with the count of the number of times this entry appears.
verbose	(Optional) Displays additional internal debugging information.
wrapping	(Optional) Displays wrapping entries.
file *filename* **original**	(Optional) Specifies the filename of the file to display. You can specify up to four trace files.
location {*node-id* | **all**}	(Optional) Specifies the node of the . The *node-id* argument is entered in the *rack/slot* notation. You can specify up to four nodes. The **all** keyword specifies all nodes.

Command Default

None

Command Modes

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show epm trace boot** command provides a simple way of tracking and time-stamping critical events to clearly understand their temporal relationship to one another and the amount of time spent performing critical operations.

Task ID

Task ID	Operations
basic services	read

The following example shows sample output from the **show epm trace boot** command:

```
RP/0/RP0/CPU0:router(admin)# show epm trace boot

Mon Jun  1 03:16:36.946 PST
22 wrapping entries (1024 possible, 0 filtered, 22 total)
Oct  8 07:54:49.610 epm/boot 0/RP0/CPU0 t1  @ 00:00:06 - [init] process-start
Oct  8 07:55:25.710 epm/boot 0/RP0/CPU0 t1  @ 00:00:42 - [insthelper] process-start
Oct  8 07:57:08.992 epm/boot 0/RP0/CPU0 t1  @ 00:02:25 - [sysmgr] process-start
Oct  8 07:57:09.785 epm/boot 0/RP0/CPU0 t7  @ 00:02:26 - [sysmgr] start-level: start
Oct  8 07:57:10.722 epm/boot 0/RP0/CPU0 t1  @ 00:02:27 - [sw_dwnld_svr] process-start
Oct  8 07:57:12.482 epm/boot 0/RP0/CPU0 t11 @ 00:02:29 - [sysmgr] start-level: admin
Oct  8 07:57:13.385 epm/boot 0/RP0/CPU0 t1  @ 00:02:30 - [instdir] process-start
Oct  8 07:57:19.638 epm/boot 0/RP0/CPU0 t1  @ 00:02:36 - [instdir_lr] process-start
Oct  8 07:58:07.045 epm/boot 0/RP0/CPU0 t9  @ 00:03:23 - [sysmgr] admin-plane-up
Oct  8 07:58:52.057 epm/boot 0/RP0/CPU0 t4  @ 00:04:08 - [cfgmgr-rp] admin-config-start
Oct  8 07:58:59.973 epm/boot 0/RP0/CPU0 t4  @ 00:04:16 - [cfgmgr-rp] admin-config-done
Oct  8 07:59:00.079 epm/boot 0/RP0/CPU0 t9  @ 00:04:16 - [sysmgr] start-level: infra
Oct  8 07:59:00.615 epm/boot 0/RP0/CPU0 t1  @ 00:04:17 - [devc-conaux] exec-available
Oct  8 07:59:02.288 epm/boot 0/RP0/CPU0 t4  @ 00:04:18 - [cfgmgr-rp] admin-plane-mount-done
Oct  8 07:59:08.157 epm/boot 0/RP0/CPU0 t6  @ 00:04:24 - [instdir] ready-for-requests
Oct  8 07:59:15.999 epm/boot 0/RP0/CPU0 t6  @ 00:04:32 - [sysmgr] start-level: active
Oct  8 07:59:32.300 epm/boot 0/RP0/CPU0 t13 @ 00:04:48 - [sysmgr] start-level: final
Oct  8 07:59:38.143 epm/boot 0/RP0/CPU0 t9  @ 00:04:54 - [sysmgr] lr-plane-up
Oct  8 07:59:38.189 epm/boot 0/RP0/CPU0 t4  @ 00:04:54 - [cfgmgr-rp] lr-config-start
Oct  8 07:59:49.898 epm/boot 0/RP0/CPU0 t4  @ 00:05:06 - [cfgmgr-rp] lr-config-done
Oct  8 07:59:50.259 epm/boot 0/RP0/CPU0 t4  @ 00:05:06 - [cfgmgr-rp]
bulk-interface-config-start
Oct  8 07:59:50.351 epm/boot 0/RP0/CPU0 t7  @ 00:05:06 - [cfgmgr-rp] node-config-done
```

In this sample output, the time stamp following the @ sign is the elapsed time in the format hh:mm:ss since the execution phase started (for example, since node start, in the case of a boot).

show reboot

To display reboot information for a node, use the **show reboot** command in

EXEC or administration EXEC

System Admin EXEC

mode.

show reboot {**history** [**reverse**] | {**first** | **last**} {**crashinfo** | **syslog** | **trace**} | **graceful**} **location** *node-id*

Syntax Description

first	(Optional) Displays information about the first ungraceful reboot.
last	(Optional) Displays information about the last ungraceful reboot.
crashinfo	Displays crash information for an ungraceful reboot.
syslog	Displays the syslogs related to an ungraceful reboot.
trace	Displays trace information for an ungraceful reboot.
graceful	Displays information about the last graceful reboot.
history	Displays the reboot history of a specific node.
reverse	(Optional) Displays the reboot history information in reverse chronological order. **Note** Starting from Cisco IOS XR Release 24.3.1, the **reverse** keyword is deprecated and will not be supported in future releases.
location *node-id*	Specifies which node to reload. The *node-id* argument is expressed in the *rack/slot* notation.

Command Default

None

Command Modes

EXEC

Administration EXEC

System Admin EXEC

Command History

Release	Modification
Release 24.3.1	The **reverse** keyword is deprecated and will not be supported in future releases. Hence the **show reboot history reverse location** command is also not supported.
Release 7.0.12	This command was introduced.

Usage Guidelines

- The **show reboot history** command displays all reboot causes stored for previous node resets.
- The **show reboot history** command output does not include information about the system reload event triggered by power cycle.
- Crash information (**crashinfo**), syslog, and kernel dumper ltrace (**trace**) is displayed for the first or last reboot if it is an ungraceful reboot.

Task ID

Task ID	Operations
system	read

This example shows the history of reloaded nodes.

```
RP/0/RP0/CPU0:ios#show reboot history location 0/RP0/CPU0
Wed Apr 17 16:55:20.748 PDT
--------------------------------------------------------------------------------
No DATE TIME (PDT) Cause Code Cause String
--------------------------------------------------------------------------------
1 Apr 12 2024 12:01:04 0x00000024 REBOOT_CAUSE_UPGRADE
2 Mar 29 2024 00:08:40 0x00000024 REBOOT_CAUSE_UPGRADE
3 Mar 05 2024 03:58:00 0x00000025 REBOOT_CAUSE_ADMIN
4 Feb 27 2024 22:58:08 0x00000024 REBOOT_CAUSE_UPGRADE
5 Feb 27 2024 00:02:34 0x00000024 REBOOT_CAUSE_UPGRADE
6 Feb 15 2024 11:06:58 0x00000024 REBOOT_CAUSE_UPGRADE
7 Feb 06 2024 18:15:40 0x00000025 REBOOT_CAUSE_ADMIN
8 Feb 02 2024 16:57:24 0x00000024 REBOOT_CAUSE_UPGRADE
--------------------------------------------------------------------------------
```

This example displays the crash information for the first reboot.

```
RP/0/RP0/CPU0:router# show reboot first crashinfo location 0/RP0/CPU0

Crashinfo Timestamp: Thu Jul 19 20:32:57 2007

20070719 20:32:57

Crash Reason: Cause code 0x21000010 Cause: Missed deadline,
  client: sc-reddrv-main, timeout: 5 Process: wd-critical-mon
Traceback: fc1941a0 fc194290 48200738 482013cc 48201c04 fc1d4fb0 Timezone UTC0

Exception at 0xfc1944c8 signal 5 c=1 f=3

Active process(s):
pkg/bin/wd-critical-mon Thread ID 1 on cpu 0
pkg/bin/l3test Thread ID 0 on cpu 1

       REGISTER INFO
      r0        r1        r2        r3
R0   01000000  4817e8c0  4820e208  000000de
       r4        r5        r6        r7
R4   fc1b4856  7fffffff  4817e738  fc1b4856
       r8        r9       r10       r11
R8   00000000  602cf522  00000000  00000000
      r12       r13       r14       r15
R12  602cf51c  4820e1a0  00000000  00000000
      r16       r17       r18       r19
R16  00000000  00000000  00000000  00000000
      r20       r21       r22       r23
```

```
R20  00000000  00000000  48200000  48200000
       r24       r25       r26       r27
R24  48200000  48200000  48200000  48200000
       r28       r29       r30       r31
R28  00000028  00000001  21000010  6029b000
         cnt        lr       msr        pc
R32  00000000  fc194290  0002d932  fc1944c8
         cnd       xer
R36  44000094  20000006

                SUPERVISOR REGISTERS

               Memory Management Registers

               Instruction BAT Registers
                Index #                Value
               IBAT0U #              0x1ffe
               IBAT0L #                0x12
               IBAT1U #                   0
               IBAT1L #                   0
               IBAT2U #          0x30000ffe
               IBAT2L #          0xf0000032
               IBAT3U #          0xfffc0003
               IBAT3L #             0x40011

               Data BAT Registers
                Index #                Value
               DBAT0U #              0x1ffe
               DBAT0L #                0x12
               DBAT1U #                   0
               DBAT1L #          0x10000012
               DBAT2U #          0x30000ffe
               DBAT2L #          0xf000006a
               DBAT3U #          0xfffc0003
               DBAT3L #             0x40011

               Segment Registers
                Index #              SR-Value
                    0 #                    0
                    1 #                    0
                    2 #                    0
                    3 #                    0
                    4 #                    0
                    5 #                    0
                    6 #                    0
                    7 #                    0
                    8 #                    0
                    9 #                    0
                   10 #                    0
                   11 #                    0
                   12 #                    0
                   13 #                    0
                   14 #                    0
                   15 #                    0

                    Exception Handling Registers
          Data Addr Reg #                DSISR
            0x602cf440 #          0x42000000
        SPRG0 #       SPRG1 #       SPRG2 #       SPRG3
        0x1 # 0x21000010 # 0x6029b000 #        0
     SaveNRestore SRR0 #    SaveNRestore SRR1
          0xfc1944c4 #             0x2d932

  Miscellaneous Registers
```

```
    Processor Id Reg #                  0
                HID0 #          0x8410c0bc
                HID1 #          0x9001ac80

               MSSCR0 #             0x88000
               MSSSR0 #                   0

 STACK TRACE
#0 0xfc194290
#1 0x48200738
#2 0x482013cc
#3 0x48201c04
#4 0xfc1d4fb0
```



Call Home Commands

- active, on page 22
- alert-group disable, on page 23
- call-home, on page 24
- call-home request, on page 25
- call-home send, on page 27
- call-home send alert-group inventory, on page 28
- call-home test, on page 29
- contact-email-addr, on page 30
- contact smart-licensing, on page 31
- contract-id, on page 32
- customer-id, on page 33
- destination address, on page 34
- destination message-size-limit, on page 35
- destination preferred-msg-format, on page 36
- destination transport-method, on page 37
- mail-server, on page 38
- phone-number, on page 39
- profile (call home), on page 40
- rate-limit, on page 41
- sender, on page 42
- show call-home, on page 43
- show call-home alert-group, on page 45
- show call-home mail-server status, on page 46
- show call-home profile, on page 47
- show call-home statistics, on page 48
- show call-home trace, on page 50
- site-id, on page 52
- street-address, on page 53
- subscribe-to-alert-group inventory, on page 54
- subscribe-to-alert-group syslog, on page 55

active

To enable a Call Home profile, use the **active** command in call home profile configuration mode. To disable a profile, use the **no** form of this command.

active
no active

Syntax Description

This command has no keywords or arguments.

Command Default

A profile is disabled by default.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must enable a profile using the **active** command so that call home messages can be triggered.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to activate a profile:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# active
```

alert-group disable

To disable an individual Call Home alert-group, use the **alert group disable** command in call home configuration mode. To enable an individual Call Home alert-group, use the **no** form of this command.

alert-group *alert-group-name* **disable**
no alert-group *alert-group-name* **disable**

Syntax Description

alert-group-name	A keyword that identifies an alert group. Valid values are: • syslog • environment • inventory

Command Default

Alert groups are enabled by default.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

By default, alert groups are enabled. Use the **alert-group disable** command to disable alert groups.

Task ID

Task ID	Operation
call-home	read, write

Example

The following example shows how to disable the inventory alert group:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# alert-group inventory disable
```

call-home

To enter call home configuration mode to configure Call Home, use the **call-home** command in global configuration mode. To remove all Call Home settings and set the technical assistance center (TAC) profile as the default, use the **no** form of this command.

call-home
no call-home

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to enter call home configuration mode:

```
Router(config)# call-home
Router(config-call-home)#
```

call-home request

To send a customer request to Cisco, use the **call-home request** command in EXEC mode.

call-home request {**bugs-list** | **command-reference** | **config-sanity** | **output-analysis** "*show-command*" | **product-advisory**} {**ccoid** *ccoid* | **profile** *profile-name*}

Syntax Description

bugs-list	Sends output of the following commands: • **show running-config sanitized** • **show version** • **show diag**
command-reference	Sends output of the following commands: • **show running-config sanitized** • **show version** • **show diag**
config-sanity	Sends output of the following commands: • **show running-config sanitized** • **show version**
output-analysis *show-command*	Sends output from the specified show command. The *show-command* argument should be enclosed in quotes ("").
product-advisory	Sends output of all commands included in the enventory message in addition to the output from the **show running-config sanitized** command.
ccoid *ccoid*	Specifies the Smart Call Home user registered ID.
profile *profile-name*	Specifies the profile to which to send the message.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The message uses the specified profile or the CiscoTAC-1 profile if no profile name is specified to send out the request the the Cisco backend. This ensures that users who use a transport gateway can use a different

profile than the CiscoTAC-1 profile to send the request to their email server first before forwarding to the Cisco backend. CiscoTAC-1 or any profile specified needs to be enabled before you can send out the request.

If the CCO ID is not specified, the contact email address of the device is used.

Each message sent includes the CLI command ouput specified for each subcommand. After the message is sent, a syslog message is displayed indicating whether the request was sent successfully or not.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to send a message with output from the **show version** command to engineer@cisco.com:

```
RP/0/RP0/CPU0:router(config)# call-home request config-sanity ccoid
xyz
RP/0/RP0/CPU0:router(config)# call-home request bugs-list
RP/0/RP0/CPU0:router(config)# call-home request output-analysis "show log"
profile TG
RP/0/RP0/CPU0:router(config)#  call-home request output-analysis "show
running-config"
```

call-home send

To send the output from a specific command as a Call Home message, use the **call-home send** command in EXEC mode.

call-home send "*cli-command*" {**email** *email-address* | **tac-service-request** *service-number*}

Syntax Description

cli-command	Any CLI command that can be run at the prompt.
email *email-address*	Specifies the email address to which to send the call home message.
tac-service-request *service-number*	Specifies the Technical Assistance Center (TAC) service request number.

Command Default

If no email address is specified, attach@cisco.com is used.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The specified CLI command is generally a show command with output that is required by the TAC to analyze a specific issue. The CLI command must be enclosed in quotes. The CLI command output is sent to the specified email address in long text format with the service request number in the subject of the email. If no email address is specified, the TAC service request number must be specified. By default, messages are sent to attach@cisco.com. If no TAC service request number is specified, TAC email is rejected.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to send a message with output from the **show version** command to engineer@cisco.com:

```
RP/0/RP0/CPU0:router(config)# call-home send "show version" engineer@cisco.com
```

call-home send alert-group inventory

To send an inventory Call Home message to all subscribed profiles or the specified profile, use the **call-home send alert-group inventory** command in EXEC mode.

call-home send alert-group inventory [**profile** *profile-name*]

Syntax Description

profile *profile-name*	Specifies the profile to which to send the inventory Call Home message.

Command Default

If no profile is specified, the message is sent to all subscribed profiles.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The profile specified by the *profile-name* argument does not need to be subscribed to the inventory alert-group.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to send an inventory message to the myprofile profile:

```
RP/0/RP0/CPU0:router(config)# call-home send alert-group inventory profile myprofile
```

call-home test

To send a test Call Home message to the specified profile, use the **call-home test** command in EXEC mode.

call-home test [*test-message-text*] **profile** *profile-name*

Syntax Description

test-message-text	Text to be sent in the test message. If the message text is not specified, a default message is sent.
profile *profile-name*	Specifies the profile to which to send the test call home message.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to send a test Call Home message:

```
RP/0/RP0/CPU0:router(config)# call-home test "this is a test message" profile myprofile
```

contact-email-addr

To specify a contact email address for the system, use the **contact-email-addr** command in call home configuration mode. To disable the contact email address, use the **no** form of this command.

contact-email-addr *email-addr*
no contact-email-addr *email-addr*

Syntax Description

email-addr	The email address of the system contact.

Command Default

No contact email address is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The contact email address is a mandatory user-configurable field that must be configured before Call Home messages are triggered.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the contact email address:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# contact-email-addr contact@cisco.com
```

contact smart-licensing

To send notification emails to your Smart Licensing account, use **contact smart-licensing** command in the **call-home** configuration mode. To disable the contact, use the **no** form of this command.

contact smart-licensing
no contact smart-licensing

Syntax Description

This command has no keywords or arguments.

Command Default

contact smart-licensing is not configured.

Command Modes

call-home configuration mode.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The Smart Call Home back-end service uses the email address configured in your Smart Licensing account as the destination address for the notification mails. Enabling Smart Licensing also enables call-home which changes several mandatory configurations in call-home. **contact smart-licensing** is one of the configurations that is automatically added when you enable smart-licensing.

Note The **contact-email-addr** command and the **contact smart-licensing** command cannot be configured simultaneously.

Task ID

Task ID	Operation
call-home	Read, Write

Example

This example shows how to configure the contact email address:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# call-home
RP/0/RP0/CPU0:router(config-call-home)# contact smart-licensing
```

contract-id

To specify a contract identifier for the system, use the **contract-id** command in call home configuration mode. To disable the contract identifier, use the **no** form of this command.

contract-id *contract-id-string*
no contract-id *contract-id-string*

Syntax Description

contract-id-string	A string that identifies a service contract.

Command Default

No contract ID is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The contract ID is an optional user-configurable field that can be used for contract information or any other identification information for the support service.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the contract ID:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# contract-id contract
```

customer-id

To specify a customer identifier for the system, use the **customer-id** command in call home configuration mode. To disable the customer identifier, use the **no** form of this command.

customer-id *contract-id-string*
no customer-id *contract-id-string*

Syntax Description

customer-id-string	A string that identifies the customer.

Command Default

No customer ID is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The customer ID is an optional user-configurable field that can be used for contract information or any other identification information of a support service.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the customer ID:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# customer-id cisco
```

destination address

To specify an email address to which Call Home messages are sent, use the **destination address** command in call home profile configuration mode. To disable the contract identifier, use the **no** form of this command.

destination address *email-address*
no destination address *email-address*

Syntax Description

email-address	Email address to which short-text and long-text Call Home messages and XML-based Call Home messages are to be sent.

Command Default

No destination email address is defined.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must define a destination email address to send out Call Home messages.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the destination email address:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# destination address user@cisco.com
```

destination message-size-limit

To specify the message size limit for call home messages for a specific profile, use the **destination message-size-limit** command in Call Home profile configuration mode. To disable the message size limit, use the **no** form of this command.

destination message-size-limit *maximum-size*
no destination message-size-limit *maximum-size*

Syntax Description

maximum-size	Maximum message size in bytes.

Command Default

The default maximum message size is 3 Mbytes.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The destination maximum message size can be optionally set to limit the size of Call Home messages.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the contract ID:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# destination maximum-message-size 2000
```

destination preferred-msg-format

To specify the message format for a profile, use the **destination preferred-msg-format** command in call home profile configuration mode. To revert to the default message format, use the **no** form of this command.

destination preferred-msg-format {**long-text** | **short-text** | **xml**}
no destination preferred-msg-format {**long-text** | **short-text** | **xml**}

Syntax Description

long-text	Specifies to send clear text long Call Home messages, that contain formatting to enable easy readability.
short-text	Specifies to send clear text short Call Home messages, that are designed for use with text pagers.
xml	Specifies to send the same text as the long text messages, with the addition of XML tagging and Adaptive Messaging Language (AML) specific transport information to allow machine-readable parsing and correct routing of the message.

Command Default

The default message format is xml.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the message format to short text:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# destination preferred-msg-format short-text
```

destination transport-method

To specify the transport method for Call Home messages for a specific profile, use the **destination transport-method** command in call home profile configuration mode. To disable the transport method, use the **no** form of this command.

destination transport-method email
no destination transport-method email

Syntax Description

email	Email is used to send call home messages.

Command Default

The default transport method is email.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The only transport method supported in this release is email.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the transport method to be email:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# destination transport-method email
```

mail-server

To specify and configure the various mail servers for sending Call Home messages, use the **mail-server** command in call home configuration mode. To remove the mail server configuration, use the **no** form of this command.

mail-server {*ip-addressname*} **priority** *priority*
no mail-server {*ip-addressname*} **priority** *priority*

Syntax Description

ip-address	An IPv4 address to use as the mail server.
name	Name of server to use as the mail server.
priority *priority*	Priority to be used to determine which of multiple configured servers to use as the mail server. Values can be from 1 to 100. A server with a lower priority is tried first.

Command Default

No mail server is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must configure a mail server if the user profile is configured to send email messages only. Only IPv4 addresses are supported.

Up to five mail servers can be configured. Specify a priority for each mail server so the system knows which to try first.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure a mail server:

```
Router# configure
Router(config) call-home
Router(config-call-home)# email-server 209.165.200.225
```

phone-number

To specify a phone number to contact regarding the system, use the **phone-number** command in call home configuration mode. To remove the configured phone number, use the **no** form of this command.

phone-number *phone-number-string*
no phone-number *phone-number-string*

Syntax Description

phone-number-string	Phone number of the contact for the system. The number should always begin with a plus sign (+).

Command Default

No phone number is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The phone number is an optional user-configurable field.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the phone number of the system contact:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# phone-number +15435432101
```

profile (call home)

To enter call home profile configuration mode to create a user-defined profile or configure an existing Call Home profile, use the **profile** command in call home configuration mode. To delete a user-defined profile, use the **no** form of this command.

profile *profile-name*
no profile *profile-name*

Syntax Description

profile-name	A string that identifies the name of profile to create or configure.

Command Default

The tac profile exists by default.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the profile command to create a new profile or modify an existing profile. When you use the **no profile** command, the user-defined profile is deleted and the CiscoTAC-1 profile is set to default. A warning message is displayed when the CiscoTAC-1 profile is set to default.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to create a profile called new-profile:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile new-profile
RP/0/RP0/CPU0:router(config-call-home-profile)#
```

Related Commands

Command	Description
active, on page 22	Enables a Call Home profile.
destination address, on page 34	Specifies an email address to which Call Home messages are sent.

rate-limit

To specify a Call Home event trigger rate limit, use the **rate-limit** command in call home configuration mode. To disable the configured rate limit, use the **no** form of this command.

rate-limit *events-count*
no rate-limit *events-count*

Syntax Description

events-count	Number of events that can be triggered per minute. The default is five events. The maximum is five events.

Command Default

5 events per minute

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read, write

Example

The following example shows how to configure the rate limit to be 3 events per minute:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# rate-limit 3
```

sender

To specify the from and reply-to email addresses for Call Home email messages, use the **sender** command in call-home configuration mode. To remove these email addresses from the configuration, use the **no** form of this command.

sender {**from** | **reply-to**}*email-address*
no sender {**from** | **reply-to**}*email-address*

Syntax Description

from	Specifies the email address to be used as the from field in Call Home email messages.
reply-to	Specifies the email address to be used as the reply-to field in Call Home email messages.
email-address	A string that identifies a valid email address.

Command Default

No sender email is configured.

Command Modes

Call-home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the from and reply-to email addresses:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# sender from user1@cisco.com
RP/0/RP0/CPU0:router(config-call-home)# sender reply-to user1@cisco.com
```

show call-home

To display information regarding the Call Home configuration, use the **show call-home** command in EXEC mode.

show call-home [**detail**]

Syntax Description

detail	Displays Call Home general settings, alert group settings, and all available profiles.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read

The following example shows how to display detailed call home configuration information:

```
Router# show call-home detail

Wed Apr  6 02:00:27.789 DST

Current call home settings:
    call home feature : disable
    call home message's from address: Not yet set up
    call home message's reply-to address: Not yet set up

    contact person's email address: Not yet set up

    contact person's phone number: Not yet set up
    street address: Not yet set up
    customer ID: Not yet set up
    contract ID: Not yet set up
    site ID: Not yet set up

    Mail-server: Not yet set up
    Rate-limit: 5 event(s) per minute

Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- --------------------------------
    environment              Enable  environmental info
    inventory                Enable  inventory info
    syslog                   Enable  syslog info
```

```
Profiles:

Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com

    Periodic inventory info message is scheduled every 24 day of the month at 16:35

    Alert-group               Severity
    ------------------------  ------------
    environment               minor
    inventory                 normal

    Syslog-Pattern            Severity
    ------------------------  ------------
    .*                        major
```

show call-home alert-group

To display available Call Home alert groups, use the **show call-home alert-group** command in EXEC mode.

show call-home alert-group

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to display Call Home alert group information:

```
RP/0/RP0/CPU0:router# show call-home alert-group

Tue Apr  5 06:51:02.860 DST

Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- --------------------------------
    environment              Enable  environmental info
    inventory                Enable  inventory info
    syslog                   Enable  syslog info
```

show call-home mail-server status

To displays the status of the configured mail servers, use the **show call-home mail-server status** command in EXEC mode.

show call-home mail-server status

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read

The following example shows sample output from the **show call-home mail-server status** command:

```
RP/0/RP0/CPU0:router# show call-home mail-server status

Please wait. Checking for mail server status ...

Mail-server[1]: Address: 64.102.124.15 Priority: 50 [Available]
```

show call-home profile

To display the Call Home profiles, use the **show call-home profile** command in EXEC mode.

show call-home profile {**all***profile-name*}

Syntax Description

all	Displays information for all profiles.
profile-name	Name of the profile for which to display information.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read

The following example shows sample output from the **show call-home profile** command:

```
Router# show call-home profile CiscoTAC-1

Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com

   Periodic inventory info message is scheduled every 4 day of the month at 12:19

    Alert-group               Severity
    ------------------------  ------------
    environment               minor

    Syslog-Pattern            Severity
    ------------------------  ------------
    .*                        major
```

show call-home statistics

To display Call Home statistics, use the **show call-home statistics** command in EXEC mode.

show call-home statistics

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read

The following example shows sample output from the **show call-home statistics** command:

```
RP/0/RP0/CPU0:router# show call-home statistics

Message Types    Total                Email
-------------   -------------------- --------------------
Total Success   2                    2
    Environment 0                    0
    Inventory   2                    2
    SysLog      0                    0
    Test        0                    0
    Request     0                    0
    Send-CLI    0                    0

Total In-Queue  0                    0
    Environment 0                    0
    Inventory   0                    0
    SysLog      0                    0
    Test        0                    0
    Request     0                    0
    Send-CLI    0                    0

Total Failed    0                    0
    Environment 0                    0
    Inventory   0                    0
    SysLog      0                    0
    Test        0                    0
    Request     0                    0
    Send-CLI    0                    0

Total Ratelimit
      -dropped  0                    0
```

```
    Environment 0                   0
    Inventory   0                   0
    SysLog      0                   0
    Test        0                   0
    Request     0                   0
    Send-CLI    0                   0

Last call-home message sent time: 2010-04-21 01:06:44 GMT+00:00
```

show call-home trace

To display Call Home trace information, use the **show call-home trace** command in EXEC mode.

show call-home trace {**all** | **error** | **event**} {**file** *filename* **original location** *node-id* | **hexdump** | **last** *n* | **location** {*node-id* | **all**} | **reverse** | **stats** | **tailf** | **unique** | **verbose** | **wrapping**}

Syntax Description

all	(Optional) Displays both error and event traces.
error	(Optional) Displays error trace entries.
event	(Optional) Displays event trace entries.
file *filename* **original location** *node-id*	(Optional) Specifies a filename and original location of the file to display.
hexdump	(Optional) Displays traces in hexadecimal format.
last *n*	(Optional) Displays the last *n* number of traces only.
location {*node-id* | **all**}	Specifies the RP node for which to display the execution path monitoring information. The *node-id* argument is expressed in the *rack/slot* notation. The **all** keyword specifies all RP nodes.
reverse	(Optional) Displays the most recent traces first.
stats	(Optional) Displays execution path statistics.
tailf	(Optional) Displays new traces as they are added.
unique	(Optional) Displays unique entries only, along with the count of the number of times this entry appears.
verbose	(Optional) Displays additional internal debugging information.
wrapping	(Optional) Displays wrapping entries.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
call-home	read

The following example shows how to display Call Home trace information:

```
RP/0/RP0/CPU0:router# show call-home trace event last 15 stats

Wed Apr  6 05:11:59.984 DST
/dev/shmem/ltrace/call_home/trace wrapping: 28.672 Mbytes/sec for 512 entries
59 wrapping entries (512 possible, 0 filtered, 59 total)
Mar  3 13:26:20.281 call_home/trace 0/RSP0/CPU0 t14 Checking mail server access during
boot-up
Mar  3 13:26:20.281 call_home/trace 0/RSP0/CPU0 t9  processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:27:20.283 call_home/trace 0/RSP0/CPU0 t9  Checking mail server access during
boot-up
Mar  3 13:27:20.283 call_home/trace 0/RSP0/CPU0 t14 processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:28:20.285 call_home/trace 0/RSP0/CPU0 t14 Checking mail server access during
boot-up
Mar  3 13:28:20.285 call_home/trace 0/RSP0/CPU0 t9  processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:29:20.287 call_home/trace 0/RSP0/CPU0 t9  Checking mail server access during
boot-up
Mar  3 13:29:20.287 call_home/trace 0/RSP0/CPU0 t14 processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:30:20.289 call_home/trace 0/RSP0/CPU0 t14 Checking mail server access during
boot-up
Mar  3 13:30:20.289 call_home/trace 0/RSP0/CPU0 t9  processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:31:20.290 call_home/trace 0/RSP0/CPU0 t9  Checking mail server access during
boot-up
Mar  3 13:31:20.290 call_home/trace 0/RSP0/CPU0 t14 processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:32:21.067 call_home/trace 0/RSP0/CPU0 t14 Checking mail server access during
boot-up
Mar  3 13:32:21.067 call_home/trace 0/RSP0/CPU0 t9  processing mail server status checking
 event: data1 0x5005784c, str1
Mar  3 13:33:21.069 call_home/trace 0/RSP0/CPU0 t9  Checking mail server access during
boot-up
```

site-id

To specify a site identifier for the system, use the **site-id** command in call-home configuration mode. To disable the site identifier, use the **no** form of this command.

site-id *site-id-string*
no site-id *site-id-string*

Syntax Description

site-id-string	A string that identifies the site.

Command Default

No site ID is defined.

Command Modes

Call-home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The site ID is an optional user-configurable field that can be used to specify a Cisco-supplied site ID or other data meaningful to the support service.

Task ID

Task ID	Operation
call-home	read, write

This example shows how to configure the site ID:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# site-id Cisco-site
```

street-address

To specify the street address of the system, use the **street-address** command in call home configuration mode. To remove the street address configuration, use the **no** form of this command.

street-address *street-address*
no street-address *street-address*

Syntax Description

street address	A string that identifies the street address of the system.

Command Default

No street address is defined.

Command Modes

Call home configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The street address is an optional user-configurable field that can be used to provide the address of the system to the support service.

Task ID

Task ID	Operation
call-home	read, write

This example shows how to configure the street address:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# street-address 170 West Tasman Dr.
San Jose, CA 95134 USA
```

subscribe-to-alert-group inventory

To configure a destination profile to receive messages for the inventory alert group, use the **subscribe-to-alert-group inventory** command in call home profile configuration mode. To disable the subscription, use the **no** form of this command.

subscribe-to-alert-group inventory [**periodic** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *time*]
no subscribe-to-alert-group inventory [**periodic** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *time*]

Syntax Description

periodic	Specifies to send an inventory message periodically.
daily	Sends daily inventory messages.
monthly *day-of-month*	Sends monthly inventory messages on the day of the month specified.
weekly *day-of-week*	Sends weekly inventory messages on the day of the week specified.
time	Time to send the inventory message, in the format `hour:minutes`.

Command Default

The inventory alert group is disabled by default. The default severity for the inventory alert group is normal.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

At least one alert group subscription is mandatory for a profile so that a valid event can be triggered.

There are two types of subscription for an inventory alert group. One is normal subscription, meaning that the severity is set to normal, and any (online insertion and removal) OIR event triggers the event. The second is periodic subscription, using the **periodic** keyword, meaning that an event is triggered only when the specified time arrives.

Task ID

Task ID	Operation
call-home	read, write

This example shows how to configure the sending of inventory messages every Monday:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group inventory periodic
 weekly monday
```

subscribe-to-alert-group syslog

To configure a destination profile to receive messages for the syslog alert group, use the **subscribe-to-alert-group syslog** command in call home profile configuration mode. To disable the subscription, use the **no** form of this command.

subscribe-to-alert-group syslog severity *severity-level* **pattern** *match*
no subscribe-to-alert-group syslog severity *severity-level* **pattern** *match*

Syntax Description

severity	Specifies the the lowest level of severity events to include in a syslog alert.
severity-level	• **catastrophic**—Includes network-wide catastrophic events in the alert. This is the highest severity. • **critical**—Includes events requiring immediate attention (system log level 1). • **debugging**—Includes debug events (system log level 7). This is the lowest severity. • **disaster**—Includes events with significant network impact. • **fatal**—Includes events where the system is unusable (system log level 0). • **major**—Includes events classified as major conditions (system log level 2). • **minor**—Includes events classified as minor conditions (system log level 3) • **normal**—Specifies the normal state and includes events classified as informational (system log level 6). This is the default. • **notification**—Includes events informational message events (system log level 5). • **warning**—Includes events classified as warning conditions (system log level 4).
pattern	Specifies a syslog string pattern to match.
match	A string that when matched in the syslog message, is included in the alert notification. If the pattern contains spaces, you must enclose it in quotes (" ").

Command Default

The syslog alert group is disabled by default. The default severity for the syslog alert group is debugging.

Command Modes

Call home profile configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

At least one alert group subscription is mandatory for a profile so that a valid event can be triggered.

Task ID

Task ID	Operation
call-home	read, write

The following example shows how to configure the syslog alert group to include severity notification:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config) call-home
RP/0/RP0/CPU0:router(config-call-home)# profile my-profile
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group syslog severity
notification pattern "UPDOWN"
```



Cisco Discovery Protocol Commands

- cdp, on page 58
- cdp holdtime, on page 59
- cdp log adjacency changes, on page 60
- cdp timer, on page 61
- clear cdp counters, on page 62
- clear cdp table, on page 63
- show cdp, on page 65
- show cdp entry, on page 67
- show cdp interface, on page 69
- show cdp neighbors, on page 71
- show cdp traffic, on page 74

cdp

To enable the Cisco Discovery Protocol (CDP) globally or on an interface, use the **cdp** command in the appropriate configuration mode. To disable CDP globally or on an interface, use the **no** form of this command.

cdp
no cdp

Syntax Description

This command has no keywords or arguments.

Command Default

CDP is disabled.

Command Modes

Interface configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

By default, CDP is disabled globally. To enable CDP, CDP must be enabled globally and then enabled for each interface.

To enable CDP globally, use the **cdp** command in global configuration mode. To disable CDP globally, use the **no** form of this command in global configuration mode.

To enable CDP on a specific interface, use the **cdp** command in interface configuration mode. To disable CDP on a specific interface, use the **no** form of this command in interface configuration mode.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to globally enable CDP:

```
RP/0/RP0/CPU0:router(config)# cdp
```

The following example shows how to enable CDP on an interface:

```
RP/0/RP0/CPU0:router(config-if)# cdp
```

cdp holdtime

To specify the time for which the receiving device should hold a Cisco Discovery Protocol (CDP) packet from your networking device before discarding it, use the **cdp holdtime** command in the appropriate configuration mode. To remove the **cdp holdtime** command from the configuration file and restore the system to its default condition with respect to this command, use the **no** form of this command.

cdp holdtime *seconds*
no cdp holdtime

Syntax Description

seconds	Holdtime to be sent in the CDP update packets, in seconds. Range is 10 to 255.

Command Default

seconds : 180

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

CDP packets are sent with a time-to-live value or holdtime, that is nonzero after an interface is enabled.

The CDP holdtime must be set to a higher number of seconds than the time between CDP transmissions, which is set using the **cdp time** command.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to specify that the CDP packets sent from the networking device are held by the receiving device for 60 seconds before being discarded. You might want to set the holdtime lower than the default setting of 180 seconds if information about your networking device changes often and you want the receiving devices to purge this information more quickly.

```
RP/0/RP0/CPU0:router(config)# cdp holdtime 60
```

cdp log adjacency changes

To log changes to the Cisco Discovery Protocol (CDP) adjacency table, use the **cdp log adjacency changes** command in the appropriate configuration mode. To disable the logging, use the **no** form of this command.

cdp log adjacency changes
no cdp log adjacency changes

Syntax Description

This command has no keywords or arguments.

Command Default

CDP adjacency table logging is disabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When CDP adjacency table logging is enabled, a syslog is generated each time a CDP neighbor is added or removed.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to enable CDP adjacency table logging:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# cdp log adjacency changes
```

When CDP adjacency table logging is enabled, a syslog is generated each time a CDP neighbor is added or removed. The following is an example of the log entry:

```
LC/0/5/CPU0:Jun 5 10:51:18.081 : cdp[109]: %L2-CDP-6-DELETED_NEIGHBOR :
CDP Neighbour TBA04110127 on interface GigabitEthernet0/5/0/0
  has been deleted, remote interface 3/2

LC/0/5/CPU0:Jun  5 10:51:33.120 : cdp[109]: %L2-CDP-6-NEW_NEIGHBOR :
New CDP neighbor TBA04110127 detected on interface GigabitEthernet0/5/0/0,
  remote interface 3/2
```

cdp timer

To specify how often the software sends Cisco Discovery Protocol (CDP) updates, use the **cdp timer** command in the appropriate configuration mode. To remove the **cdp timer** configuration command from the configuration file and restore the system to its default condition with respect to this command, use the **no** form of this command.

cdp timer *seconds*
no cdp timer

Syntax Description

seconds	Frequency with which the Cisco IOS XR software sends CDP updates, in seconds. Range is 5 to 254. The default is 60.

Command Default

seconds : 60

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

A lower timer setting causes CDP updates to be sent more frequently.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to set the CDP timer to 80 seconds, which is less frequent than the default setting of 60 seconds:

```
RP/0/RP0/CPU0:router(config)# cdp timer 80
```

clear cdp counters

To reset Cisco Discovery Protocol (CDP) traffic counters to zero (0), use the **clear cdp counters** command in EXEC mode.

clear cdp counters location *node-id*

Syntax Description

location *node-id*	Clears CDP traffic counters for the designated node. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

The counters are set to zero.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to clear CDP counters. The **show cdp traffic** output shows that all traffic counters have been reset to zero (0).

```
RP/0/RP0/CPU0:router# clear cdp counters
RP/0/RP0/CPU0:router# show cdp traffic

  CDP counters:
          Packets output: 0, Input: 0
          Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
          No memory: 0, Invalid packet: 0, Truncated: 0
          CDP version 1 advertisements output: 0, Input: 0
          CDP version 2 advertisements output: 0, Input: 0
          Unrecognize Hdr version: 0, File open failed: 0
```

clear cdp table

To clear and automatically resize the table that contains Cisco Discovery Protocol (CDP) information about neighbors, use the **clear cdp table** command in EXEC mode.

clear cdp table location *node-id*

Syntax Description

location *node-id*	Clears and resizes the CDP table for the designated node. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **clear cdp table** command to clear and resize the CDP table that contains the neighbor entries. The new table size is calculated according to the recommended hash table size, as seen in the **show cdp** command output.

Task ID

Task ID	Operations
cdp	read, write

The following example shows how to clear and resize the CDP table. The output of the **show cdp neighbors** command before and after use of the **clear cdp table** command shows that all information has been deleted from the table:

```
RP/0/RP0/CPU0:router# show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
TBA04341195(15la Mg0/RP1/CPU0/0    171        T S         WS-C2924  0/1

RP/0/RP0/CPU0:router# clear cdp table
RP/0/RP0/CPU0:router# show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
```

The **show cdp** command shows that the table has been resized:

```
RP/0/RP0/CPU0:router# show cdp

Global CDP information:
        Sending CDP packets every 60 seconds
        Sending a holdtime value of 180 seconds
        Sending CDPv2 advertisements is enabled
```

show cdp

To display global Cisco Discovery Protocol (CDP) information, including CDP version, timer, and holdtime information, use the **show cdp** command in

EXEC

mode.

show cdp

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values

Command Modes

EXEC

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show cdp** command to display CDP version, timer, and holdtime information relative to CDP operations.

Task ID

Task ID	Operations
cdp	read

The following example shows how to use the **show cdp** command to verify the CDP global settings:

```
RP/0/RP0/CPU0:router# show cdp

Global CDP information:
        Sending CDP packets every 20 seconds
        Sending a holdtime value of 30 seconds
        Sending CDPv2 advertisements is not enabled
```

Table 5: show cdp Field Descriptions

Field	Definition
Sending CDP packets every 20 seconds	Interval between transmissions of CDP advertisements. This field is controlled by the **cdp timer** command.
Sending a holdtime value of 30 seconds	Time for which the device directs the neighbor to hold a CDP advertisement before discarding it. This field is controlled by the **cdp holdtime** command.

Field	Definition
Sending CDPv2 advertisements is not enabled	State of being enabled or disabled for the transmission of CDP version 2-type advertisements. This field is controlled by the **cdp advertise v1** command.

show cdp entry

To display information about a specific neighboring device or all neighboring devices discovered using Cisco Discovery Protocol (CDP), use the **show cdp entry** command in

EXEC

mode.

show cdp entry {*****_entry-name_} [**protocol** | **version**]

Syntax Description

*****	Displays all CDP neighbors.
entry-name	Name of a neighbor about which you want information.
protocol	(Optional) Displays protocol information associated with CDP neighbor entries.
version	(Optional) Displays version information associated with CDP neighbor entries.

Command Default

This command displays information about a particular device that has been discovered by CDP.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
cdp	read, write

The following example shows sample output from the **show cdp entry** command. Information about device ID, address, platform, interface, holdtime, and version is displayed.

```
RP/0/RP0/CPU0:router# show cdp entry TBA04341195

-------------------------
Device ID: TBA04341195(sys-235)
SysName : sys-235
Entry address(es):
  IP address: 172.16.23.9
Platform: WS-C6006,  Capabilities: Trans-Bridge Switch
Interface: MgmtEth0/RP1/CPU0/0
Port ID (outgoing port): 4/18
Holdtime : 157 sec

Version :
```

```
WS-C6006 Software, Version McpSW: 7.2(2) NmpSW: 7.2(2)
Copyright (c) 1995-2002 by Cisco Systems

advertisement version: 2
VTP Management Domain: 'sys'
Native VLAN: 125
Duplex: full
```

Table 6: show cdp entry Field Descriptions

Field	Description
Device ID	ID code assigned during installation of the router.
Entry address(es)	Addresses of the platform, selected interface, and port ID.
Platform	Platform name.
Capabilities	Special functions that the platform can perform (in this case the platform is a trans-bridge switch).
Interface	Interface location expressed in *rack* / *slot* / *module* / *port* notation.
Port ID (outgoing port)	Location of the port in use by the interface.
Holdtime	Time (in seconds) for which the device directs the neighbor to hold a CDP advertisement before discarding it. This field is controlled by the **cdp holdtime** command.
Version	Software version.
advertisement version	Version number of the advertising protocol.
VTP Management Domain	VLAN Trunking Protocol (VTP) domain name of neighbor device.
Native VLAN	VLAN ID.
Duplex	Duplex setting: half or full.

show cdp interface

To display information about the interfaces on which Cisco Discovery Protocol (CDP) is enabled, use the **show cdp interface** command in the appropriate mode.

show cdp interface [*type interface-path-id* | **location** *node-id*]

Syntax Description

type	(Optional) Interface type. For more information, use the question mark (**?**) online help function.
interface-path-id	(Optional) Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
location *node-id*	(Optional) Displays detailed CDP information for the designated node. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

This command displays information about the interfaces on which CDP has been enabled.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show cdp interface** command to display information about any CDP interfaces. When an interface is specified in the command syntax, information is displayed about the specific interface. Not specifying the interface displays information about all interfaces.

Task ID

Task ID	Operations
cdp	read, write

The following example shows sample output from the **show cdp interface** command. Information about the status, CDP timer, and holdtime settings is displayed for all interfaces on which CDP is enabled.

```
RP/0/RP0/CPU0:router# show cdp interface

POS0/2/0/0 is Up
    Encapsulation HDLC
    Sending CDP packets every 120 seconds
```

```
    Holdtime is 240 seconds
POS0/2/0/1 is Up
    Encapsulation HDLC
    Sending CDP packets every 120 seconds
    Holdtime is 240 seconds
POS0/2/0/2 is Up
    Encapsulation HDLC
    Sending CDP packets every 120 seconds
    Holdtime is 240 seconds
POS0/2/0/3 is Up
    Encapsulation HDLC
    Sending CDP packets every 120 seconds
    Holdtime is 240 seconds
MgmtEth0/RP1/CPU0/0 is Up
    Encapsulation ARPA
    Sending CDP packets every 120 seconds
    Holdtime is 240 seconds
```

The following example shows sample output from the **show cdp interface** command with an interface specified. Information about the status, CDP timer, and holdtime settings is displayed for Packet-over-SONET/SDH (POS) interface 0/2/0/1 only.

```
RP/0/RP0/CPU0:router# show cdp interface pos 0/2/0/1

POS0/2/0/1 is Up
    Encapsulation HDLC
    Sending CDP packets every 60 seconds
    Holdtime is 180 seconds
```

Table 7: show cdp interface Field Descriptions

Field	Description
POS0/2/0/1 is Up	Current condition of POS interface 0/0/2/1.
Encapsulation HDLC	Interface is encoding packets using the Cisco HDLC Layer 2 encapsulation.
Sending CDP packets every 60 seconds	Interval between transmissions of CDP advertisements. This field is controlled by the **cdp timer** command.
Holdtime is 180 seconds	Time for which the device directs the neighbor to hold a CDP advertisement before discarding it. This field is controlled by the **cdp holdtime** command.

show cdp neighbors

To display detailed information about neighboring devices discovered using Cisco Discovery Protocol (CDP), use the **show cpd neighbors** command in

EXEC

mode.

show cdp neighbors [*type interface-path-id* | **location** *node-id*] [**detail**]

Syntax Description

type	(Optional) Interface type. For more information, use the question mark (**?**) online help function.
interface-path-id	(Optional) Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
location *node-id*	(Optional) Displays detailed CDP information for the designated node. The *node-id* argument is entered in the *rack/slot* notation.
detail	(Optional) Displays detailed information about a neighbor or neighbors, including network address, enabled protocols, holdtime, and software version. The output includes information about both IPv4 and IPv6 addresses.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show cpd neighbors** command to display information about any CDP neighbors. When a location is specified in the command syntax, information about the neighbor is displayed for the specified node. Not specifying the location displays information about the neighbor for all interfaces.

Use the command with the **detail** keyword to display additional information, including IPv6 neighbors.

Task ID

Task ID	Operations
cdp	read

The following example shows sample output from the **show cpd neighbors** command:

```
RP/0/RP0/CPU0:router# show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater

Device ID        Local Intrfce   Holdtme  Capability  Platform  Port ID

TBA04110127      Gi0/7/0/0       173      T S         WS-C6506  3/9
cisco_1          Gi0/7/0/2       171      R
ASR9K
     Gi0/4/0/2
```

Table 8: show cdp neighbors Field Descriptions

Field	Description
Capability Codes	Type of device that can be discovered.
Device ID	Name of the neighbor device.
Local Intrfce	Protocol being used by the connectivity media and the interface number.
Holdtme	Remaining time, in seconds, for which the current device holds the CDP advertisement from a sending router before discarding it.
Capability	Type of the device listed in the CDP Neighbors table. Values are as follows: R—Router T—Transparent bridge B—Source-routing bridge S—Switch H—Host I—Internet Group Management Protocol (IGMP) device r—Repeater
Platform	Product number of the device.
Port ID	Protocol and port number of the device.

The following examples illustrates sample output for IPv4 and IPv6 neighbors from the **show cpd neighbors** command with the **detail** keyword:

```
RP/0/RP0/CPU0:router# show cdp neighbor detail

-------------------------
Device ID: uut-user
SysName : uut-user
Entry address(es):
IPv4 address: 1.1.1.1
IPv6 address: 1::1
IPv6 address: 2::2
Platform: cisco 12008/GRP, Capabilities: Router
```

```
Interface: Gi0/4/0/3
Port ID (outgoing port): Gi0/2/0/3
Holdtime : 177 sec

Version :
Cisco IOS XR Software, Version 0.0.0[Default]
Copyright (c) 2005 by cisco Systems, Inc.

advertisement version: 2
```

Table 9: show cdp neighbors detail Field Descriptions

Field	Definition
Device ID	Name of the neighbor device.
Entry address(es)	List of network addresses of neighbor devices. The address can be in IP or in Connectionless Network Service (CLNS) protocol conventions.
Platform	Product name and number of the neighbor device.
Capabilities	Device type of the neighbor. This device can be a router, a bridge, a transparent bridge, a source-routing bridge, a switch, a host, an IGMP device, or a repeater.
Interface	Interface being used by the connectivity medium.
Port ID	Port number of the port on the current device.
Holdtime	Remaining time (in seconds) for which the current device holds the CDP advertisement from a sending router before discarding it.
Version	Software version of the neighbor device.
advertisement version	Version number of the advertising protocol.

show cdp traffic

To display information about the traffic gathered between devices using Cisco Discovery Protocol (CDP), use the **show cdp traffic** command in

EXEC

mode.

show cdp traffic [**location** *node-id*]

Syntax Description

location *node-id*	(Optional) Displays CDP information for the CDP packets sent and received on the designated node only. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

Displays CDP information aggregated across all nodes.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
cdp	read

The following example illustrates sample output from the **show cdp traffic** command:

```
RP/0/RP0/CPU0:router# show cdp traffic

CDP counters :
          Packets output: 50662, Input: 40414
          Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
        No memory: 0, Invalid packet: 0, Truncated: 0
        CDP version 1 advertisements output: 0, Input: 0
        CDP version 2 advertisements output: 50662, Input: 40414
          Unrecognize Hdr version: 0, File open failed: 0
```

Table 10: show cdp traffic Field Descriptions

Field	Definition
Packets output	Number of CDP advertisements sent by the local device. Note that this value is the sum of the CDP version 1 advertisements output field and the CDP version 2 advertisements output field.

Field	Definition
Input	Number of CDP advertisements received by the local device. Note that this value is the sum of the CDP version 1 advertisements input field and the CDP version 2 advertisements input field.
Hdr syntax	Number of CDP advertisements having bad headers that have been received by the local device.
Chksum error	Number of times the checksum (verifying) operation failed on incoming CDP advertisements.
Encaps failed	Number of times CDP failed to send advertisements on an interface because of a failure caused by the bridge port of the local device.
No memory	Number of times that the local device did not have enough memory to store the CDP advertisements in the advertisement cache table when the device was attempting to assemble advertisement packets for transmission and parse them when receiving them.
Invalid packet	Number of invalid CDP advertisements received and sent by the local device.
Truncated	Number of times truncated CDP advertisements were sent because there was not enough space in the CDP packet to hold all CDP type-length-values (TLVs).
CDP version 1 advertisements output	Number of CDP version 1 advertisements sent by the local device.
Input	Number of CDP version 1 advertisements received by the local device.
CDP version 2 advertisements output	Number of CDP version 2 advertisements sent by the local device.
Input	Number of CDP version 2 advertisements received by the local device.
Unrecognize Hdr version	Number of packets received from a CDP version that was outside the current configuration.
File open failed	Number of times that CDP failed to connect to one of the underlying services it uses.



Configuration Management Commands

- abort, on page 79
- alias, on page 80
- apply-template, on page 83
- clear comment, on page 84
- clear configuration commits, on page 85
- clear configuration inconsistency, on page 87
- clear configuration inconsistency replica, on page 89
- clear configuration sessions, on page 90
- commit, on page 91
- configuration commit auto-save, on page 95
- configure, on page 97
- description (interface), on page 99
- do, on page 100
- end, on page 101
- end-group, on page 103
- end-template, on page 104
- exclude-group, on page 105
- exit, on page 106
- hostname, on page 108
- load, on page 109
- load commit changes, on page 111
- load configuration failed, on page 112
- load configuration removed, on page 113
- load rollback changes, on page 114
- man, on page 115
- more, on page 118
- pwd (config), on page 122
- rollback configuration, on page 123
- root, on page 126
- save configuration, on page 128
- save configuration changes, on page 129
- save configuration commit changes, on page 130
- save configuration failed, on page 131

- save configuration merge, on page 132
- save configuration removed, on page 133
- save rollback changes, on page 134
- service cli commit-optimized enable, on page 135
- set default-afi, on page 136
- set default-safi, on page 137
- set default-vrf, on page 138
- show, on page 140
- show aliases, on page 143
- show apply-group, on page 144
- show running-configuration filter vrf, on page 145
- show configuration failed, on page 147
- show configuration changes, on page 148
- show configuration commit changes, on page 150
- show configuration commit list, on page 154
- show configuration (config), on page 156
- show configuration failed incompatible, on page 158
- show configuration failed remove, on page 159
- show configuration failed rollback, on page 161
- show configuration failed startup, on page 162
- show configuration history, on page 163
- show configuration inconsistency replica, on page 166
- show configuration persistent, on page 167
- show configuration removed, on page 169
- show configuration rollback changes, on page 170
- show configuration running, on page 172
- show configuration running-config, on page 173
- show configuration sessions, on page 175
- show default-afi-safi-vrf, on page 177
- show history, on page 178
- template, on page 180

abort

To terminate a configuration session and discard all uncommitted changes without system confirmations, use the **abort** command in any configuration mode.

abort

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Any configuration mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **abort** command to terminate a configuration session and return to EXEC mode from any configuration mode. This command discards all uncommitted configuration changes. You are prompted to commit the changes.

Task ID

Task ID	Operations
Task ID for the feature or mode impacted by the command	Operation for the feature or mode impacted by the command

The following example shows how to use the **abort** command to discard all changes made during a configuration session:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)#
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.1.1.1 255.0.0.0
RP/0/RP0/CPU0:router(config-if)# abort
RP/0/RP0/CPU0:router#
```

alias

To create a command alias, use the **alias** command in XR Config mode. To delete an alias, use the **no** form of this command.

alias *alias-name*[*(param-list)*]*content*
no alias *alias-name*

Syntax Description

alias-name	Name of the command alias. Alias names can be a single word or multiple words joined by a hyphen (-) or an underscore (_).
param-list	(Optional) Parameters assigned to the alias. These parameters are filled in at execution time.
content	Original command syntax. Valid abbreviations of the original command syntax can be entered for the *content* argument.

Command Default

No command aliases are configured.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Cisco IOS XR software supports generic alias definitions for various entities. Any physical or logical entity can have an alias as a reference. For example, an alias can refer to a command, a partial command, a group of commands, a location, or an IP address.

An alias must first be defined. The alias can then be used in command lines in place of the defined entity.

Following is a list of properties for an alias:

- An alias can be used anywhere and in any mode.
- An alias can have zero, one, or many parameters.
- An alias can refer to those parameters with the $ sign.
- If an alias refers to more than one command, the commands must be separated by a semicolon (;).
- The size of the **alias** command is limited to 1024 characters.

The alias command can be used anywhere. If the content referenced by the alias is invalid or inappropriate in that context or mode, the system issues a warning message containing the substituted content.

An alias name should not be a subset of the keywords that it represents as alias. Substitution is done only when the entered input match fails completely. For instance, the attempt to define an alias with “config? as the alias name fails, as shown in the following example:

```
RP/0/RP0/CPU0:router(config)# alias config set_host hostname router
RP/0/RP0/CPU0:router(config)# show configuration

alias set_host hostname router
```

Use the **show aliases** command to display all command aliases or the command aliases in a specified mode.

Task ID

Task ID	Operations
logging	read, write

The following example shows how to create an alias named ipbr for the **show ipv4 interface brief** command, commit the configuration, enter XR EXEC mode and then enter the configured alias:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# alias ipbr show ipv4 interface brief
RP/0/RP0/CPU0:router(config)# show configuration

Building configuration...
alias ipbr show ipv4 interface brief
end
RP/0/RP0/CPU0:router(config)# commit
RP/0/RSP0/CPU0:Feb 21 04:42:57.017 : config[65689]: %MGBL-LIBTARCFG-6-COMMIT :
Configuration committed by user 'lab'.   Use 'show configuration commit changes 1000000022'
 to view the changes.
RP/0/RSP0/CPU0:router(config)# end
RP/0/RP0/CPU0:Mar 27 22:19:05 : config[65739]: %SYS-5-CONFIG_I : Configured from console
by lab
RP/0/RP0/CPU0:router# ipbr
RP/0/RP0/CPU0:router# show ipv4 interface brief

Interface                      IP-Address      Status                Protocol
Loopback0                      1.1.1.1         Up                    Up
Loopback999                    unassigned      Up                    Up
MgmtEth0/0/CPU0/0              12.29.56.21     Up                    Up
RP/0/RP0/CPU0:router#
```

The following example shows how to define an alias, mycompany-10ge, for POS interface 1/0/2/3 and then how to use that alias to shut down the interface:

```
RP/0/RP0/CPU0:router(config)# alias mycompany-10ge gigabitethernet1/0/2/3
RP/0/RP0/CPU0:router(config)# interface mycompany-10ge
RP/0/RP0/CPU0:router(config-if)# shutdown
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)#
```

The following example shows the use of a parameter name in an alias definition:

```
RP/0/RP0/CPU0:router(config)# alias shint (intname) show interface $intname
```

The following example shows an alias defined with one parameter and two commands:

```
RP/0/RP0/CPU0:router(config)# alias shint_both (intname) show interface $intname;show run
interface $intname
```

The following example shows the use of the alias shint_both inEXEC mode:

```
RP/0/RP0/CPU0:router(exec)# shint_both(gigabitethernet1/2/3/4)
```

Two commands are issued, as follows:

```
RP/0/RP0/CPU0:router(exec)# show interface gigabitethernet1/2/3/4; show run interface
gigabitethernet1/2/3/4
```

apply-template

To apply a template to the target configuration, use the **apply-template** command in XR Config mode.

apply-template *template-name* [**(***param-list***)**]

Syntax Description

template-name	Name of the template to be applied to the running configuration. Use the **template** command to define a template.
param-list	(Optional) Up to five template parameters.

Command Default

No templates are applied to the target configuration.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **apply-template** command to apply a template to the target configuration. Templates allow you to create a template name that represents a group of configuration commands.

Use the **template** command to define a template. Use the **end-template** command to exit template configuration mode and return to global configuration mode. Use the **show-running** command with the optional **template** *template-name* keyword and argument to display the contents of a template.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to define a template and then apply the template to the target configuration:

```
RP/0/RP0/CPU0:router(config)# template hostname-template
RP/0/RP0/CPU0:router(config-TPL)# hostname router1
RP/0/RP0/CPU0:router(config-TPL)# end-template
RP/0/RP0/CPU0:router(config)# apply-template hostname-template
```

clear comment

To discard a comment associated with a configuration, use the **clear comment** command in any configurationorXR Config mode.

clear comment

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Any configuration mode

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **clear comment** command clears any comments that were added for a specific configuration in the configuration file. After you enter the **clear comment** command, enter the configuration for which you want to delete the comment on a separate line.

To enter configuration comments, enter ! followed by the comment. The comment you enter is associated with the next configuration entered. For example:

```
RP/0/RP0/CPU0:router#!router1 is located in xxx
RP/0/RP0/CPU0:router# hostname router1
RP/0/RP0/CPU0:router# commit
```

The comment is displayed in the output of the **show running-config** command:

```
RP/0/RP0/CPU0:router# show running-config
...
!router1 is located in xxx
hostname router1
...
```

Task ID

Task ID	Operations
Task ID for the feature or configuration mode impacted by the command	Operation for the feature or configuration mode impacted by the command

The following example shows how to discard the comment associated with the configuration ipv4 address 1.1.1.1 255.0.0.0.

```
RP/0/RP0/CPU0:router(config-if)# clear comment
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.1.1.1 255.0.0.0
```

clear configuration commits

To delete old commit IDs from the commit database to free up disk space, use the **clear configuration commits** command in or XR EXEC mode.

clear configuration commits {**diskspace** *kilobytes* | **oldest** *number-of-commits*}

Syntax Description

diskspace *kilobytes*	Deletes as many commit IDs (beginning with the oldest available commit ID) from the commit database as required to free the number of kilobytes (KB) specified for the *kilobytes* argument. The range for the number of kilobytes of disk space to free is 1 to 4194304. **Note** The amount of disk space freed may vary depending on the size and number of commits present in the commit database.
oldest *number-of-commits*	Deletes the number of commit IDs specified for the *number-of-commits* argument. **Note** Use the online help (**?**) function to display the range of commit IDs available for deletion.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **clear configuration commits** command to delete the number of commit IDs available for rollback operations. The most recent 100 commits are retained by the system. As new commit IDs are added, the oldest commit IDs are discarded and are no longer available for rollback operations.

Note The **clear configuration commits** command deletes commits from the commit database only. The running configuration, thus, is not changed.

Note When a commit ID is deleted from the commit database, it is no longer available for rollback and can no longer be used to display commit changes (with the **show configuration rollback changes** command).

Use the **rollback configuration** command to roll back the current running configuration to a previous configuration. Use the **show configuration rollback changes** command to display a list of the commit IDs available for rollback operations or to display the changes that would be made by the **rollback configuration** command.

Task ID

Task ID	Operations
config-services	execute

The following example shows how to delete the oldest 16 commit IDs to free up disk space. After entering this command, you will be prompted to confirm the deletion.

```
RP/0/RP0/CPU0:router# clear configuration commits oldest 16

Deleting 16 rollback points '1000000021' to '1000000036'
256 KB of disk space will be freed. Continue with deletion?[confirm] y
```

clear configuration inconsistency

To clear an inconsistency alarm for a router configuration, use the **clear configuration inconsistency** command in XR EXEC mode.

clear configuration inconsistency

Syntax Description

This command has no keywords or arguments.

Command Default

EXEC mode: Clears the inconsistency alarms for an SDR configuration.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 3.0.0	This command was introduced.

Usage Guidelines

An inconsistency alarm is set when there is a failure to restore the configuration; this can occur during router startup, or when a line card or route processor (RP) card is inserted or when there is an OIR (Online Insertion and Removal).

When the inconsistency alarm is set, all configuration commit operations fail until the alarm is cleared using the **clear configuration inconsistency** command. This command clears the alarm and removes the failed configuration.

Enter the **clear configuration inconsistency** command to clear the alarm and allow commit operations to continue.

Note To reapply the failed configuration, you must reapply and recommit the configuration. Use the **load configuration failed** command with the **startup** keyword to populate the target configuration with the contents of the previous failed configuration from the startup configuration.

Use the **show configuration history** command with the **alarm** keyword to view the inconsistency alarm set and alarm clear events in the configuration history log.

Command Modes

To clear the inconsistency alarms for the router, enter the **clear configuration inconsistency** command in XR EXEC mode.

Task ID

Task ID	Operations
config-services	execute

The following example shows how to clear the inconsistency alarms for a router configuration. The command is entered in EXEC mode.

```
Router# clear configuration inconsistency

Creating any missing directories in Configuration File system...OK
```

```
Initializing Configuration Version Manager...OK
Syncing commit database with running configuration...OK
```

In the following example, a history of the inconsistency alarms set and cleared for the configuration are displayed using the **show configuration history** command with the **alarm** keyword:

```
Router# show configuration history alarm

  Sno.  Event      Info                          Time Stamp
  ~~~~  ~~~~~      ~~~~                          ~~~~~~~~~~
  1     alarm      inconsistency alarm raised    Thu Jun 22 15:23:15 2009
  2     alarm      inconsistency alarm cleared   Thu Jun 22 15:42:30 2009
  3     alarm      inconsistency alarm raised    Sun Jul  9 13:39:57 2009
  4     alarm      inconsistency alarm cleared   Sun Jul  9 14:15:48 2009
  5     alarm      inconsistency alarm raised    Sat Jul 15 18:18:26 2009
  6     alarm      inconsistency alarm cleared   Sat Jul 15 19:21:03 2009
```

clear configuration inconsistency replica

To resolve configuration inconsistencies on a replica node, use the **clear configuration inconsistency replica** command in administration EXECorXR EXEC mode.

clear configuration inconsistency replica location *node-id*

Syntax Description

location *node-id*	Resolves the configuration inconsistencies on the designated node. The *node-id* argument is expressed in the *rack/slot* notation.

Command Default

Administration EXEC mode: Resolves any configuration inconsistencies for the admin plane configuration.

XR EXEC mode: Resolves any configuration inconsistencies for the configuration.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

In administration EXEC mode, the replica node for the **clear configuration inconsistency replica** command is the standby designated system controller (DSC). In EXEC mode, the replica nodes are the route switch processors (RSPs) that can become the designated shelf controller (DSC).

Use the **clear configuration inconsistency replica** command if there is a configuration inconsistency between the standby DSC and the current active DSC; or alternatively, if the configuration on any nodes that could become the DSC is not the same as the configuration on the current DSC. To determine if you have a configuration inconsistency, use the **show configuration inconsistency replica** command.

To clear configuration inconsistencies for the admin plane configuration, enter the **clear configuration inconsistency replica** command in administration EXEC mode.

To clear configuration inconsistencies for an SDR configuration, enter the **clear configuration inconsistency replica** command in EXEC mode for that SDR.

Task ID

Task ID	Operations
config-services	execute

The following example shows how to clear any configuration inconsistencies for the DSC configuration by using the **clear configuration inconsistency replica** command in EXEC mode:

```
RP/0/RP0/CPU0:router# clear configuration inconsistency replica location 0/rp1/cpu0

The replica has been repaired.
```

clear configuration sessions

To clear (end) an active configuration session, use the **clear configuration sessions** command in administration EXEC orXR EXEC mode.

clear configuration sessions *session-id*

Syntax Description

session-id	Identifier for the configuration session to be terminated.

Command Default

None

Command Modes

Administration EXEC

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **clear configuration sessions** command to clear a configuration session. This command can be used to end the configuration sessions of another user. Any uncommitted changes to a user's target configuration are discarded.

Use the **show configuration sessions** command to identify active configuration sessions.

When a configuration session is cleared, a message is displayed on the terminal of the terminated user. For example:

```
RP/0/RP0/CPU0:router(config)# This configuration session was terminated by user 'user_a'
from line 'aux0_0_CPU0'
```

Task ID

Task ID	Operations
config-services	execute

The following example shows how to clear an active configuration session. In this example, the **show configuration sessions** command displays the active configuration session. The **clear configuration sessions** command clears the active configuration session.

```
RP/0/RP0/CPU0:router# show configuration sessions

Current Configuration Session   Line           User      Date                      Lock
00000211-002c409b-00000000      con0_1_CPU0  UNKNOWN   Mon Feb  2 01:02:09 2009

RP/0/RP0/CPU0:router# clear configuration sessions 00000211-002c409b-00000000

session ID '00000211-002cb09b-00000000' terminated
```

commit

To commit the target configuration to the active (running) configuration, use the **commit** command in any configuration XR Config mode System Admin Config mode.

commit [**best-effort**] [**comment** *line*] [**confirmed** [*seconds* | **minutes** *minutes*]] [**force**] [**label** *line*] [**replace**] [**save-running filename** *file_path*]

Syntax Description

best-effort	(Optional) Merges the target configuration with the running configuration and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.
comment *line*	(Optional) Assigns a comment to a commit. This text comment is displayed in the commit entry displayed in the output for the **show configuration commit list** command with the optional **detail** keyword.
confirmed [*seconds* | **minutes** *minutes*]	(Optional) Commits the configuration on a trial basis for the time specified in seconds or minutes. **Note** The **confirmed** option is not available in administration configuration mode.
force	(Optional) Forces a commit operation in low-memory conditions.
label *line*	(Optional) Assigns a meaningful label. This label is displayed (instead of the autogenerated commit ID) in the output for the **show configuration commit list** .
replace	(Optional) Replaces the entire running configuration with the contents of the target configuration.
save-running filename *file_path*	(Optional) Saves the running configuration to a specified file.

Command Default

The default behavior is *pseudo-atomic*, meaning that all changes must succeed for the entire commit operation to succeed. If any errors are found, none of the configuration changes take effect.

Command Modes

Any configuration mode

XR Config mode

System Admin Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Changes made during a configuration session are inactive until the **commit** command is entered. By default, the commit operation is *pseudo-atomic*, meaning that all changes must succeed for the entire commit operation to succeed. If any errors are found, none of the configuration changes takes effect.

To replace the default numeric ID for the commit, use the optional **label** keyword. This label is displayed (instead of the autogenerated commit ID) in the output for the **show configuration commit list** command.

Enter an optional comment with the **comment** keyword to provide additional information about the commit action. This comment is displayed in the output for the **show configuration commit list** command with the **detail** keyword.

Use the optional **confirmed** *minutes* keyword and argument to commit a configuration on a trial basis for a minimum of 30 seconds and a maximum of 300 seconds (5 minutes). During the trial configuration period, enter the **commit** command to confirm the configuration. If the **commit** command is not entered, then the system reverts to the previous configuration when the trial time period expires. The confirmed option is not available in administration configuration mode.

You can use the **commit** command in conjunction with the **load** command. Load a new configuration with the **load** command, and use the **commit** command with the **replace** keyword to have the loaded configuration become the active (running) configuration.

Note The **commit replace** operation supports a maximum of five concurrent active SSH sessions. If more than five SSH sessions are active, the **commit replace** command may fail.

Use the optional **save-running filename** *file_path* keywords and argument to save the running configuration to a specified file. To configure automatic saving of the configuration file on every commit, use the **configuration commit auto-save** command. If automatic saving of the configuration file is already enabled, specifying **save-running filename** *file_path* with the **commit** command has no additional effect.

In pseudo-atomic commit, if an error occurs on one or more of the configurations in a commit, other configurations which are already part of the running configuration in the same commit are reverted.

Caution Saving the running configuration to a file is CPU intensive.

Note If you use the **commit** command without previously loading a target configuration, a blank configuration is committed.

Note If you use the **commit** command with the **replace** keyword, it does not affect the mode of an 8-port E1/T1 SPA. If the mode is E1 before using the **commit replace** command, it remains E1. However, since the default mode is T1, the router does not recognize that the mode is E1. To change the mode to T1, you must first use the **hw-module subslot cardtype e1** command to add the E1 mode into the configuration so that it correlates with the system. Then manually reload the router and it boots in T1 mode.

For more information regarding the **hw-module subslot cardtype** command, refer to *Interface and Hardware Component Command Reference for Cisco 8000 Series Routers*.

Warning Performing removal and reconfiguration of services with varying attributes within a single commit may result in unpredictable behavior such as mis-programming and so on. We recommend that you implement service configuration changes incrementally to ensure accurate control and forwarding plane programming.

Task ID

Task ID	Operations
Task ID for the feature or configuration mode impacted by the command	Operation for the feature or configuration mode impacted by the command

Committing the Target Configuration to the Active Running Configuration

The following example shows how to commit the target configuration to the active running configuration. In this example, the **commit** command saves changes to the router hostname.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hostname router1
RP/0/RP0/CPU0:router(config)# commit

RP/0/RSP0/CPU0:Feb 21 04:42:57.017 : config[65689]: %MGBL-LIBTARCFG-6-COMMIT :
Configuration committed by user 'user_a'.
Use 'show configuration commit changes 1000000033' to view the changes.
```

Adding a Comment to a Configuration Commit

The following example shows how to use the **commit** command with the optional **comment** *line* keyword and argument to assign a text description to the commit operation. The comment is then displayed in the output of the **show configuration commit list** command with the **detail** keyword.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hostname router2
RP/0/RP0/CPU0:router(config)# commit comment new name for router

RP/0/RP0/CPU0:Feb 21 04:42:57.017 : config[65689]: %MGBL-LIBTARCFG-6-COMMIT :
Configuration committed by user 'user_a'.   Use 'show configuration commit
changes 1000000226' to view the changes.

RP/0/RP0/CPU0:router2(config)# end
RP/0/RP0/CPU0:router2# show configuration commit list detail

1) CommitId: 1000000226                Label: NONE
   UserId:   user_a                    Line:  con0_RP1_CPU0
   Client:   CLI                       Time:  12:59:26 UTC Wed Feb 04 2004
   Comment:  new name for router

2) CommitId: 1000000225                Label: NONE
   UserId:   user_a                    Line:  con0_RP1_CPU0
   Client:   CLI                       Time:  12:58:32 UTC Wed Feb 04 2004
   Comment:  NONE
```

Changing the Commit ID to a Text Label

The following example shows how to use the **commit** command with the optional **label** *line* keyword and argument to change the commit ID to a text label for easier identification. The label is then displayed in the output of the **show configuration commit list** command.

```
RP/0/RP0/CPU0:router2# configure
RP/0/RP0/CPU0:router2(config)# hostname router3
RP/0/RP0/CPU0:router2(config)# commit label new_name

RP/0/RP0/CPU0:Feb 21 04:42:57.017 : config[65689]: %MGBL-LIBTARCFG-6-COMMIT :
Configuration committed by user 'user_a'.
Use 'show configuration commit changes 1000000227' to view the changes.

RP/0/RP0/CPU0:router3(config)# end
RP/0/RP0/CPU0:router3# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    new_name    user_a    con0_1_C  CLI         13:00:53 UTC Wed Feb 04 2004
2    1000000226  user_a    con0_1_C  CLI         12:59:26 UTC Wed Feb 04 2004
3    1000000225  user_a    con0_1_C  CLI         12:58:32 UTC Wed Feb 04 2004
```

Commit a Configuration for a Specified Time

The following example shows how to use the **commit** command with the optional **confirmed** keyword and number *argument*. The configuration changes are committed only for the specified number of seconds. You can then either confirm the commit operation or discard the changes.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hostname router3
RP/0/RP0/CPU0:router(config)# commit confirmed 30
RP/0/RP0/CPU0:router3(config)# end
```

configuration commit auto-save

To enable automatic saving of the running configuration to a specified file on every commit, use the **configuration commit auto-save** command in XR Config mode mode. To disable automatic saving of the running configuration to a specified file on every commit, use the **no** form of the command.

configuration commit auto-save [**filename** *file_path* | **password** { **clear** { **encryption-aes** | **encryption-default** } *password* | **encrypted** | { **encryption-aes** | **encryption-default** } *password* } | **maximum** *number* | **timestamp** | **wait-time** *duration*]

Syntax Description

filename *file_path*	Specifies the location to which to save the running configuration.
password *password*	Specifies the password of the remote URL.
clear	Specifies an unencrypted password.
encryption-aes	Saves the password in aes encrypted form.
encryption-default	Saves password in default encrypted form.
encrypted	Specifies an encrypted password.
maximum *number*	Specifies how many maximum auto-saves are possible.
timestamp	Includes the timestamp in the auto-save filename.
wait-time*duration*	Specifies how long to wait to auto-save after the commit done to start the next auto-save.The *duration*can be specified as: days *days* | hours *hours* | minutes *minutes* | seconds *seconds*

Command Default

None

Command Modes

XR Config mode

System Admin Config mode

Command History

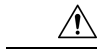
Release	Modification
Release 7.10.1	This command was modified to support public key authentication.
Release 7.9.1	This command was modified to include, **scp**,**sftp**, **password**, **wait-time**, **timestamp**, and **maximum** keywords.
Release 7.0.12	This command was introduced

Usage Guidelines

The **configuration commit auto-save** command configures the system to save the running configuration to the specified file and location every time a **commit** command is run. Alternatively, you can save the configuration on a one-time basis by specifying the **save-running** keyword when you run the **commit** command. Use the following syntax when using **tftp**, **ftp**, **scp**, **sftp** or **rcp** as options.

From Cisco IOS XR Release 7.9.1 scp and sftp options are available under **filename**.

[***tftp*|*ftp**rcp**scp**sftp***]



Caution Saving the running configuration to a file is CPU intensive.

Task ID

Task ID	Operations
config-services	write

The following example shows how to configure the system to save the running configuration to the file `//test-folder/test_123`, backup encrypted password,append time-stamp, configure maximum number of auto-save files possible, and specify wait-time before backing up the files whenever the configuration is commited:

```
Router#configure
Router(config)#configuration commit auto-save
Router(config-cfg-autosave)#filename sftp://user1@server1://test-folder/test_123
Router(config-cfg-autosave)#password clear encryption-default cisco
Router(config-cfg-autosave)#timestamp
Router(config-cfg-autosave)#maximum 10
Router(config-cfg-autosave)#wait-time days 0 hours 0 minutes 0 seconds 5
Router(config-cfg-autosave)#commit
```

While you are using public key authentication to save the running configuration, you don't need to mention password.

configure

To enter global configuration mode or administration configuration mode, use the **configure** command inXR EXEC mode or .

configure [**exclusive** | **terminal**]

Syntax Description

exclusive	(Optional) Locks the router configuration. The system configuration can be made only from the login terminal.
terminal	(Optional) Configures the system from the login terminal. This is the default.

Command Default

If the **configure** command is entered without a keyword, the system is configured from the login terminal.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Configuration modes are used to enter changes to a target configuration session and commit those changes to the running configuration. A router running Cisco IOS XR software contains multiple configurations:

- The configuration for a router. This mode is used to configure - specific features such as routing protocols.
- The administration configuration for system-wide resources and settings. Some features can be configured only in administration configuration mode.

XR Config mode

Use the **configure** command in XR EXEC mode to enter XR Config mode and create a new target configuration for an SDR. From global configuration mode, you can enter any configuration mode. Configuration changes entered in global configuration mode impact the SDR to which the user is currently logged in.

System Admin Config mode

Use the **configure** command in to enter System Admin Config mode and create a new target configuration. From , you can enter any configuration mode. Configuration changes entered in can impact resources for the entire router. See the command reference documentation for a specific command to determine the impact of commands entered in .

Router Prompt

After you enter the **configure** command, the system appends "(config)" to the router prompt, indicating that the router is in a configuration mode. For example:

- The following prompt indicates that you are in global configuration mode for an SDR:

```
RP/0/RP0/CPU0:router(config)#
```

- The following prompt indicates that you are in administration configuration mode:

```
RP/0/RP0/CPU0:router(admin-config)#
```

Locking a Configuration Session

To lock the configuration so that no other user can commit changes to the running configuration during your configuration session, issue the **configure** command with the **exclusive** keyword.

Committing Changes and Returning to XR EXEC mode or

Changes to the target configuration remain inactive until the **commit** command is entered. To leave global configuration or administration configuration mode and return to the XR EXEC mode or prompt, issue the **end** or **exit** command; you are prompted to commit any uncommitted changes.

To leave configuration mode and return directly to XR EXEC mode or without being prompted to commit changes and without saving changes to the target configuration, enter the **abort** command in any configuration mode.

The following example shows how to enter global configuration mode from XR EXEC mode and then enter interface configuration mode to configure an IPv4 address , the **configure** command commits the configuration, and the **end** command terminates the configuration session and return the router to XR EXEC mode.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.1.1.1 255.0.0.0
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router(config-if)# end
RP/0/RP0/CPU0:router#
```

description (interface)

To add a description to an interface configuration, use the **description** command in interface configuration mode. To remove the description, use the **no** form of this command.

description *comment*
no description

Syntax Description

comment	Comment or a description applied to the interface. The maximum number of characters is 1022.

Command Default

No description is configured.

Command Modes

Interface configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **description** command to add a description to an interface configuration. The maximum number of characters is 1022.

Task ID

Task ID	Operations
interface	read, write

The following example shows how to add a description to an interface configuration. In this example, the **description** command names a Management Ethernet interface.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface mgmteth 0/

1/CPU0/0
RP/0/RP0/CPU0:router(config-if)# description Management Ethernet Interface
```

do

To execute an EXEC mode command from a configuration mode, use the **do** command in any configuration mode.

do *exec-command*

Syntax Description

exec-command	EXEC mode command to be executed.

Command Default

None

Command Modes

Any configuration mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To display the various mode commands that are available to execute with the **do** command, use the online help (?) function at the configuration mode prompt.

Note The **configure** and **describe** commands are not supported with the **do** command.

Task ID

Task ID	Operations
Task ID for the EXEC command that you are using	read

The following example shows how to execute an EXEC command from interface configuration mode. In this example, the **do** command displays output from the **show protocols** command within interface configuration mode:

```
RP/0/RP0/CPU0:router(config)# interface tengige 0/1/0/1
RP/0/RP0/CPU0:router(config-if)# do show protocols

Routing Protocol "BGP 1"

Address Family IPv4 Unicast:
  Distance: external 20 internal 200 local 200
```

end

To terminate a configuration session and return directly to XR EXEC mode , use the **end** command in any configuration mode.

end

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Any configuration mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **end** command to exit any configuration mode and return directly to XR EXEC mode . If you enter this command without committing the changes to the target configuration, you are prompted to do so:

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to XR EXEC mode .

  If errors are found in the running configuration, the configuration session does not end. To view the errors, enter the **show configuration** (config) command with the **failed** keyword.

- Entering **no** exits the configuration session and returns the router to XR EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

Note Entering **Ctrl-Z** is functionally equivalent to entering the **end** command.

Use the **abort** command to exit the configuration session and return to XR EXEC mode without being prompted to commit changes and without saving changes to the target configuration.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to use the **end** command to end a configuration session. Changes stored in the target configuration are committed by answering **yes**.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige 0/2/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.1.1.1 255.0.0.0
```

```
RP/0/RP0/CPU0:router(config-if)# end

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: yes
RP/0/RP0/CPU0:router#
```

end-group

To exit from configuration group submode and return to global configuration mode, use the end-group command in group configuration mode.

end-group

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Group configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

After you have included all configuration statements that you want in a particular configuration group, use the **end-group** command to exit group configuration mode.

Task ID

Task ID	Operation
config-services	read, write

This example shows how to complete the configuration of a configuration group and exit group configuration mode:

```
RP/0/RP0/CPU0:router(config)# group g-int-gige
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1514
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
RP/0/RP0/CPU0:router(config)#
```

end-template

To exit template configuration mode and return to XR Config mode, use the **end-template** command in template configuration mode.

end-template

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

Template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **end-template** command to exit template configuration mode after you have completed the template definition.

To define a template, use the **template** command. To apply a template to the target configuration, use the **apply-template** command. To view the contents of a template, use the **show running-config** command with the optional **template** *template-name* keyword and argument.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to enter template configuration mode, define a template named "hostname-template" and then exit from template configuration mode:

```
RP/0/RP0/CPU0:router(config)# template hostname-template
RP/0/RP0/CPU0:router(config-TPL)# hostname router-cs1
RP/0/RP0/CPU0:router(config-TPL)# end-template
RP/0/RP0/CPU0:router(config)#
```

exclude-group

To exclude (or override) a configuration group (or groups) to be inherited by the router configuration, use the **exclude-group** command in the appropriate configuration mode. To delete the set exclusion, use the **no** form of this command.

exclude-group *group-name*

Syntax Description

group-name	Configuration group name that needs to be excluded.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

More than one configuration group can be excluded simultaneously. A maximum of eight groups can be specified at one time.

Note From Release 6.3.1 onwards, you can enter Flexible CLI config groups, **apply-group** and **exclude-group** command in any order as long as the entire commit has all the group definitions needed.

Task ID

Task ID	Operation
config-services	read, write

Example

This example shows how to delete the group G_interface using the **exclude-group** command:

```
RP/0/RP0/CPU0:router (config) # exclude-group G_interface

exclude-group G_INTERFACE
ipv4 address 12.21.50.100 255.255.0.0
!
interface GigabitEthernet0/0/0/1
ipv4 address 12.21.51.100 255.255.0.0
```

exit

To close an active terminal session and log off the router, use the **exit** command in XR EXEC mode .

To return the router to the next higher configuration mode, use the **exit** command in any configuration mode.

exit

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Any configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To log off from a terminal session, enter the **exit** command in XR EXEC mode .

When exiting from global or administration configuration mode to XR EXEC mode , you are prompted to commit any uncommitted configuration changes.

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?[cancel]:
```

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to XR EXEC mode .

  If errors are found in the running configuration, the configuration session does not end. To view the errors, enter the **show configuration** (config) command with the **failed** keyword.
- Entering **no** exits the configuration session and returns the router to XR EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

Note Entering the **exit** command from global configuration is functionally equivalent to entering the **end** command.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to return the router to the next higher command mode. In this example, the **exit** command exits from interface configuration mode and returns to global configuration mode. The **exit** command is entered a second time to exit from global configuration mode and return

to XR EXEC mode. Because the configuration has not been committed explicitly (with the **commit** command), the system prompts to commit the configuration changes made during the session.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige 0/2/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.1.1.1 255.0.0.0
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# exit
Uncommitted changes found, commit them before exiting(yes/no/cancel)?[cancel]: yes
```

The following example shows how to use the **exit** command from XR EXEC mode to log off from a terminal session:

```
RP/0/RP0/CPU0:router# exit

router con0_RP1_CPU0 is now available

Press RETURN to get started.
```

hostname

To specify or modify the hostname for the router, use the **hostname** command in XR Config mode.

hostname *name*

Syntax Description

name	New hostname for the router.

Command Default

The factory-assigned default hostname is “ios.”

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The hostname is used in prompts and default configuration filenames.

No blank or space characters are permitted as part of a name. Do not expect case to be preserved. Uppercase and lowercase characters look the same to many Internet software applications. It may seem appropriate to capitalize a name the same way you might do in English, but conventions dictate that computer names appear all lowercase. For more information, see RFC 1178, *Choosing a Name for Your Computer*.

Task ID

Task ID	Operations
root-lr	read, write

The following example shows how to change the router hostname:

```
RP/0/RP0/CPU0:router(config)# hostname router1
```

load

To populate the target configuration with the contents of a previously saved configuration file, use the **load** command in global configuration or administration configuration mode.

load *device*:*directory-path*

Syntax Description

device: directory-path	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

If the full path of the file is not specified, the present working directory is used.

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 25.4.1	The ftp and tftp keywords are deprecated.
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **load** command to populate the target configuration with the contents of a previously saved configuration. When loading a file, you must specify the device, directory path, and filename of the configuration file.

Use the **commit** command in conjunction with the **load** command. Load a new configuration with the **load** command, and use the **commit** command with the **replace** keyword to have the loaded configuration become the active (running) configuration.

Use the **show configuration failed** (config) command with the optional **load** keyword to display syntax errors that occurred during the last load operation.

Starting with Cisco IOS XR Software Release 25.4.1, do not use the FTP or TFTP protocol for file transfers, as this is unsecure and deprecated. Instead, use SFTP (Secure File Transfer Protocol) or SCP (Secure Copy Protocol) for improved security, as they provide encrypted communication.

Syslog warnings will be generated, when the command is configured.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to load a target configuration file into the current configuration session. The current configuration session is then populated with the contents of the file.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# load disk1:myconfig.cfg
RP/0/RP0/CPU0:router(config)# show config
```

```
Building configuration...
interface TenGigE 0/3/0/0
 description My 10 GE Interface
 ipv4 address 10.10.11.20 255.0.0.0
!
end
```

load commit changes

To populate the target configuration with changes from previous configuration commits, use the **load commit changes** command in global configuration or administration configuration mode.

load commit changes {*commit-id* | **since** *commit-id* | **last** *number-of-commits*}

Syntax Description

commit-id	Specific configuration commit.
since *commit-id*	Loads all configuration changes committed into the target buffer since (and including) a specific configuration commit, *commit-id*.
last *number-of-commits*	Loads the configuration changes into the target buffer that have been made during the last number of configuration commits specified with the *number-of-commits* argument.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **load commit changes** command to populate the target configuration with changes from previous configuration commits. The changes are not applied until you enter the **commit** command.

Use the **show configuration** (config) command to display the target configuration.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to populate the target configuration with changes from a previous configuration commit:

```
RP/0/RP0/CPU0:router(config)# load commit changes since 1000000006

Building configuration...
Loading.
223 bytes parsed in 1 sec (222)bytes/sec
```

load configuration failed

To populate the target configuration with the contents of the previous failed configuration commit, use the **load configuration failed** command in global configuration or administration configuration mode.

load configuration failed {**commit** | **startup** [**previous** *number-of-reloads*] [**noerror**]}

Syntax Description

commit	Loads the failed configuration from the last commit.
startup	Loads the failed configuration from the startup configuration.
previous *number-of-reloads*	(Optional) Loads the failed configurations from a previous router reload. Valid *number-of-reloads* values are 1 to 4.
noerror	(Optional) Excludes the error reasons when the failed configurations are loaded.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **load configuration failed** command to populate the target configuration with the contents of the previous failed configuration commit.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to populate the target configuration with the contents of the previous failed configuration commit:

```
RP/0/RP0/CPU0:router(config)# load configuration failed startup
Loading.
32 bytes parsed in 1 sec (31)bytes/sec
```

load configuration removed

To populate the target configuration with the contents of the previous removed configuration, use the **load configuration removed** command in global configuration or administration configuration mode.

load configuration removed *config-id*

Syntax Description

config-id	Identifier of the removed configuration to load.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **load configuration removed** command to populate the target configuration with the contents of the removed configuration during installation operations.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to populate the target configuration with the contents of the removed configuration during installation:

```
RP/0/RP0/CPU0:router(config)# load configuration removed 20070316021626.cfg
```

load rollback changes

To populate the target configuration with the contents of a previous configuration, use the **load rollback changes** command in global configuration or administration configuration mode.

load rollback changes {*commit-id* | **last** *number-of-commits* | **to** *commit-id*}

Syntax Description

commit-id	Rolls back the configuration changes for a specific configuration commit.
last *number-of-commits*	Rolls back to the configuration that existed before the last number of commits (specified with the *number-of-commits* argument) were made.
to *commit-id*	Rolls back to the running configuration that existed before the configuration specified with the *commit-id* argument.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **load rollback changes** command to load rollback configuration changes to the target configuration. This command is similar to the **rollback configuration** command. The difference between the commands is that the **load rollback changes** command copies the rollback changes to the target configuration and does not commit the changes until the changes are explicitly committed with the **commit** command.

Use the **show configuration rollback changes** command to display rollback changes.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to populate the target configuration with the contents of a previous configuration:

```
RP/0/RP0/CPU0:router(config)# load rollback changes 1000000004

Building configuration...
Loading.
302 bytes parsed in 1 sec (301)bytes/sec
```

man

Cisco IOS XR software provides online help for standard command-line interface (CLI) commands using manual (man) pages. To display manual pages, use the **man** command in EXEC mode.

man {**command** *command-name* | **feature** [*feature-name*] | **keyword** *keywords*}

Syntax Description

command *command-name*	Displays the manual pages for a specific command. The *command-name* argument must include the complete command name.
feature [*feature-name*]	Displays all commands available in the feature. Use the **man** command with the **feature** keyword to list the available feature names.
keyword *keywords*	Displays a list of command names that match the keywords. Enter one or more keywords to match in a command. When entering multiple keywords, the keywords must be entered in the same sequential order as they are in the command.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must have the documentation PIE installed before you can use the **man** command. If you attempt to run this command without the documentation PIE installed, an error is displayed as shown in the following example:

```
RP/0/RP0/CPU0:router# man command show install

Building index table...
Warning. Unable to get directory info for '/pkg/man' :No such file or directory.
 Discarding!
man [5521656]:Building index table failed. No entries found
```

For information about installing optional software PIEs, see the *Upgrading and Managing Cisco IOS XR Software* module in *System Management Configuration Guide for Cisco 8000 Series Routers*.

Use the **man** command to display the manual pages for a specific command on the basis of the command name, a feature, or a keyword. Each man page contains the command name, syntax, command mode, usage, examples, and related commands.

The **man** command queries and displays command information about the router. A query can be based on keywords or a feature. The **feature** *feature-name* keyword and argument display all commands that match the feature. For example, entering **man feature - base-1** displays all commands that match the -base-1 feature. The **keyword** *keywords* keyword and argument display all commands that contain the specified keyword. For example, **man keyword ipv4** displays all commands that contain ipv4.

Task ID

Task ID	Operations
basic-services	read

The following example shows how to display the manual page for the **arp timeout** command:

```
RP/0/RP0/CPU0:router# man command arp timeout

COMMAND
arp timeout

DESCRIPTION


To specify how long dynamic entries learned on an interface remain in the
Address Resolution Protocol (ARP) cache, use the arp timeout command in
interface configuration mode. To remove the arp timeout command from the
configuration file and restore the system to its default condition with
respect to this command, use the no form of this command.

arp timeout seconds

no arp timeout<seconds>

SYNTAX DESCRIPTION

seconds
Time, in seconds, for which an entry remains in the ARP cache. The
range is from 0 to 4294967. A value of 0 means that entries are never
cleared from the cache. The default is 14400.

DEFAULTS

Entries remain in the ARP cache for 14400 seconds (4 hours).

COMMAND MODES

Interface configuration

COMMAND HISTORY

Release
Modification

Release 2.0
This command was introduced.

USAGE GUIDELINES

To use the arp timeout command, you must be a member of a user group
```

```
associated with the cef task ID.

For detailed information about user groups and task IDs, refer to the
Configuring AAA Services on Cisco IOS-XR Software module of the Cisco IOS-XR
System Security Configuration Guide.

This command is ignored when issued on interfaces that do not use ARP. Also,
ARP entries that correspond to the local interface or that are statically
configured by the user never time out.

The show interfaces command displays the ARP timeout value in
hours:minutes:seconds, as follows:

* * * * * * * * * * * * * * * * START OF LISTING * * * * * * * * * * * * * * * *

ARP type: ARPA, ARP Timeout 04:00:00
* * * * * * * * * * * * * * * * END OF LISTING * * * * * * * * * * * * * * * *

EXAMPLES

The following example shows how to set the ARP timeout to 3600 seconds to
allow entries to time out more quickly than the default:

* * * * * * * * * * * * * * * * START OF LISTING * * * * * * * * * * * * * * * *
RP/0/RP0/CPU0:router# configure

RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP1/CPU0/0

RP/0/RP0/CPU0:router(config-if)# arp timeout 3600
* * * * * * * * * * * * * * * * END OF LISTING * * * * * * * * * * * * * * * *

RELATED COMMANDS

Command
Description

clear arp-cache
Deletes all dynamic entries from the ARP cache.

show arp (cache)
Displays the entries in the ARP table.

show interfaces
Displays statistics for all interfaces configured on the networking
device.
```

more

To display the contents of a file, use the **more** command in EXEC or administration EXEC mode.

more [**/ascii** | **/binary** | **/ebcdic**] *filesystem*:*directory-path* **location** [*node-id* | **all**] {| **begin** *regular-expression* | | **exclude** *regular-expression* | | **include** *regular-expression*}

Syntax Description

/ascii	(Optional) Displays a binary file in ASCII format.
/binary	(Optional) Displays a file in hexadecimal or text format.
/ebcdic	(Optional) Displays a binary file in ebcdic format.
filesystem:directory-path	File system location of the file to be displayed. Include the file system alias for the *filesystem* argument, followed by a colon, and the directory path of the file to be displayed.
location [*node-id* | **all**]	(Optional) Displays the contents of a file on a designated node or all nodes.
regular-expression	(Optional) Regular expression found in the file.
|	Vertical bar (the "pipe" symbol) indicates that an output processing specification follows.
begin	(Optional) Begins unfiltered output of the **more** command with the first line that contains the regular expression.
exclude	(Optional) Displays output lines that do not contain the regular expression.
include	(Optional) Displays output lines that contain the regular expression.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **more** command to display any text file, especially an ASCII file stored on the router or accessible through the network. The file can be a configuration file or any other text file.

Filtering Output

This table shows filter options for the output displayed by the **more** command.

Table 11: Filtering Options

Command	Purpose
more *filesystem:* **| begin** *regular-expression*	Begins unfiltered output of the **more** command with the first line that contains the regular expression.
more *filesystem:* **| exclude** *regular-expression*	Displays output lines that do not contain the regular expression.
more *filesystem:* **| include** *regular-expression*	Displays output lines that contain the regular expression.

Adding a Filter at the --More-- Prompt

You can also specify a filter at the --More-- prompt of a **more** command output. To filter output from the --More-- prompt, enter a forward slash (**/**) followed by a regular expression. The filter remains active until the command output finishes or is interrupted (using **Ctrl-Z** or **Ctrl-C**).

- A second filter cannot be specified at a --More-- prompt if a filter has already been specified at the original command or at a previous --More-- prompt.
- The minus sign (–) preceding a regular expression displays output lines that do not contain the regular expression.
- The plus sign (+) preceding a regular expression displays output lines that contain the regular expression.



Note After you specify a filter for a **more** command, you cannot specify another filter at the next --More-- prompt. The first specified filter remains until the **more begin** command output finishes or until you interrupt the output. The use of the keyword does not constitute a filter.

Task ID

Task ID	Operations
filesystem	execute

The following example shows partial sample output from the **more** command. The output displays a configuration file saved on the hard disk drive.

```
# more harddisk:/user/alternate.cfg

!! Last configuration change at 15:52:55 UTC Fri Feb 13 2009 by UNKNOWN
!
line console
exec-timeout 0 0
!
interface MgmtEth0/RP1/CPU0/0
 ipv4 address 10.32.45.154 255.0.0.0
 !
interface TenGigE0/1/0/0
 ipv4 address 10.32.45.155 255.0.0.0
 keepalive disable
```

```
 !
interface TenGigE0/1/0/1
 ipv4 address 10.32.45.156 255.0.0.0
 keepalive disable
 !
interface TenGigE0/1/0/2
/ip
 ipv4 address 10.32.45.157 255.0.0.0
 keepalive disable
 !
interface TenGigE0/1/0/3
 ipv4 address 10.32.45.158 255.0.0.0
 keepalive disable
 !
interface TenGigE0/2/0/0
 ipv4 address 10.32.45.159 255.0.0.0
 keepalive disable
 !
 --More--
```

The following example shows partial sample output from the **more** command. The output begins with unfiltered output from the first line that contains the regular expression “ipv4.” In this example, a new search is specified that begins with output lines that contain the regular expression “ipv4.”

```
RP/0/RP0/CPU0:router# more disk0:config.backup | begin ipv4

ipv4 address 2.2.2.2 255.255.255.255
!
interface TenGigE0/3/1/0
 shutdown
!
interface TenGigE0/3/1/2
 shutdown
!
interface TenGigE0/2/1/0
 ipv4 address 1.1.1.1 255.255.255.0
 keepalive disable
!
interface TenGigE0/2/1/1
  ipv4 address 1.1.1.1 255.255.255.0
  keepalive disable
!
interface TenGigE0/2/1/2
  ipv4 address 1.1.1.1 255.255.255.0
  keepalive disable
!
interface TenGigE0/2/1/3
 shutdown
!
  /ipv4

filtering...
 ipv4 address 1.1.1.1 255.255.255.0
 proxy-arp disable
 shutdown
!
interface TenGigE 0/1/0/0
 ipv4 address 1.1.1.1 255.255.255.0
 proxy-arp disable
!
route ipv4 0.0.0.0/0 12.25.26.5
route ipv4 223.255.254.254/32 12.25.0.1
```

```
end
```

The following example shows partial sample output of the **more** command on the sample file config.backup in disk0:. The command usage is `more disk0:config.backup | include log`. At the --More-- prompt, a new search is specified that begins with output lines that contain the regular expression “aaa.”

```
RP/0/RP0/CPU0:router# more disk0:config.backup | include log

logging trap
logging trap informational
logging console debugging
logging history size 1
.
.
.
/aaa

filtering...
aaa authentication login default none
```

The following example shows partial sample output from the **more** command. The output excludes lines that contain the regular expression “alias.” In this example, at the --More-- prompt, a new search is specified, beginning with output lines that contain the regular expression “ipv4 address.”

```
RP/0/RP0/CPU0:router# more disk0:myconfig/file | exclude alias

Building configuration...
!! Last configuration change at 18:17:00 UTC Thu May 16 2009 by lab
!
hostname router
line console
 exec-timeout 0 0
 width 132
 length 0
 session-timeout 0
/ipv4 address

filtering...
ipv4 address 10.10.1.1 255.255.255.255
!
interface Loopback200
 ipv4 address 10.20.1.1 255.255.255.255
!
interface TenGigE0/0/0/0
 ipv4 address 10.30.1.1 255.255.0.0
 keepalive 100
!
interface preconfigure TenGigE0/1/0/1
 shutdown
end
```

pwd (config)

To display the current configuration submode from a configuration submode, use the **pwd** command in any supported configuration submode.

pwd

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Any subconfiguration mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to use the **pwd** command from an interface configuration submode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige 0/6/4/5
RP/0/RP0/CPU0:router(config-if)# pwd

interface TenGigE0/6/4/5
RP/0/RP0/CPU0:router(config-if)#
```

rollback configuration

To roll back the running configuration to a previous configuration, use the **rollback configuration** command inEXEC or administration EXEC mode.

rollback configuration {**last** *number-of-commits* | **to** *commit-id*} {**best-effort** | **force**} [**label** *label*] **comment** *comment*

Syntax Description

last *number-of-commits*	Rolls back to the configuration that existed before the last number of commits (specified with the *number-of-commits* argument) were made.
to *commit-id*	Rolls back to the running configuration that existed before the configuration specified with the *commit-id* argument.
best-effort	Rolls back to the configuration that existed before the last n commits, and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.
force	(Optional) Specifies to override any commit blocks.
label *label*	(Optional) Assigns a text label to this rollback. The *label* argument must begin with a letter.
comment *comment*	(Optional) Assigns a text comment to this rollback. The *comment* argument can be up to 60 characters long.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Each time the **commit** command is entered, a commit ID is assigned to the new configuration. You can revert the system to the configuration of a previous commit ID with the **rollback configuration** command:

- Use the **to** keyword to revert to the configuration that existed *before* the configuration specified with the *commit-id* argument.
- Use the **last** keyword to revert to the configuration that existed *before* the last number of configuration commits (specified with the *number-of-commits* argument) were made.
- Use **show configuration commit list** to display a list of the commit IDs available for rollback operations.

Note The most recent 100 commits are retained by the system. As new commit IDs are added, the oldest commit IDs are discarded and are no longer available for rollback operations.

Use the **force** keyword to override commits that would fail otherwise. This is useful in the event of a low-memory condition on the router, to revert to a commit that would remove a configuration that caused the low-memory condition.

Note The rollback operation may fail if you try to rollback two (or more) commits where the individual commits involve the configuration and removing of the configuration of the same item, and there is a dependency of one item over another in any of the individual commit operations.

Task ID

Task ID	Operations
root-lr (EXEC)	read, write
root-system (administration EXEC)	read, write

Rolling Back to a Specific Commit ID

The following example shows how to roll back to a specific commit ID. In this example, the **show configuration commit list** command displays the available rollback points. The configuration is then rolled back to a prior commit with the **rollback configuration** command.

```
RP/0/RP0/CPU0:router# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000009  lab       con0_0_C  Rollback    02:41:08 UTC Sun Sep 26 2009
2    1000000008  lab       con0_0_C  CLI         02:40:30 UTC Sun Sep 26 2009
3    1000000007  lab       con0_0_C  CLI         02:39:54 UTC Sun Sep 26 2009
4    1000000006  lab       con0_0_C  Rollback    02:38:40 UTC Sun Sep 26 2009
5    1000000005  lab       con0_0_C  CLI         02:37:35 UTC Sun Sep 26 2009
6    1000000004  lab       con0_0_C  CLI         02:37:04 UTC Sun Sep 26 2009

RP/0/RP0/CPU0:router# rollback configuration to 1000000008

Loading Rollback Changes.
Loaded Rollback Changes in 1 sec
Committing.
1 items committed in 1 sec (0)items/sec
Updating.RP/0/RP0/CPU0:Sep 26 02:42:09.318 : config_rollback[65707]: %LIBTARCFG-
6-COMMIT : Configuration committed by user 'lab'.   Use 'show commit changes 100
0000010' to view the changes.
```

```
Updated Commit database in 1 sec
Configuration successfully rolled back to '1000000008'.
```

Rolling Back to a Span of Configuration Commits

The following example shows how to roll back to the configuration that existed prior to the last two configuration commits:

```
RP/0/RP0/CPU0:router# rollback configuration last 2

Loading Rollback Changes.
Loaded Rollback Changes in 1 sec
Committing.
1 items committed in 1 sec (0)items/sec
Updating.
Updated Commit database in 1 sec
Configuration successfully rolled back 2 commits.
```

root

To return to configuration mode from a configuration submode, use the **root** command in any supported configuration submode.

root

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Any subconfiguration mode except the following:

- The **root** command is not available under the route-policy submodes, because it requires the **end-policy** command to exit out of the configuration.
- The **root** command is not available in template submode, but is available in the submodes configurable under the template submode.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read

The following example shows how to use the **root** command to return to configuration mode from the interface configuration submode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige 0/1/0/0
RP/0/RP0/CPU0:router(config-if)# root
RP/0/RP0/CPU0:router(config)#
```

The following example shows how to use the **root** command from a submode configurable under the template submode. In this example, the **root** command is used to return to configuration mode from the username submode:

Note The recommended range for a user-defined username is 2-253 characters.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# template test
RP/0/RP0/CPU0:router(config-TPL)# username xyz
RP/0/RP0/CPU0:router(config-un)# root
RP/0/RP0/CPU0:router(config)# show conf
```

```
Building configuration...
template test
 username xyz
 !
 end-template
 end
```



Tip The **root** command is not available from the template submode, but is available in the submodes configurable under the template submode.

save configuration

To save the contents of a configuration to a file, use the **save configuration** command in global configuration or administration configuration mode.

save configuration [**running**] *device***:***directory-path*

Syntax Description

running	(Optional) Saves the contents of the running configuration.
device: directory-path	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To save a configuration to a file, use the **save configuration** command.

To save a configuration that failed to a file, use the **save configuration failed** command.

Task ID

Task ID	Operations
config-services	read

The following example shows the configuration saved to disk0: from global configuration mode:

```
RP/0/RP0/CPU0:router(config)# save configuration disk0:sample3

Destination file name (control-c to abort): [/sample3]?
Building configuration.
1 lines built in 1 second
[OK]
```

The following example shows the configuration saved to disk1 from administration EXEC mode:

```
RP/0/RP0/CPU0:router(admin-config)# save configuration disk1:sample4

Destination file name (control-c to abort): [/sample4]?
Building configuration.
1 lines built in 1 second
[OK]
```

save configuration changes

To save the changes of a configuration to a file, use the **save configuration changes** command in global configuration or administration configuration mode.

save configuration changes *device***:***directory-path*

Syntax Description

device: directory-path	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To save the configuration changes to be made during a replace operation to a file, use the **save configuration changes** command.

Task ID

Task ID	Operations
config-services	read

The following example shows the configuration saved to disk0: from global configuration mode:

```
RP/0/RP0/CPU0:router(config)# save configuration changes disk0:sample3

Destination file name (control-c to abort): [/sample3]?
Building configuration.
1 lines built in 1 second
[OK]
```

save configuration commit changes

To save the changes for a commit, or a series of commits, to a file, use the **save configuration commit changes** command in global configuration or administration configuration mode.

save configuration commit changes {*commit-id* | **last** *number-of-commits* | **since** *commit-id*} *device***:***directory-path*

Syntax Description

commit-id	Specific commit ID.
last *number-of-commits*	Saves changes made in the most recent *number-of-commits*.
since *commit-id*	Saves changes made since (and including) a specific *commit-id*.
device: directory-path	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **save configuration commit changes** command to save the changes made in a commit operation to a file. You can specify a specific commit ID, all the changes since a specified commit ID, or the changes that occurred during the last *n* commits.

Task ID

Task ID	Operations
config-services	read

The following example saves the changes from the last two commit operations to disk0:

```
RP/0/RP0/CPU0:router# save configuration commit changes last 2 disk0:sample1

Destination file name (control-c to abort): [/sample1]?
Building configuration.
5 lines built in 1 second
[OK]
```

save configuration failed

To save the contents of the failed configuration, use the **save configuration failed** command inglobal configuration or administration configuration mode.

save configuration failed [**load**|**noerrors**|**startup** [**previous** *number*] [**noerror**]] *device***:***directory-path*

Syntax Description

load	(Optional) Saves the failed configuration (syntax errors) in the last reload.
noerrors	(Optional) Excludes the error reasons from the saved configuration.
startup	(Optional) Saves the failed configuration during startup.
previous *number*	(Optional) Saves a failed startup configuration from the specified previous sessions. The *number* argument is a value between 1 and 4 that indicates how many failed startup configurations to save.
device: directory-path	Storage device and directory path of the configuration file to be saved.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To save a configuration to a file, use the **save configuration** command.

To save a configuration that failed to a file, use the**save configuration failed** command.

To save a configuration that failed during startup to a file, use the **save configuration failed** command with the **startup** keyword.

Task ID

Task ID	Operations
config-services	read

The following example saves the failed configuration to disk0:

```
RP/0/RP0/CPU0:router# save configuration failed disk1:/configs
```

save configuration merge

To save the contents of a merged configuration to a file, use the **save configuration merge** command in global configuration or administration configuration mode.

save configuration merge *device***:***directory-path*

Syntax Description		
	device **:** *directory-path*	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read

The following example shows the configuration saved to disk0:

```
RP/0/RP0/CPU0:router# save configuration merge disk0:sample3

Destination file name (control-c to abort): [/sample3]?
Building configuration.
1 lines built in 1 second
[OK]
```

save configuration removed

To save the contents of a removed configuration to a file, use the **save configuration removed** command in global configuration or administration configuration mode.

save configuration removed *removed-configuration-file device*:*directory-path*

Syntax Description

removed-configuration-file	Specifies the name of the removed configuration file.
device:*directory-path*	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When a package is deactivated, the configuration belonging to that package is removed from the running configuration and saved to a file. To save a copy of the removed configuration file, use the **save configuration removed** command.

Task ID

Task ID	Operations
config-services	read

To view a list of the available removed configuration files, use the **save configuration removed** command followed by a question mark:

```
RP/0/RP0/CPU0:router(config)# save configuration removed ?

    20051208042507.cfg  Removed configuration.
    20051208044553.cfg  Removed configuration.
    <cr>
```

In the following example, a removed configuration is saved to disk0: and assigned the filename "sample3:"

```
RP/0/RP0/CPU0:router(config)# save configuration removed 20051208042507.cfg disk0:sample3

  Destination file name (control-c to abort): [/sample3]?
  Building configuration.
  1 lines built in 1 second
  [OK]
```

save rollback changes

To save the rollback changes, use the **save rollback changes** command in global configuration or administration configuration mode.

save rollback changes {*commit-id* | **last** *number-of-commits* | **to commit-id**} *device***:***directory-path*

Syntax Description

commit-id	Specific commit ID.
last *number-of-commits*	Saves the rollback changes for the last *n* commits
to *commit-id*	Saves rollback changes up to a specific *commit-id*.
device: directory-path	Storage device and directory path of the configuration file to be loaded into the target configuration.

Command Default

None

Command Modes

Global configuration

Administration configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **save rollback changes** command to save the changes that would be made in a configuration rollback to a specific commit point or for a series of commits.

Task ID

Task ID	Operations
config-services	read

The following example shows that the rollback changes for the commit point 5 are saved to the file sample4 on disk0:

```
RP/0/RP0/CPU0:router# save rollback changes last 1 disk0:sample4

Destination file name (control-c to abort): [/sample4]?
Building configuration.
6 lines built in 1 second
[OK]
```

service cli commit-optimized enable

To prevent the re-application of the commands which are already present in the running configuration of the router, use the **service cli commit-optimized enable** command in XR Config mode.

service cli commit-optimized enable

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

None

Example

```
Router# configure
Router(config)# service cli commit-optimized enable
Router(config)# commit
```

set default-afi

To set the default address family identifier (AFI) for the current session, use the **set default-afi** command in EXEC mode.

set default-afi {**all** | **ipv4** | **ipv6**}

Syntax Description

all	Sets the default AFI to IPv4 and IPv6 for the current session.
ipv4	Sets the default AFI to IPv4 for the current session. This is the default setting.
ipv6	Sets the default AFI to IPv6 for the current session.

Command Default

The default AFI setting is set to IPv4 for all sessions.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **set default-afi** command to set the default AFI for the current session. This command acts as a keystroke shortcut for **show** commands. If the default AFI setting is set to IPv4, then you would not have to specify the **ipv4** keyword for **show** commands that support the **ipv4** keyword. For example, if the AFI setting is set to IPv4, you could issue the **show route** command without specifying the **ipv4** keyword to display IPv4 routes in the Routing Information Base (RIB).

Use the **show default-afi-safi-vrf** command to display the default AFI setting.

Task ID

Task ID	Operations
basic-services	read, write

The following example shows how to set the default AFI to IPv6:

```
RP/0/RP0/CPU0:router# set default-afi ipv6

%% Default Address Family Identifier is set to 'ipv6'
```

set default-safi

To set the default subaddress family identifier (SAFI) for the current session, use the **set default-safi** command in EXEC mode.

set default-safi {**all** | **multicast** | **unicast**}

Syntax Description

all	Sets the default SAFI to multicast and unicast for the current session.
multicast	Sets the default SAFI to multicast for the current session.
unicast	Sets the default SAFI to unicast for the current session. This is the default setting.

Command Default

The default SAFI setting is set to unicast for all sessions.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **set default-safi** command to set the default SAFI setting for the current session. This command acts as a keystroke shortcut for **show** commands. If the default SAFI setting is set to unicast, you would not have to specify the **unicast** keyword for **show** commands that support that keyword. For example, if the default SAFI setting is set to unicast, you could issue the **show router** command without specifying the **unicast** keyword to display information about unicast address prefixes in the Routing Information Base (RIB).

Use the **show default-afi-safi-vrf** command to display the default SAFI setting.

Task ID

Task ID	Operations
basic-services	read, write

The following example shows how to set the default SAFI to multicast:

```
Router# set default-safi multicast

%% Default Sub-Address Family Identifier is set to 'multicast'
```

set default-vrf

To set the default VPN routing and forwarding (VRF) instance for the current session, use the **set default-vrf** command in EXEC mode.

set default-vrf {*name* | **none**}

Syntax Description

name	Default VPN routing and forwarding name.
none	Sets the default VPN routing and forwarding name to empty.

Command Default

The default VRF setting is set to empty.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **set default-vrf** command to set the default VRF setting for the current session. This command acts as a keystroke shortcut for **show** commands. For example, if the default VRF is configured, you can issue the **show route** command without specifying the VRF name.

When the default VRF for the session is set to **none**, then IPv4 routes for the system default VRF are displayed.

Note To override the default VRF setting, specify the VRF name in the **show** command.

Use the **show default-afi-safi-vrf** command to display the default VRF setting.

In the following example, the default VRF is set to “dft_vrf:”

```
Router# set default-vrf dft_vrf

  %% Default Virtual Routing/Forwarding is set to 'dft_vrf'
```

In the following command, the **show route** command is entered without specifying a VRF name. The results for the "dft_vrf" VRF are displayed because the default VRF was set to “dft_vrf.”

```
Router# show route ipv4

  % No matching vrf found
```

When the default VRF for the session is set to **none**, the system default VRF routes are displayed. In the following example, the default VRF is set to (empty) and the **show route** command displays the system default VRF information:

```
Router# set default-vrf none
```

```
%% Default Virtual Routing/Forwarding is set to ''

Router# show route ipv4

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local

Gateway of last resort is 12.29.0.1 to network 0.0.0.0

  S*   0.0.0.0/0 [1/0] via 12.29.0.1, 00:31:30
  L    10.10.10.10/32 is directly connected, 3d02h, Loopback1
  C    12.29.0.0/16 is directly connected, 00:31:30, MgmtEth0/0/CPU0/0
  L    12.29.56.21/32 is directly connected, 00:31:30, MgmtEth0/0/CPU0/0
```

show

To display information about the system configuration or operational state, use the **show** command in EXEC mode, administration EXEC mode, or any configuration mode.

show *command*[| **begin** *regular-expression* | | **exclude** *regular-expression* | | **file** *filesystem:* | | **include** *regular-expression*]

Syntax Description

command	Supported **show** command.
|	Vertical bar (the "pipe" symbol) indicates that an output processing specification follows.
regular-expression	(Optional) Regular expression found in **show** command output.
begin	(Optional) Begins unfiltered output of the **show** command with the first line that contains the regular expression.
exclude	(Optional) Displays output lines that do not contain the regular expression.
file *filesystem:*	(Optional) Writes the output lines that contain the regular expression to the specified file on the specified file system. Include the file system alias for the *filesystem* argument, followed by a colon, and the directory path and filename.
include	(Optional) Displays output lines that contain the regular expression.

Command Default

None

Command Modes

EXEC

Administration EXEC

Any configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show** commands display information about the system and its configuration. To display a list of the available **show** commands, use the question mark (?) online help function.

Filtering Output

Search options for the **show** command are shown in this table.

Table 12: Show Command Search Options

Command	Purpose
show *command* | **begin** *regular-expression*	Begins unfiltered output of the **show** command command with the first line that contains the regular expression.

Command	Purpose
show *command* | **exclude** *regular-expression*	Displays output lines that do not contain the regular expression.
show *command* | **include** *regular-expression*	Displays output lines that contain the regular expression.
show *command* | **file** *filesystem:*	Writes the output lines that contain the regular expression to the specified file on the specified file system.

Adding a Filter at the --More-- Prompt

You can also specify a filter at the --More-- prompt of a **show** command output. To filter output from the --More-- prompt, enter a forward slash (/) followed by a regular expression. The filter remains active until the command output finishes or is interrupted (using **Ctrl-Z** or **Ctrl-C**).

- If a filter is specified at the original command or a previous --More-- prompt, a second filter cannot be applied.
- The use of the **begin** keyword does not constitute a filter.
- The minus sign (–) preceding a regular expression displays output lines that do not contain the regular expression.
- The plus sign (+) preceding a regular expression displays output lines that contain the regular expression.

Task ID

Task ID	Operations
Task ID for the feature used with the **show** command	read

For example, the **show interfaces** command requires read privileges in the interface task ID.

The following example shows output from the **show interface | include protocol** command. In this example, the **show** command command includes only lines in which the regular expression “protocol” appears:

```
RP/0/RP0/CPU0:router# show interface | include protocol

  Null0 is up, line protocol is up
  0 drops for unrecognized upper-level protocol
  TenGigE0/2/0/0 is administratively down, line protocol is administratively down
  0 drops for unrecognized upper-level protocol
  TenGigE0/2/0/1 is administratively down, line protocol is administratively down
  0 drops for unrecognized upper-level protocol
  TenGigE0/2/0/2 is administratively down, line protocol is administratively down
  0 drops for unrecognized upper-level protocol
  TenGigE0/2/0/3 is administratively down, line protocol is administratively down
  0 drops for unrecognized upper-level protocol
  FastEthernet0/RP0/CPU0/0 is administratively down, line protocol is administratively
  down
  FastEthernet0/RP0/CPU0/0 is administratively down, line protocol is administratively
  down
  0 drops for unrecognized upper-level protocol
```

On most systems, the **Ctrl-Z** key combination can be entered at any time to interrupt the output and return to EXEC mode. For example, use the **show running-config | begin hostname** command to

start the display of the running configuration file at the line containing the hostname setting, then use **Ctrl-Z** when you get to the end of the information you are interested in.

The following example shows sample output from the **show configuration running | begin line** command. The output begins with unfiltered output from the first line that contains the regular expression “line.” In this example, at the --More-- prompt, a new search is specified that begins with output lines that contain the regular expression “ipv4.”



Note The use of the **begin** keyword does not constitute a filter.

```
RP/0/RP0/CPU0:router# show configuration running | begin line

  Building configuration...
  line console
   exec-timeout 120 120
  !
  logging trap
  --More--
  /ipv4

  filtering...
  route ipv4 0.0.0.0 255.255.0.0 pos0/2/0/0
  interface TenGigE0/2/0/0
   ipv4 address 172.19.73.215 255.255.0.0
  end
```

show aliases

To display all defined aliases or the aliases defined in a specified mode, use the **show aliases** command in EXEC mode.

show aliases

Syntax Description

This command has no keywords or arguments.

Command Default

Displays all aliases currently configured on the system.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show aliases** command to display all aliases currently configured on the system.

Task ID

Task ID	Operations
basic-services	read

The following example illustrates sample output from the **show aliases** command. The output displays a summary of all the command aliases configured.

```
RP/0/RP0/CPU0:router# show aliases

exec mode aliases:
ipv4_brief              show ipv4 interface brief

interface mode aliases:
sample_int              tengige 0/2/0/0
```

show apply-group

To display the applied configuration groups, use the **show apply-group** command in EXEC mode.

show apply group

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The output of this show command indicates if the group is a part of the global apply-group or is a part of the sub-mode level apply-group.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show apply-group** command:

```
RP/0/RP0/CPU0:router #  show apply-group
             Global       Non-Global
Groups Reference Count Reference Count
------ --------------- ---------------
B               1             0
C               1             0
```

Reference count can either be 0 ro 1. 0 indicates that the group is not applied globally; 1 indicates that the group is globally applied.

show running-configuration filter vrf

To filter running configuration specific to vrf, use the **show running configuration filter vrf** command in the Administration EXEC, Administration Configuration, or Global Configuration modes.

show running-configuration filter vrf *vrf-name*

Syntax Description

vrf-name	Specifies the vrf-name.

Command Default

None

Command Modes

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 24.2.11	This command was introduced.

Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Task ID

Task ID	Operations
basic-services	read

This example shows how to filter the the currently running (committed) configuration for a specific VRF.

```
Router#show running-configuration filter vrf vrf80
!! Building configuration...
!! IOS XR Configuration 24.2.11.32M
!! Last configuration change at Mon Jan 15 05:09:20 2024 by cisco
!
vrf vrf80
 address-family ipv4 unicast
  import route-target
   1:80
  !
  export route-target
   1:80
  !
 !
 address-family ipv6 unicast
  import route-target
   1:80
  !
  export route-target
   1:80
```

```
  !
 !
!
 neighbor 192.0.2.1
   remote-as 200
   ebgp-multihop 4
   update-source Loopback90
   address-family ipv4 unicast
    route-policy PASS in
    route-policy PASS out
   !
  !
 !
!
end
```

show configuration failed

To display information about a configuration that failed during the last commit , use the **show configuration failed** command in EXEC mode.

show configuration failed [**inheritance**]

Syntax Description

inheritance	Displays the failed configuration details at the inheritance level.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Without the inheritance keyword, this command displays the failed configuration information in brief.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to run the **show configuration failed** command:

Note When there are two (or more) groups that have failed, the ordering of the failed groups is displayed in the same order as the apply-group statement.

```
RP/0/RP0/CPU0:router (config) #  show config failed
!! SEMANTIC ERRORS: This configuration was rejected by
!! the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.
apply-group GROUP-1 GROUP-2 GROUP-3 GROUP-4 GROUP-5
!% Please issue "show configuration failed inheritance" for details.
Applying following groups failed: GROUP-2 GROUP-4 GROUP-5
```

show configuration changes

To display the configuration changes to be made during a replace operation, use the **show configuration changes** command in global configuration or administration configuration System Admin Config mode .

show configuration changes [**diff**]

Syntax Description

diff	(Optional) Displays the changes in UNIX-like format.

Command Default

None

Command Modes

XR Config mode

System Admin Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read
basic-services	read

The following example shows the changes to be made during a replace operation:

```
RP/0/RP0/CPU0:router(config)# show configuration changes diff

  Building configuration...
  # hostname router
  # hostname bla
  - logging console
  - telnet vrf default ipv4 server disable
  - domain ipv4 host xhu-u5
  - domain ipv4 host coax-u10
  - domain ipv4 host coax-u10.cisco.com
  - domain name
  - interface Loopback1
  -  ipv4 address 10.0.0.2 255.255.255.224
  - !
  - interface Loopback2
  -  description
  - !
  - interface Loopback5
  -  description
  - !
  - interface Loopback6
  -  description
  - !
  - interface MgmtEth0/0/CPU0/0
```

```
-  ipv4 address 10.0.0.1 255.255.255.224
- !
- interface GigabitEthernet0/2/0/0
-  shutdown
- !
- interface GigabitEthernet0/2/0/1
-  shutdown
- !
- interface GigabitEthernet0/2/0/2
-  shutdown
- !
- router static
-  address-family ipv4 unicast
-   0.0.0.0/0 255.255.255.224
-  !
- !
end
```

show configuration commit changes

To display the changes made to the running configuration by previous configuration commits, a configuration commit, or for a range of configuration commits, use the **show configuration commit changes** command in EXEC, administration EXEC, administration configuration, or global configuration mode.

show configuration commit changes {*commit-id* | **since** *commit-id* | **last** *number-of-commits* | **all** } [**diff**]

Syntax Description

since	Displays all changes committed to the running configuration since (and including) a specific configuration commit.
commit-id	Displays configuration changes for a specific configuration commit.
last *number-of-commits*	Displays the changes made to the running configuration during the last number of configuration commits specified for the *number-of-commits* argument.
all	Displays commit ID and configurations completed for last 100 commits.
diff	(Optional) Displays added lines, changed lines, and deleted lines.

Command Default

None

Command Modes

EXEC

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Each time a configuration is committed with the **commit** command, the configuration commit operation is assigned a commit ID. The **show configuration commit changes** command displays the configuration changes made since the specified commit.

To display a list of the available commit IDs, enter the **show configuration commit list** command. You can also display the commit IDs by entering the **show configuration commit changes** command with the online help function (**?**).

You can't view commit IDs from a different release if the syntax or semantics of the configuration changed in the current release.

Note Syntax of a configuration refers to its structure and format, while the semantics of a configuration refers to its backend interpretation.

Task ID

Task ID	Operations
config-services	read

The following example shows sample output from the **show configuration commit changes** command. The output displays commit IDs.

```
RP/0/RP0/CPU0:router# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000077  lab       con0_1_C  CLI         15:42:45 UTC Fri Jan 30 2009
2    1000000076  lab       con0_1_C  Rollback    15:30:39 UTC Fri Jan 30 2009
3    1000000075  lab       con0_1_C  Rollback    15:25:26 UTC Fri Jan 30 2009
4    1000000074  lab       con0_1_C  Rollback    15:04:29 UTC Fri Jan 30 2009
5    1000000073  lab       con0_1_C  CLI         14:49:07 UTC Fri Jan 30 2009
6    1000000072  lab       con0_1_C  CLI         14:48:35 UTC Fri Jan 30 2009
```

The following example shows sample output from the **show configuration commit changes** command with the *commit-id* argument. In this example, the output displays the changes made in the configuration commit assigned commit ID 1000000077.

```
RP/0/RP0/CPU0:router# show configuration commit changes 1000000077

Building configuration...
alias exec shrun show configuration running
alias exec shver show version
end
```

The following example shows sample output from the **show configuration commit changes** command with the **since** *commit-id* keyword and argument. In this example, the output displays the configuration changes made since the configuration commit assigned commit ID 1000000077 was committed.

```
RP/0/RP0/CPU0:router# show configuration commit changes since 1000000077

Building configuration...
no hw-module node 0/RP0/CPU0 shutdown
hostname router
logging trap
no logging console
logging history size 1
alias exec shrun show configuration running
alias exec shver show version
interface MgmtEth0/RP1/CPU0/0
 ipv4 address 12.25.34.10 255.255.0.0
```

```
 no shutdown
 !
 interface preconfigure MgmtEth0/RP0/CPU0/0
 no shutdown
 !
no route ipv4 0.0.0.0/0 12.7.0.1
route ipv4 0.0.0.0/0 12.25.0.1
route ipv4 223.255.254.254/32 12.25.0.1
telnet ipv4 server enable
end
```

The following example shows sample output from the **show configuration commit changes** command with the **diff** keyword. In the display, the following symbols signify changes:

+ indicates an added line.

– indicates a deleted line.

# indicates a modified line.

```
RP/0/RP0/CPU0:router# show configuration commit changes last 1 diff

Building configuration...
+ interface Loopback1000
+ ipv4 address 190.190.180.1 255.255.255.255
!
end

+ interface Loopback1000
+ ipv4 address 190.190.180.1 255.255.255.255
  !
  end
```

The following example shows sample output from the **show configuration commit changes** command with the **all** keyword. In this example, the output displays the list of configurations that are committed in last 100 commits along with their commit-ID.

```
RP/0/RP0/CPU0:router# show configuration commit changes all
Commit ID :  1000000001
-------------------------------------------------------
Building configuration...
!! IOS XR Configuration 0.0.0
interface GigabitEthernet0/0/0/1
speed 100
!
end

Commit ID :  1000000002
-------------------------------------------------------
Building configuration...
!! IOS XR Configuration 0.0.0
interface GigabitEthernet0/0/0/1
no speed 100
!
end

Commit ID :  1000000003
-------------------------------------------------------
Building configuration...
```

```
!! IOS XR Configuration 0.0.0
interface GigabitEthernet0/0/0/0
!
shutdown
end
```

show configuration commit list

To display information about the configuration commits stored in the commit database, use the **show configuration commit list** command in EXEC, administration EXEC, administration configuration, or global configuration mode.

show configuration commit list [*number-of-commits*] [**detail**]

Syntax Description

number-of-commits	(Optional) Number of commits (beginning with the most recent commit) that are available for rollback.
detail	(Optional) Displays detailed commit information, including comments.

Command Default

If this command is entered without any optional arguments or keywords, the output displays information about all the configuration commits stored in the commit database.

Command Modes

EXEC

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show configuration commit list** command to list the commit IDs (up to 100) that are available for rollback.

Note The most recent 100 commits are retained by the system. As new commit IDs are added, the oldest commit IDs are discarded and are no longer available for rollback operations.

Task ID

Task ID	Operations
config-services	read

The following example shows sample output from the **show configuration commit list** command. The output displays the commit IDs that are available for rollback.

```
RP/0/RP0/CPU0:router# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000010  UNKNOWN   con0_0_C  Rollback    02:25:53 UTC Fri Feb 06 2009
2    1000000009  UNKNOWN   con0_0_C  CLI         02:23:09 UTC Fri Feb 06 2009
```

```
3     1000000008  UNKNOWN   con0_0_C  CLI         02:22:54 UTC Fri Feb 06 2009
4     1000000007  UNKNOWN   con0_0_C  CLI         02:22:18 UTC Fri Feb 06 2009
5     1000000006  UNKNOWN   con0_0_C  CLI         02:07:21 UTC Fri Feb 06 2009
```

The following table describes the significant fields shown in the display.

Table 13: show configuration commit list Field Descriptions

Field	Description
SNo.	Serial number of the commit entry.
Label/ID	If a label was assigned to a commit, the first 10 characters of the label display; otherwise, the autogenerated commit ID displays.
User	User who executed the commit.
Line	Line in which the user session was established. In some cases, this field may display "UNKNOWN" or "SYSTEM". These fields indicate that an internal commit was made by the system.
Client	The management interface used to make the commit.
Time Stamp	Time and date when the commit was executed.

show configuration (config)

To display information about the current configuration session (target configuration), use the **show configuration** command in any configuration mode.

show configuration [**merge**] [**running**]

Syntax Description

merge	(Optional) Displays the configuration that occurs if the contents of the uncommitted changed (target configuration) are committed to the running configuration.
running	(Optional) Displays the running (committed) configuration.

Command Default

When the **show configuration** command is entered without an argument, the uncommitted changes to the target configuration are displayed.

Command Modes

Any configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show configuration** command to display details on uncommitted configuration changes.

Use the **show configuration** command with the **running** keyword to display the running (active) configuration.

Prior to committing the target configuration, use the **show configuration** command with the **merge** keyword from any configuration mode to display the result of merging the target configuration with the running configuration.

Task ID

Task ID	Operations
basic-services	read

In this example, the **show configuration** command displays uncommitted changes made during a configuration session:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige0/3/0/3
RP/0/RP0/CPU0:router(config-if)# description faq
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.10.11.20 255.0.0.0
RP/0/RP0/CPU0:router(config-if)# show configuration

Building configuration...
interface TenGigE0/3/0/3
 description faq
 ipv4 address 10.10.11.20 255.0.0.0
 end
```

The following example shows sample output from the **show configuration** command with the optional **merge** keyword. The command is entered during a configuration session. The output displays the result of merging the target and running configuration, without committing the changes.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tengige0/3/0/3
RP/0/RP0/CPU0:router(config-if)# description faq
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.10.11.20 255.0.0.0
RP/0/RP0/CPU0:router(config-if)# show configuration merge

Building configuration...
hostname router
interface TenGigE0/0/0/0
 ipv4 address 1.2.3.4 255.0.0.0
 exit
interface TenGigE0/3/0/3
 description faq
 ipv4 address 1.1.1.1 255.0.0.0
 shutdown
 end
```

show configuration failed incompatible

To display any configurations that were removed from the running configuration because they were not understood by the software being activated, use the **show configuration failed incompatible** command in EXEC or administration EXEC mode.

show configuration failed incompatible

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Any configurations in the running configuration that are not understood by new software being installed are removed from the running configuration. To see which configurations were removed, use the **show configuration failed incompatible** command.

Task ID

Task ID	Operations
config-services	read

show configuration failed remove

To display information about a configuration that failed while being removed during installation operations, use the **show configuration failed remove** command in EXEC or administration EXEC mode.

show configuration failed remove

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read

The following example shows a failed commit operation:

```
RP/0/RP0/CPU0:router# show configuration failed remove

!! SEMANTIC ERRORS: This configuration was rejected by
!! the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.

multicast-routing
 no address-family ipv4
 !!% Process did not respond to sysmgr
 address-family ipv4
  no interface all enable
!!% Process did not respond to sysmgr
!
!
```

Because the configuration failed to be removed, it is still displayed in the output from the **show running-configuration** command as expected:

```
RP/0/RP0/CPU0:router# show running-configuration
...
router pim vrf default address-family ipv4
 auto-rp candidate-rp GigabitEthernet0/2/0/3 scope 255 group-list 224/4 interval 10
!
multicast-routing
```

```
 address-family ipv4
  interface all enable
!
!
```

show configuration failed rollback

To display information about a configuration that failed in the last rollback operation, use the **show configuration failed rollback** command in EXEC or administration EXEC mode.

show configuration failed rollback

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read
root-lr	read

show configuration failed startup

To display information about a configuration that failed at startup, use the **show configuration failed** command in EXEC or administration EXEC mode.

show configuration failed startup [**noerror** | **previous** *number*]

Syntax Description

noerror	(Optional) Displays the configuration that failed at startup without an error reason.
previous *number*	(Optional) Displays the previous failed startup configuration or configurations. The *number* argument is a value from 1 to 4, which displays the failed startup configurations in previous of sessions.

Command Default

If no keywords are specified, this command displays the details of the failed startup configuration including error reasons.

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read

show configuration history

To display a history of configuration events, use the **show configuration history** command in EXEC, administration EXEC, administration configuration, or global configuration mode.

show configuration history [**alarm** | **backup** | **cfs-check** | **commit** | **rebase** | **shutdown** | **startup**] [**first** *number* | **last** *number* | **reverse**] [**detail**]

Syntax Description

alarm	(Optional) Displays alarm events.
backup	(Optional) Displays configuration backup events.
cfs-check	(Optional) Displays CFS check events.
commit	(Optional) Displays commit events.
rebase	(Optional) Displays commit database consolidation events.
shutdown	(Optional) Displays shutdown events.
startup	(Optional) Displays startup events, including alternate configurations, failed configurations, and other events.
first *number*	(Optional) Displays the first x number of events, where x is the *number* argument.
last *number*	(Optional) Displays the last *x number* events. Replace with the number of events to display.
reverse	(Optional) Displays the most recent events first.
detail	(Optional) Displays detailed information, including comments.

Command Default

When entered without any optional arguments or keywords, this command displays all configuration events. The oldest events are displayed at the top of the list for each event type.

Command Modes

EXEC

Administration EXEC

Administration configuration

Global Configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show configuration history** command to display information about the last (up to) 1500 configuration events.

Use one of the available keywords to display the configuration event only for that event type. Use the **first** *number* and **last** *number* keywords and arguments to display a specified number of events. Use the **reverse** keyword to display the newest events at the top of the list.

Task ID

Task ID	Operations
config-services	read

In the following example, the **show configuration history** command is used to display the history of all configuration events for an SDR:

```
RP/0/RP0/CPU0:router# show configuration history

Sno.  Event      Info                           Time Stamp
~~~~  ~~~~~      ~~~~                           ~~~~~~~~~~
1     alarm      inconsistency alarm raised     Thu Jun 22 15:23:15 2009
2     startup    configuration applied          Thu Jun 22 15:23:32 2009
3     OIR config restore                        Thu Jun 22 15:23:25 2009
4     OIR config restore                        Thu Jun 22 15:23:33 2009
5     OIR config restore                        Thu Jun 22 15:23:33 2009
6     OIR config restore                        Thu Jun 22 15:23:34 2009
7     OIR config restore                        Thu Jun 22 15:23:34 2009
8     OIR config restore                        Thu Jun 22 15:23:35 2009
9     OIR config restore                        Thu Jun 22 15:23:36 2009
10    OIR config restore                        Thu Jun 22 15:23:37 2009
11    OIR config restore                        Thu Jun 22 15:23:37 2009
12    OIR config restore                        Thu Jun 22 15:23:38 2009
13    OIR config restore                        Thu Jun 22 15:23:38 2009
14    OIR config restore                        Thu Jun 22 15:23:39 2009
15    OIR config restore                        Thu Jun 22 15:23:39 2009
16    OIR config restore                        Thu Jun 22 15:23:40 2009
17    OIR config restore                        Thu Jun 22 15:23:40 2009
18    OIR config restore                        Thu Jun 22 15:23:42 2009
19    OIR config restore                        Thu Jun 22 15:23:42 2009
20    OIR config restore                        Thu Jun 22 15:23:42 2009
21    OIR config restore                        Thu Jun 22 15:23:43 2009
 --More--
```

In the following example, the **show configuration history** command is used to display only the startup configuration events:

```
RP/0/RP0/CPU0:router# show configuration history startup

Sno.  Event      Info                           Time Stamp
~~~~  ~~~~~      ~~~~                           ~~~~~~~~~~
1     startup    configuration applied          Thu Jun 22 15:23:32 2009
2     startup    configuration applied          Sat Jul  1 15:02:24 2009
3     startup    configuration applied          Sat Jul  8 17:36:52 2009
4     startup    configuration applied          Sun Jul  9 13:40:27 2009
5     startup    configuration applied          Sat Jul 15 18:18:54 2009
```

In the following example, the **show configuration history** command with the **commit detail** keywords is used to display additional details regarding the commit events:

```
RP/0/RP0/CPU0:router# show configuration history commit detail

1)  Event: commit         Time: Thu Jun 22 15:44:33 2009
     Commit ID: 1000000001 Label:
     User: lab        Line: vty0
     Client: CLI           Comment:

2)  Event: commit         Time: Thu Jun 22 16:58:18 2009
```

```
     Commit ID: 1000000002 Label:
     User: lab        Line: vty2
     Client: CLI           Comment:

3)  Event: commit         Time: Thu Jun 22 16:58:39 2009
     Commit ID: 1000000003 Label:
     User: lab        Line: vty2
     Client: CLI           Comment:

4)  Event: commit         Time: Sat Jul  1 15:29:31 2009
     Commit ID: 1000000001 Label:
     User: lab        Line: vty0
     Client: CLI           Comment:

5)  Event: commit         Time: Sat Jul  1 15:32:25 2009
     Commit ID: 1000000002 Label:
     User: lab        Line: vty0
  --More--
```

Table 14: show configuration history Field Descriptions

Field	Description
SNo.	Serial number of the entry.
Event	Type of configuration event.
Info	Summary of the configuration action.
Time Stamp	Time and date when the event was run.
Label/ID	If a label was assigned to a commit, the first 10 characters display; otherwise, the autogenerated commit ID displays.
User	User who issued the command.
Line	Line in which the user session was established. In some cases, this field may display “UNKNOWN” or “SYSTEM”. These fields indicate that an internal action was made by the system.
Client	The management interface used to make the event.

show configuration inconsistency replica

To display any configuration inconsistencies on a replica node, use the **show configuration inconsistency replica** command in EXEC or administration EXEC mode.

show configuration inconsistency replica location *node-id* [**detail**]

Syntax Description

location *node-id*	Displays any configuration inconsistencies on the designated node. The *node-id* argument is expressed in the *rack/slot* notation.
detail	Displays a detailed list of inconsistencies.

Command Default

Administration EXEC mode: Displays configuration inconsistencies for the admin plane configuration.

EXEC mode: Displays configuration inconsistencies for an SDR configuration.

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

In administration EXEC or EXEC mode, the replica node for the **show configuration inconsistency replica** command is the standby designated system controller (DSC). .

Use the **show configuration inconsistency replica** command, before performing a manual switchover or DSC migration, to verify that the node in line to take over for the DSC is in good shape. If any problems are reported, use the **clear configuration inconsistency replica** command to correct them.

Task ID

Task ID	Operations
config-services	read

The following example shows a configuration with inconsistencies:

```
RP/0/RSP0/CPU0:router# show configuration inconsistency replica location 0/rsp1/cpu0

The replica at location 0/RSP1/CPU0 is inconsistent.
Please run 'clear configuration inconsistency replica location 0/RP1/CPU0'.
```

The following example shows sample output after the inconsistencies have been resolved:

```
RP/0/RSP0/CPU0:Router# show configuration inconsistency replica location 0/rsp1/cpu0

Replica is consistent
```

show configuration persistent

To display the persistent configuration, use the **show configuration persistent** command in EXEC mode.

show configuration persistent [**diff**]

Syntax Description

diff	(Optional) Displays the difference between the running configuration and persistent configuration. This option is available only on the DSC.

Command Default

If no argument is specified, the **show configuration persistent** command displays the entire contents of the persistent configuration file.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The persistent configuration is the configuration stored in nonvolatile memory, from which the running configuration is restored after the router is reloaded. The running configuration should be the same as the persistent configuration. Use the **show configuration persistent** command with the **diff** keyword to check if there is a difference between the running configuration and the persistent configuration.

Task ID

Task ID	Operations
config-services	read

The following example shows that there is no difference between the running configuration and the persistent configuration:

```
RP/0/RP0/CPU0:router# show configuration persistent diff

Building configuration...
  end
```

The following example shows a difference between the running configuration and the persistent configuration:

```
RP/0/RP0/CPU0:router# show configuration persistent diff

Building configuration...
router vrrp
interface tengige0/1/0/1.1
vrrp 1 preempt delay 300
!
interface tengiget0/1/0/1.2
vrrp 1 preempt delay 300
!
interface tengige0/1/0/1.3
```

```
vrrp 1 preempt delay 300
```

show configuration removed

To display a configuration removed during installation operations, use the **show configuration removed** command in EXEC or administration EXEC mode.

show configuration removed *config-id*

Syntax Description

config-id	Name of removed configuration. Type (?) to see a list of the names of all removed configurations.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
config-services	read

The following example shows a removed configuration:

```
RP/0/RP0/CPU0:router# show configuration removed 20060301112919.cfg

xml agent corba
http server
end
```

show configuration rollback changes

To display changes that would be made by the **rollback configuration** command or to display the list of commit IDs, use the **show configuration rollback changes** command in EXEC, administration EXEC, administration configuration, or global configuration mode.

show configuration rollback changes {*commit-id* | **to** *commit-id* | **last** *number-of-commits*} [**diff**]

Syntax Description

commit-id	Name of configuration. When a specific *commit-id* is specified, only the changes that would occur if only the specified commit is rolled back are displayed.
to *commit-id*	Displays the changes that will occur to the running configuration if the system is rolled back to the configuration specified with the *commit-id* argument.
last *number-of-commits*	Displays the changes that will occur to the running configuration if the system is rolled back to the last number of commits specified with the *number-of-commits* argument.
diff	(Optional) Displays added lines, changed lines, and deleted lines.

Command Default

None

Command Modes

EXEC

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines



Note The most recent 100 commits are retained by the system. As new commit IDs are added, the oldest commit IDs are discarded and are no longer available for rollback operations.

Use the *commit-id* argument without the **to** keyword to display the changes for a particular commit. This can be useful for troubleshooting actions of the **rollback configuration** command.

Task ID

Task ID	Operations
config-services	read

The following example shows sample output from the **show configuration rollback changes** command with the **to** *commit-id* keyword and argument. The output displays the configuration

changes that would occur if the configuration were to be rolled back to the configuration commit specified for the argument.

```
RP/0/RP0/CPU0:router# show configuration rollback changes to 1000000007

Building configuration...
hostname old-name
end
```

The following example shows sample output from the **show configuration rollback changes** command **last** *number-of-commits* keyword and argument. The output displays the configuration changes that would occur if the configuration were to be rolled back to the number of configuration commits specified for the argument.

```
RP/0/RP0/CPU0:router# show configuration rollback changes last 2

Building configuration...
hostname orig_name
interface POS0/1/0/1
 shutdown
!
end
```

The following example shows sample output from the **show configuration rollback changes** command with the **diff** keyword.

In the display, the following symbols signify changes:

- + indicates an added line.
- – indicates a deleted line.
- # indicates a modified line.

```
RP/0/RP0/CPU0:router
show configuration rollback changes last 1 diff

Building configuration...
 interface Loopback1000
#  ipv4 address 1.1.1.1 255.255.255.255
!
end
```

show configuration running

To display the running configuration, use the **show configuration running** command in the appropriate mode.

show configuration running [*config-keyword*]

Syntax Description

config-keyword	(Optional) Specific configuration to display.

Command Default

None

Command Modes

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show configuration running** command to display the currently active configuration.

Task ID

Task ID	Operations
basic-services	read

This example shows the currently running (committed) configuration from administration mode.

```
RP/0/RP0/CPU0:router(admin)# show configuration running

Building configuration...
username lab
 secret 5 $1$XNWt$j8RscNdncKSRoMSnqSpbj/
 group root-system
!
end
```

show configuration running-config

To display the running configuration, use the **show configuration running-config** command in EXEC mode.

show configuration running-config [*config-keyword*]

Syntax Description

inheritance	(Optional) Displays the configuration inherited from any applied configuration group.
no-annotation	(Optional) Suppresses the display of inheritance messages, when the **inheritance** keyword is used.
config-keyword	(Optional) Specific configuration to display.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the *config-keyword* argument to display the running configuration for a specific keyword only.

Display the Configuration from Configuration Groups

By default, if configuration groups are applied in the configuration, they are displayed as they are configured. For example:

```
RP/0/RP0/CPU0:router# show configuration running-config
!
group G-INTERFACE-MTU
  interface 'GigabitEthernet.*'
    mtu 1500
end-group
!
interface GigabitEthernet1/0/0/7
  apply-group G-INTERFACE-MTU
!
```

To display the actual configuration as inherited from any applied configuration groups, use the **inheritance** keyword:

```
RP/0/RP0/CPU0:router# show configuration running-config inheritance
!
interface GigabitEthernet1/0/0/7
 ## Inherited from group G-INTERFACE-MTU
 mtu 1500
!
```

Use the **no-annotation** keyword to suppress the display of the Inheritance messages, "## Inherited from group ...".

Task ID

Task ID	Operations
basic-services	read

This example shows the currently running (committed) configuration:

```
RP/0/RP0/CPU0:router# show configuration running-config

Building configuration...

  !! Last configuration change at 15:36:31 UTC Thu Nov 17 2009 by lab
  sessions  Users with active configuration sess
!n
hostname router
line consolestartup   Sh
 exec-timeout 0 0onfiguration
!
logging console debugging
  |         Ou
snmp-server community public RW
  <cr>
RP/0/0/
ipv4 source-routeadmin)#show confi
key chain IPSLA ?
 key 10
  key-string password 1
  ipv4 address 10.0.0.0 255.255.255.0
  encapsulation ppp
  keepalive disable
!
 interface POS0/7/0/0
 shutdown
!
interface POS0/7/0/1
  shutdown
!
interface POS0/7/0/2
 shutdown
!
interface POS0/7/0/3
 shutdown
!
 route ipv4 0.0.0.0/0 12.7.0.1
ipsla
  responder
 !
!
end
```

show configuration sessions

To display the active configuration sessions, use the **show configuration sessions** command in EXEC or administration EXEC mode.

show configuration sessions [**detail**]

Syntax Description

detail	(Optional) Displays detailed information.

Command Default

None

Command Modes

EXEC

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show configuration sessions** command to display the active configuration sessions. Use the **clear configuration sessions** command to clear a configuration session. The **show configuration sessions** command can be used with the **clear configuration sessions** command to verify that an active configuration session was cleared.

Task ID

Task ID	Operations
config-services	read

The following example shows sample output from the **show configuration sessions** command:

```
RP/0/RP0/CPU0:router# show configuration sessions

Current Configuration Session  Line       User     Date                     Lock
00000050-001200bb-00000000     con0_5_CPU cisco    Fri Feb 16 17:23:47 2007
```

Table 15: show configuration sessions Field Descriptions

Field	Description
Session	System-generated configuration session ID number.
Line	Line in which the user session was established. In some cases, this field may display “UNKNOWN” or “SYSTEM.” These fields indicate that an internal commit was made by the system.
User	User who initiated the configuration session.
Date	Time and date the configuration session was started.

Field	Description
Lock	Locked running-configuration. An asterisk (*) displayed in this field means the session has been locked. Only one session can lock the running configuration at a time.

show default-afi-safi-vrf

To display the default address family identifier (AFI), subaddress family identifier (SAFI), and VPN routing and forwarding (VRF) instance for the current session, use the **show default-afi-safi-vrf** command in EXEC mode.

show default-afi-safi-vrf

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show default-afi-safi-vrf** command to display the default AFI and SAFI settings for the current session. The AFI and SAFI settings are controlled by the following commands:

- set default-afi, on page 136
- set default-safi, on page 137
- set default-vrf, on page 138

Task ID

Task ID	Operations
basic-services	read

The following example shows sample output from the **show default-afi-safi-vrf** command:

```
RP/0/RP0/CPU0:router# show default-afi-safi-vrf

%% Default AFI/SAFI/VRF for this session is:
     Address Family Identifier:       'ipv4'
     Sub-Address Family Identifier:   'unicast'
     Virtual Routing/Forwarding:      ''
```

show history

To display a history of commands executed in EXEC, administration EXEC, administration configuration, or global configuration mode use the **show history** command in one of the supported modes.

show history [**detail**]

Syntax Description

detail	(Optional) Displays detailed history information.

Command Default

None

Command Modes

EXEC

Administration EXEC

Administration configuration

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show history** command displays a history of the command entered for the current command mode. For example, enter the **show history** command to display a history of commands entered in EXEC mode. Enter the **show history** command in global configuration mode to display a history of the commands entered in global configuration mode.

Task ID

Task ID	Operations
config-services	read
basic-services	read

In the following example, the **show history** command is run in EXEC mode to display a history of the command entered inEXEC mode:

```
RP/0/RP0/CPU0:router# show history
configure
admin
show history
```

In the following example, the **show history** command is run in global configuration mode to display a history of the command entered in global configuration mode:

```
RP/0/RP0/CPU0:router(config)# show history
interface pos 0/1/0/0
ipv4 address 10.0.0.0
root
end
```

```
describe line default autocommand config
line default autocommand configure
end
show history
```

template

To create a template name and enter template configuration mode, use the **template** command in global configuration mode. To remove a template definition, use the **no** form of this command.

template *name*
no template *name*

Syntax Description

name	Unique name for the template to be created.

Command Default

No templates are defined.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **template** command to enter template configuration mode. From template configuration mode, you can group a subset of configuration commands in a named template. Commonly used sets of configuration commands can be grouped into a named template. Defining a template is similar to creating a C macro function. A template provides modularity and ease of use during user configuration.

Use the **end-template** command to exit template configuration mode. After defining a template, use the **apply-template** command to apply the template. Use the **show running-config** command with the optional **template** keyword and *template-name* argument to display the contents of a template.

Task ID

Task ID	Operations
config-services	read, write

The following example shows how to enter template configuration mode to create a template. In this example, a template named “pre-pos” is defined for the preconfigured Packet-over-SONET/SDH (POS) interface 0/1/0/1. The **end-template** command is used to exit from template configuration mode.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# template pre-pos
RP/0/RP0/CPU0:router(config-TPL)# interface preconfigure pos0/1/0/0
RP/0/RP0/CPU0:router(config-if-pre)# ipv4 address 10.3.32.154 255.0.0.0
RP/0/RP0/CPU0:router(config-if-pre)# end-template
RP/0/RP0/CPU0:router(config)#
```

Note After configuring a template, you may want to display the contents of the configured template. To display a template configuration, use the **show running-config** command with the **template** *name* keyword and argument.

The following example shows sample output from the **show running-config** command with the **template** *name* keyword and argument. In this example, the output displays the contents of a template named "pre-pos."

```
RP/0/RP0/CPU0:router# show running-config template pre-pos

template pre-pos
 interface preconfigure POS0/1/0/0
  ipv4 address 10.3.32.154 255.0.0.0
!
end-template
```



File System Commands

- cd, on page 184
- cfs check, on page 185
- clear configuration ascii inconsistency, on page 187
- copy, on page 188
- delete, on page 196
- dir, on page 197
- mkdir, on page 199
- pwd, on page 200
- rmdir, on page 201
- show filesystem, on page 202

cd

To change the current working directory, use **cd** command in XR EXEC mode.

cd *filesystem* **:**

Syntax Description

filesystem **:**	(Optional) Location of the new working directory. Include the file system alias for the *filesystem* argument, followed by a colon and optionally, the name of a directory.

Command Default

The default file directory is **disk0:/usr**.

Command Modes

XR EXEC mode.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The current working directory is the directory used when EXEC commands that have an optional argument are entered without that argument. Use **cd** command to define the working directory. For example, when the **dir** command is entered without specifying the *filesystem* argument, the files in the current working directory are displayed.

Use **cd** command without an argument to set the working directory back to the default directory, **disk0:/usr**.

The following example shows how to change the current working directory to the root directory on the hard disk. In this example, the **pwd** command confirms that the working directory has changed to the root directory on the hard disk.

```
RP/0/RP0/CPU0:router# cd harddisk:
RP/0/RP0/CPU0:router# pwd

harddisk:
```

The following example shows how to change the current working directory to the default file directory by specifying the **cd** command without a location. In this example, the **pwd** command confirms that the working directory has changed to the default file directory.

```
RP/0/RP0/CPU0:router# cd
RP/0/RP0/CPU0:router# pwd

disk0:/usr
```

cfs check

To clear any inconsistencies between running configuration and binary startup configuration maintained on the disk use **cfs check** command in XR EXEC mode.

cfs check

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.1	This command was introduced.
Release 24.2.11	This command was modified to increase the commit count from 20 to 40.

Usage Guidelines

Use this command to clear any inconsistencies between running configuration and binary startup configuration maintained on the disk.

On executing this command the existing binary startup configuration maintained on the disk will be discarded and the entire binary startup configuration will be recreated from system's running configuration. This will clear any inconsistency between the two.

Note While this command runs, redundancy of the designated shelf controller (DSC) is disabled.

Note Executing this command will take a lock to the configuration database, which will prevent any commit operation until this operation completes.

Task ID

Task ID	Operations
root-lr	read, write

Examples

The following example shows how to perform a CFS check:

```
Router# cfs check

Creating any missing directories in Configuration File system...OK
Initializing Configuration Version Manager...OK
Syncing commit database with running configuration...OK
Re-initializing cache files...OK
```

```
Updating Commit Database.  Please wait...[OK]
```

Related Commands

Command	Description
show configuration history	Displays **cfs check** events executed successfully.
clear configuration inconsistency	Performs the same operation as **cfs check**, can be used interchangeably.

clear configuration ascii inconsistency

To perform an ASCII backup of the system's running configuration and to clear inconsistencies between running configuration and ASCII backup copy maintained on the disk, use the **clear configuration ascii inconsistency** command in XR EXEC mode.

clear configuration ascii inconsistency

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.5.1	This command was introduced.
Release 24.2.11	This command was modified to include resetting the ASCII backup timer.

Usage Guidelines

Use this command to perform a forced ASCII backup and reset the periodic ASCII backup timer. Once the backup is complete, the router will automatically initiate the next periodic ASCII backup operation only after 55 minutes from the time the **clear configuration ascii inconsistency** command is executed.

On executing this command, the ASCII backup will synchronize with the latest running configuration up to the point of the last commit made before executing the command. This clears any inconsistencies between the running configuration and the ASCII backup copy stored on disk. Additionally, this command will reset the periodic ASCII backup timer.

Task ID

Task ID	Operations
config-services	execute

Examples

The following example shows how to perform an ASCII backup and reset the ASCII backup timer to zero:

```
Router# clear configuration ascii inconsistency
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Warning: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!! It is recommended to run this command only when all nodes in router      !!!!
!!!! are in IOS-XR RUN state. To determine node state, run following command: !!!!
!!!! 'show platform'.                                                         !!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Warning: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Proceed with the command ?[confirm] y
 Ascii configuration backup is in progress...
Configuration ascii backup complete
```

Related Commands

Command	Description
show configuration history	Displays **clear configuration ascii inconsistency** events executed successfully.

copy

To copy a file from a source (such as a network server) to a destination (such as a flash disk), use **copy** command in XR EXEC mode.

copy *source* { **location** *node-id* *destination* **location** { *node-id* | **all** } | **running-config** [**atomic**] }

Syntax Description

source	Filename including the directory path or network location of the file. The possible sources are: *directory-path* —Directory path of the file from which the file is copied. **access-list** { **ipv4**	**ipv6** }—Copies an access list (EXEC mode only). **config:** —Copies from disk0: file system. **disk0:** —Copies from disk0: file system. **ftp:** —Copies from an FTP network server. The syntax is **ftp:**[[[//*username* [:*password*]@] *location*]/*directory*]/*filename.* **harddisk:** —Copies from the hard disk drive file system (if present). **http:** —Copies from one webserver to another over a network. The syntax is **http:**//*username* **:** *password*@*ip-address* **:** *port* / *directory-path* **https:** —Copies from the https: file system. The syntax is **https:**//*username* **:** *password*@*ip-address* **:** *port* / *directory-path* **sftp:** —Copies from an SFTP network server. The syntax is **sftp:**[[[//*username* [:*password*]@] *location*]/*directory*]/*filename.* **scp:** —Copies from an SCP network server. The syntax is **scp:**[[[//*username* [:*password*]@] *location*]/*directory*]/*filename.* **prefix-list** {**ipv4**	**ipv6**}—Copies from a prefix list (EXEC mode only). **rootfs:** —Copies from the rootfs: file system. **running-config** —Copies from the current system configuration. **tftp:** —Copies from a TFTP network server. The syntax is **tftp:**[[//*location*]/*directory*]/*filename* **xml-schema** —Copies the XML schema files as a tar ball file (.tar.gz) [EXEC mode only].
destination	Filename including the directory path or network location of the file.
location *node-id*	Specifies a node. The *node-id* argument is expressed in the *rack*/*slot* notation.
location all	Copies to all nodes.
running-config	Applies the source configuration file to the running configuration of the system.
atomic	(Optional) Applies the changes to the running configuration only if there are no errors

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 25.4.1	The **ftp,tftp, and rcp** keywords are deprecated.
Release 7.10.1	This command was modified to support public key authentication.
Release 7.9.1	This command was modified to support SFTP and SCP options.
Release 7.0.12	This command was introduced.

Usage Guidelines Source and destination can each be a configuration file, a text file, or a file system. Enter source and destination URL information, usernames, and passwords and issue the **copy** command. The networking device prompts for any missing information.

The exact format of the *source* and *destination* arguments vary according to the file or directory location. Enter the device or network location for the file system type.

Filenames can include the following characters:

! # $ % + 0 1 2 3 4 5 6 7 8 9 @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ ] ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~

The following characters can be used with the stated limitations:

- ` needs backslash before this character
- – cannot be the first character
- . cannot be the last character
- = cannot be the filename without other characters

The following characters cannot be used in filenames:

" ( ) * , / : < > ? \ | ' & ;

The maximum length allowed for a filename is 254 characters including the path. If a filename longer than 254 characters is specified, the filename is truncated to 254 characters.

To copy a file from a source on the router to a destination on the router, specify a source **location** *node-id* and a destination **location** *node-id* . To copy the file to all nodes, use the **location all** keywords.

In the alias syntax for the **ftp:** , **rcp:** , **tftp:** , **sftp:** , and **scp:** keywords, the location is either an IP address or a hostname. The filename is specified relative to the directory used for file transfers.

When no alias is specified, the networking device looks for a file in the current directory. To view the current directory, enter the **pwd** command.

Note During processing of the **copy** command, you might see the “C” character. For all files being copied, “C” indicates that the copy process is taking place. The entire copying process might take several minutes and differs from protocol to protocol and from network to network.

Table 16: Network Protocols Supported by Cisco IOS XR Software

Prefix	Name	Description
tftp:	Trivial File Transfer Protocol	*TFTP* is a simplified version of FTP that allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password).
ftp:	File Transfer Protocol	*FTP* is an application protocol, part of the TCP/IP protocol stack, and is used for transferring files between network nodes. FTP requires a username and password.
rcp:	Remote Copy Protocol	The rcp protocol allows users to copy files to and from a file system residing on a remote host or server on the network. The rcp protocol uses TCP to ensure the reliable delivery of data. The rcp protocol downloads require a username.
http:	Hypertext Transfer Protocol	*HTTP* protocol allows users to transfer files from one webserver to another over a network. The user authentication depends on the webserver configuration.
sftp:	Secure File Transfer Protocol	*SFTP* is an application protocol is used for secure transferring files between the router and and an archieve server. SFTP requires a username and password.
scp:	Secure Copy Protocol	*SCP* is an application protocol is used for secure transferring files between the router and and an archieve server. SFTP requires a username and password.

Additional usage guidelines are in the following sections.

Invalid Combinations of Source and Destination

Some combinations of source and destination are invalid. Specifically, you cannot copy the following:

- From a running configuration to a running configuration
- From a network device to a network device (for example, **copy ftp: rcp:**)

Using TFTP

TFTP is a simplified version of FTP that allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password).

The syntax is as follows:

copy tftp://*hostname* /*ipaddress*/*directory-path pie name target-device* [**location** {*node-id* | **all**}]

Example:

```
Router# copy tftp://1.1.1.1/images/software.pie disk1:
```



Note Some Cisco IOS XR images may be larger than 32 MB, and the TFTP services provided by some vendors may not support a file this large. If you do not have access to a TFTP server that supports files larger than 32 MB, download the software image using FTP or rcp as described in the following sections.

Using FTP

FTP servers require a username and password for each client request. Cisco IOS XR software sends the first valid username in the following list:

1. The username and password specified in the **copy** command, if a username is specified.

   The syntax is as follows:

   copy ftp:// *username* **:** *password* **@** *hostname or ipaddress*/*directory-path*/*pie-name target-device* [**location** {*node-id* | **all**}]

   Example:

   ```
   Router# copy ftp://john:secret@10.1.1.1/images/software.pie disk1:
   ```

2. An "anonymous" username and password. The anonymous password is "root@ip address," where "ip address" is the IP address of the local networking device.
3. A password "username@iosname.domain" formed by the networking device. The variable "username" is the username associated with the current session, "iosname" is the configured hostname, and "domain" is the domain of the networking device.

The username and password must be associated with an account on the FTP server. If you are writing to the network server, the FTP server must be properly configured to accept the FTP write request from the user on the networking device.

If the network server has a directory structure, the configuration file or image is written to or copied from the directory associated with the username on the network server. For example, if the system image resides in the home directory of a user on the network server, specify the name of that user as the remote username.

Refer to the documentation for your FTP server for more details.

Using rcp

The rcp protocol requires a username upon each request. When you copy a configuration file or image between the networking device and an rcp server, the Cisco IOS XR software sends the first valid username in the following list:

1. The remote username specified in the **copy** command, if one is specified.
2. The username set by the **rcp client username** command, if the command is configured.
3. The networking device hostname.

For the rcp copy request to process successfully, an account must be defined on the network server for the remote username. If the network administrator of the destination server did not establish an account for the remote username, this command does not run successfully. If the network server has a directory structure, the configuration file or image is written to or copied from the directory associated with the remote username on

the network server. For example, if the system image resides in the home directory of a user on the network server, specify the name of that user as the remote username.

If you are writing to the network server, the rcp server must be properly configured to accept the rcp write request from the user on the networking device. For UNIX systems, add an entry to the .rhosts file for the remote user on the rcp server. Suppose the networking device contains the following configuration lines:

```
hostname Rtr1
ip rcp remote-username User0
```

If the IP address of the networking device translates to company.com, then the .rhosts file for User0 on the rcp server should contain the following line:

```
company.com Rtr1
```

See the documentation for your rcp server for more details.

If you are using a personal computer as a file server, the computer must support remote shell (rsh) protocol.

Note Starting with Cisco IOS XR Software Release 25.4.1, do not use the FTP, TFTP and RCP protocols for, as these are unsecure and deprecated. Instead, use SFTP (Secure File Transfer Protocol) or SCP (Secure Copy Protocol) for improved security, as they provide encrypted communication.

Syslog warnings will be generated if the following XR configuration and EXEC commands are used:

XR Configuration Commands:

- **ftp client**
- **ip ftp**
- **tftp client**
- **ip tftp**

EXEC Commands:

- **install add source ftp**
- **install add source tftp**
- **install source tftp**
- **install source ftp**
- **install replace tftp**
- **install replace ftp**
- **copy using ftp**
- **copy using tftp**
- **load ftp**
- **load tftp**

For XR configuration commands, periodic syslog warnings will be generated once every 30 days as long as FTP or TFTP configured. For EXEC commands, syslog warnings will be displayed each time these commands are executed.

Using xml-schema

Use the **xml-schema** keyword to obtain the most up-to-date XML schemas (.xsd files) from the router. Using this keyword is useful to prevent the use of outdated schemas in the event that router software updates include schema updates. The tar ball file includes all active schema files. It does not include schemas that are activated by specific package installation envelopes (PIEs) if those PIEs are not installed and activated on the router.

Using HTTP or HTTPS

You can transfer files from one webserver to another on a network using HTTP or HTTPS. The user authentication depends on the webserver configuration. Supported file copy operations are:

- Copy a file from webserver to device via HTTP or HTTPS.

- Copy a file from device to webserver via HTTP or HTTPS.

Note Copying a file from sys-admin via HTTP or HTTPS is not supported.

The syntax is as follows:

copyhttps://*username* **:** *password*@*ip-address* **:** *port target-device* [**location** {*node-id* | **all**}]

Example:

The following example shows how to copy a file from http server, where user credentials are not required, and server listens to the default port.

```
Router# copy http://1.1.1.1/images/software.pie disk1:
```

The following example shows how to copy a file from http server, where user credentials are required, and server listens to the default port.

```
Router# copy http://user:cisco@1.1.1.1/images/software.pie disk1:

UserID: user
Password: cisco
```

The following example shows how to copy a file from http server, where user credentials are required, and server listens to a specific port.

```
Router# copy http://user:cisco@1.1.1.1:45/images/software.pie disk1:


UserID: user
Password: cisco
Specific listen port: 45
```

Copying to the Running Configuration

When you use the **copy** command to copy a configuration file to the **running-config** destination, the configuration in the file is applied to the running configuration of the system. This is a configuration operation. By default, the copy is carried out in a best-effort manner. This means that if some configuration lines from the file cannot be applied, the remaining configuration is still integrated into the system. In this case, a partial configuration is committed. When the **atomic** keyword is used, partial configurations are not committed. This means that even if one error occurs in the parsing or committing phase, no changes are made to the system. To view any errors when applying the configuration, use the **show configuration failed** command.

Task ID

Task ID	Operations
filesystem	execute

The following example shows how to copy a file from a FTP server to disk1:

```
Router#copy ftp://john:secret@10.1.1.1/images/comp-cisco8k-full.pie disk1:
```

The following example shows how to copy a file from an rcp server to disk1:

```
Router#copy rcp://john@10.1.1.1/images/comp-cisco8k-full.pie disk1:
```

The following example shows how to copy a configuration file to running-config destination:

```
Router#copy running-config disk0:/running-config.txt
Thu Apr 20 15:53:49.116 UTC
Destination file name (control-c to cancel): [/disk0:/running-config.txt]?
Building configuration.
188 lines built in 1 second
[OK]
```

The following example shows how to copy a file from a SCP and SFTP server using public key authentication:

```
Router#copy running-config scp://root@192.0.4.2//var/opt/run_conf_scp.txt

Router#copy running-config sftp://root@192.0.4.2//var/opt/run_conf_sftp.txt
```

delete

To delete files, use **delete** command in the appropriate mode.

delete [/**noprompt**] [/**ena**] *filesystem***:** *filename* **location** {*node-id* | **all**}

Syntax Description

/noprompt	(Optional) Causes no prompt for confirmation before deleting the specified files.
/ena	(Optional) Deletes all files from and below the current working directory. Starting with Cisco IOS XR 26.2.1 release, the /**ena** option with the **delete** command is supported only when the total size of all file names to be deleted does not exceed the 16 KB size. For operations involving a larger number of files that exceed this limit, use the **delete** command without the **/ena** option.
filesystem **:**	(Optional) Location of the file to be deleted. Include the file system alias for the *filesystem* argument, followed by a colon, and, optionally, the name of a directory.
filename	Filename of the file to be deleted.
harddisk	Deletes the harddisk
location {*node-id* | **all**}	Deletes a file from a designated node. The *node-id* argument is entered in the *rack*/*slot* notation. The **all** keyword specifies to delete the file from all nodes.

Command Default

A filename must be specified. If a filename is entered without a file system or directory path, the present working directory is used.

Command Modes

XR EXEC mode.

.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When a file is deleted, it is removed from the system and cannot be restored (undeleted).

Use the **dir** command to display the list of files on a storage device.

The following example shows how to delete a file:

```
RP/0/RP0/CPU0:router# delete rbtest

Delete disk1:/rbtest[confirm]y
```

dir

To display a list of files on a file system or in a specific directory, use the **dir** command in XR EXEC mode .

dir [/**all** | /**ena** | /**recurse**] [*filesystem***:**] [*filename*] **location** {*node-id* | **all**}

Syntax Description

/all	(Optional) Lists deleted files, undeleted files, and files with errors.
/ena	(Optional) Recognizes subdirectories.
/recurse	(Optional) Recursively lists subdirectories.
filesystem **:**	(Optional) Name of the directory containing the files to be displayed. Include the file system alias for the *filesystem* argument, followed by a colon, and, optionally, the name of a directory.
filename	(Optional) Name of the files to display. The files can be of any type. You can use wildcards in the filename. A wildcard character (*) matches all patterns. Strings following a wildcard are ignored.
location {*node-id* | **all**}	(Optional) Specifies the node from which to display a list of files. The *node-id* argument is entered in the *rack*/*slot* notation. The **all** keyword specifies to display files on all nodes.

Command Default

When **dir** command is entered without keywords or arguments, the contents of the present working directory are displayed.

Command Modes

XR EXEC mode.

.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

If you enter the **dir** command without specifying a directory, the contents of the present working directory are displayed. The **all** keyword displays all files, including deleted files. The size associated with the directory name is the total size for all files in that directory.

The following example shows how to display the contents of a directory:

```
RP/0/RP0/CPU0:router# dir harddisk:/log

Directory of harddisk:/log

5527        drwx  4096        Thu Aug 28 11:21:48 2008  boot_28_Aug_2008_11_21_49
5533        drwx  4096        Thu Aug 28 11:38:54 2008  boot_28_Aug_2008_11_38_54
5538        drwx  4096        Fri Sep  5 13:28:54 2008  boot_05_Sep_2008_13_28_54
5543        drwx  4096        Mon Sep  8 08:55:52 2008  boot_08_Sep_2008_06_59_08
```

```
--More--
```

mkdir

To create a new directory on a file system, use the **mkdir** command in the appropriate mode.

mkdir *filesystem*:[**location** {*node-id* | **all**}]

Syntax Description

filesystem:	File system on which to create a new directory.
location {*node-id* | **all**}	(Optional) Specifies the node where the file system is located. The *node-id* argument is expressed in the *rack/slot* notation. Use the **all** keyword to indicate all nodes.

Command Default

No default behavior or values

Command Modes

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

After you issue the **mkdir** command, Cisco IOS XR software prompts you to specify the name of the directory to be created. When specifying the name of the new directory, include the directory path where you want the new directory to reside. If you do not specify a directory path, the new directory is created in the /usr directory of the file system specified for the *filesystem:* argument.

The following example shows how to create a directory named newdir. The **dir** command is used to verify that the directory has been added.

```
RP/0/RP0/CPU0:router# mkdir harddisk:

Create directory filename []?newdir
Created dir harddisk:/newdir
RP/0/RP0/CPU0:router# dir harddisk:

  Directory of harddisk:

  11193       drwx  4096        Fri Feb 13 06:45:05 2009  newdir
  37146       drwx  4096        Sun Dec 14 15:30:48 2008  malloc_dump
  43030       drwx  4096        Wed Dec 24 11:20:52 2008  tracebacks
  43035       drwx  4096        Thu Jan  8 18:59:18 2009  sau
  51026       drwx  4096        Sat Dec 27 02:52:46 2008  tempA
  51027       drwx  4096        Sat Dec 27 02:04:10 2008  dir.not.del
  -430307552  -rwx  342         Fri Jan 16 10:47:38 2009  running-config
  -430305504  -rwx  39790       Mon Jan 26 23:45:56 2009  cf.dat

  39929724928 bytes total (39883231232 bytes free)
```

pwd

To display the present working directory, use the **pwd** command in

EXEC mode

System Admin EXEC

pwd

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

EXEC

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **pwd** command to show what directory or file system is specified as the default by the **cd** command.

The following example shows how to display the present working directory:

```
RP/0/RP0/CPU0:router# pwd
disk0:/usr
```

rmdir

To remove an existing directory, use the **rmdir** command in the appropriate mode.

rmdir *filesystem*: **location** {*node-id* | **all**}

Syntax Description

filesystem	Name of the file system from which to delete a directory, followed by a colon.
location {*node-id* | **all**}	Specifies the node where the file system is located. The *node-id* argument is expressed in the *rack/slot* notation. Use the **all** keyword to indicate all nodes.

Command Default

No default behavior or values

Command Modes

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **rmdir** command to remove directories (for example, to free up disk space) from a file system. After you issue the **rmdir** command, the Cisco IOS XR software prompts you to specify the name of the directory to be deleted.

When a directory contains files, you must remove the files before deleting the directory. Use the **delete** command to remove files.

The following example shows how to delete a subdirectory from the hard disk. The **dir** command is used to verify that the directory has been deleted.

```
RP/0/RP0/CPU0:router# rmdir harddisk:

Remove directory filename []?newdir
Delete harddisk:/newdir[confirm]y
RP/0/RP0/CPU0:router# dir harddisk:

  Directory of harddisk:

  37146       drwx  4096        Sun Dec 14 15:30:48 2008  malloc_dump
  43030       drwx  4096        Wed Dec 24 11:20:52 2008  tracebacks
  43035       drwx  4096        Thu Jan  8 18:59:18 2009  sau
  51026       drwx  4096        Sat Dec 27 02:52:46 2008  tempA
  51027       drwx  4096        Sat Dec 27 02:04:10 2008  dir.not.del
  -430307552  -rwx  342         Fri Jan 16 10:47:38 2009  running-config
  -430305504  -rwx  39790       Mon Jan 26 23:45:56 2009  cf.dat

  39929724928 bytes total (39883235328 bytes free)
```

show filesystem

To display the layout and contents of file systems, use the **show filesystem** command in XR EXEC mode

System Admin EXEC

show **filesystem** *filesystem name* [**location** { *node-id* | **all** }] [**detail**]

Syntax Description

filesystem name	Name of the file system to display. File systems of type **ext2**, **ext4**, **tmpfs**, **devtmpfs**, **fuse.tftp_fs**, and **fuse.ftp_fs** are supported. Example: **disk0:**, **disk1:**, **harddisk:**
location {*node-id* | **all**}	(Optional) Specifies the node where the file system is located. The *node-id* argument is expressed in the *rack/slot* notation. Use the **all** keyword to indicate all nodes.
detail	Generates detailed information about each file system name or mount point, storage component or file system type, total size, available, and utilized space. File systems of type **ext2**, **ext4**, **tmpfs**, **devtmpfs**, **fuse.tftp_fs**, and **fuse.ftp_fs** are supported.

Command Default

The file system for the active RP is displayed.

Command Modes

XR EXEC mode

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.
Release 26.1.1	The **detail** keyword was introduced.

Usage Guidelines

Use the **show filesystem** command to learn the alias names (prefixes) of the file systems supported by your networking device.

The following example shows sample output from the **show filesystem** command:

```
RP/0/RP0/CPU0:router# show filesystem

File Systems:

       Size(b)      Free(b)        Type  Flags  Prefixes
         65536        53248       tmpfs     rw  /dev
    3097792512    919834624        ext4     rw  /
    3954253824   3953881088  flash-disk     rw  apphost:
    1014038528    178888704        ext4     rw  /dev/sde
   10304516096   9611448320       tmpfs     rw  /dev/shm
    4121808896   4121300992       tmpfs     rw  /run
    3097792512    919834624        ext4     ro  /proc/cmdline
      67108864     66973696       tmpfs     rw  /tmp
        524288       524288       tmpfs     rw  /mnt
   10304516096  10304516096       tmpfs     rw  /var/volatile
   14997434368  14994739200    harddisk     rw  harddisk:
    1848250368   1799409664        ext4     rw  /var/log
    1848250368   1845977088       flash     rw  /misc/config
```

```
  3697676288   3697311744  flash-disk     rw  disk0:
  4121808896   4121300992       tmpfs     rw  /run/netns
 14997434368  14994739200        ext4     rw  /install/instagt_cache
 14997434368  14994739200        ext4     rw  /install/tmp
 14997434368  14994739200        ext4     rw  /install/tmp_staging
           0            0     network     rw  ftp:
           0            0     network     rw  tftp:
 14997434368  14994739200        ext4     rw  /install/instmgr_cache
```

The following example shows sample output from the **show filesystem** command using the optional **location** *node-id* keyword and argument:

```
RP/0/RP0/CPU0:router# show filesystem location 0/rp0/cpu0

File Systems:

      Size(b)      Free(b)        Type  Flags  Prefixes
        65536        53248       tmpfs     rw  /dev
   3097792512    919834624        ext4     rw  /
   3954253824   3953881088  flash-disk     rw  apphost:
   1014038528    178888704        ext4     rw  /dev/sde
  10304516096   9611407360       tmpfs     rw  /dev/shm
   4121808896   4121300992       tmpfs     rw  /run
   3097792512    919834624        ext4     ro  /proc/cmdline
     67108864     66973696       tmpfs     rw  /tmp
       524288       524288       tmpfs     rw  /mnt
  10304516096  10304516096       tmpfs     rw  /var/volatile
  14997434368  14994739200    harddisk     rw  harddisk:
   1848250368   1799348224        ext4     rw  /var/log
   1848250368   1845977088       flash     rw  /misc/config
   3697676288   3697311744  flash-disk     rw  disk0:
   4121808896   4121300992       tmpfs     rw  /run/netns
  14997434368  14994739200        ext4     rw  /install/instagt_cache
  14997434368  14994739200        ext4     rw  /install/tmp
  14997434368  14994739200        ext4     rw  /install/tmp_staging
           0            0     network     rw  ftp:
           0            0     network     rw  tftp:
  14997434368  14994739200        ext4     rw  /install/instmgr_cache
```

Table 17: show filesystem Field Descriptions

Field	Description
Size(b)	Amount of memory in the file system, in bytes.
Free(b)	Amount of free memory in the file system, in bytes.
Type	Type of file system.
Flags	Permissions for file system.
Prefixes	Alias for the file system.

This example shows a truncated output of the **show filesystem detail** command.

```
RP/0/RP0/CPU0:ios#show filesystem detail
Tue Jan 13 10:37:57.670 UTC
Filesystem name:/run
storage-component:run
```

```
total size(Mb):1024
free(Mb):1023
free space available for user(Mb):1023
utilized(Mb):1
type:tmpfs

Filesystem name:/dev
storage-component:none
total size(Mb):9874
free(Mb):9874
free space available for user(Mb):9874
utilized(Mb):0
type:devtmpfs

Filesystem name:/
storage-component:/dev/mapper/main-xr-vg-install-rootfs-snapshot
total size(Mb):28198
free(Mb):21726
free space available for user(Mb):20280
utilized(Mb):6472
type:ext4

Filesystem name:/dev/shm
storage-component:tmpfs
total size(Mb):9904
free(Mb):8746
free space available for user(Mb):8746
utilized(Mb):1158
type:tmpfs

Filesystem name:/tmp
storage-component:tmpfs
total size(Mb):4096
free(Mb):4096
free space available for user(Mb):4096
utilized(Mb):0
type:tmpfs
```

Table 18: show filesystem detail Field Descriptions

Field	Description
Filesystem name	The mount point or directory where the filesystem is attached.
storage-component	The underlying storage device or logical volume backing the filesystem.
total size (Mb)	The total capacity of the file system in megabytes.
free (Mb)	The total free (unused) space in the files ystem in megabytes.
free space available for user (Mb)	The amount of free space available for non-privileged users in megabytes.
utilized (Mb)	The space currently used or occupied in the filesystem in megabytes.
type	The type of filesystem, for example, **mnt**, representing a mount point or mount namespace.



Manageability Commands

- aaa map-to username, on page 206
- iteration, on page 207
- nvgen default-sanitize, on page 209
- show xml schema, on page 210
- streaming, on page 211
- throttle, on page 212
- xml agent, on page 213
- xml agent ssl, on page 214
- xml agent tty, on page 215

aaa map-to username

To map the SPIFFE ID to a username for secure authentication and authorization mechanism in gRPC services, use the **aaa map-to username** command in the XR Config mode.

aaa map-to username *username* **spiffe-id any**

Syntax Description

username	Specifies the username that needs to be mapped with the SPIFFE ID.

Command Default

By default, the system uses `spiffe-user` as username.

Command Modes

XR Config mode

Command History

Release	Modification
Release 24.2.11	This command was introduced.

Usage Guidelines

Each SPIFFE ID supports only one username. Also, ensure that the username is already configured in the system.

Task ID

Task ID	Operations
config-services	read, write

Examples

The following example shows how to map the SPIFFE ID to a username in gRPC connection:

```
Router#configure
Router(config)#aaa map-to username cisco spiffe-id any
Router(config)#commit
```

iteration

To configure the iteration size for large XML agent responses, use the iteration command in xml agent configuration mode. To revert to the default iteration settings, use the **no** form of this command.

iteration {**off** | **on** **size** *iteration-size*}
no iteration

Syntax Description

off	Disables iteration, meaning that the entire XML response is returned, regardless of its size. Use of this option is not recommended.
on	Enables iteration, meaning that large XML responses are broken into chunks according to the iteration chunk size.
size *iteration-size*	Specifies the size of the iteration chunk, in Kbytes. Values can range from 1 to 100,000.

Command Default

Iteration is enabled; the *iteration-size* is 48.

Command Modes

XML agent

TTY XML agent

SSL XML agent

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When the XML agent returns a large response, it splits the response into chunks and returns one chunk at a time. External clients then need to send a GetNext request to obtain the next chunk. Use the **iteration** command to control the size of iteration chunks. A larger chunk value allows larger chunks to be received in a shorter period of time, possibly making the router system busier. A smaller chunk value allows smaller chunks to be received over a longer period of time, but does not make the router busy You can also specify to disable iteration completely using the **iteration off** command.



Note It is not recommended to disable iteration, since this could result in large transient memory usage.

To specify the TTY or SSL iteration size specifically, use the **iteration** command from the appropriate command mode.

Task ID

Task ID	Operations
config-services	read, write

Example

The following example shows how to configure the iteration chunk size to 100 Kbytes.

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml)# iteration on size 100
```

The following example shows how to disable iteration:

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml)# iteration off
```

The following example shows how to turn on iteration with the default iteration size:

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml)# no iteration off
```

The following example shows how to change the iteration size to the default iteration size.

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml)# no iteration on size 100
```

The following example shows how to change the iteration size of the TTY agent to 3 Kbytes:

```
RP/0/RP0/CPU0:router(config)# xml agent tty
RP/0/RP0/CPU0:router(config-xml-tty)# iteration on size 3
```

The following example shows how to turn off the iteration of the SSL agent:

```
RP/0/RP0/CPU0:router(config)# xml agent ssl
RP/0/RP0/CPU0:router(config-xml-ssl)# iteration off
```

nvgen default-sanitize

To enable sanitizing Strings, Usernames, Passwords, Comments, or IP Addresses in the output for **show running configurations** command, use the **nvgen default-sanitize** command.

nvgen default-sanitize { **strings** | **usernames** | **passwords** | **comments** | **ipaddrs** }

Syntax Description

strings	Removes the description strings in the running configuration and replaces it with **<removed>** phrase.
usernames	Removes the usernames in the running configuration and replaces it with **<removed>** phrase.
password	Removes the passwords in the running configuration and replaces it with **<removed>** phrase.
comments	Removes the comments in the running configuration and replaces it with **<comments removed>** phrase.
ipaddrs	Removes the IP addresses in the running configuration and replaces it with **<removed>** phrase.

Command Default

The output for **show running configurations** command includes sensitive information such as Strings, Usernames, Passwords, Comments, or IP Addresses.

Command Modes

Configuration mode

Command History

Release	Modification
Release 7.5.4	This command was introduced.

Usage Guidelines

None

Examples

The following example shows how to sanitize show running configurations:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# nvgen default-sanitize strings
RP/0/RP0/CPU0:router(config)# nvgen default-sanitize usernames
RP/0/RP0/CPU0:router(config)# nvgen default-sanitize passwords
RP/0/RP0/CPU0:router(config)# nvgen default-sanitize comments
RP/0/RP0/CPU0:router(config)# nvgen default-sanitize ipaddrs
RP/0/RP0/CPU0:router(config)# commit
```

show xml schema

To browse the XML schema and data, use the **show xml schema** command in

EXEC

mode.

show xml schema

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show xml schema** command runs the XML schema browser so that you can browse the XML schema and data.

Task ID

Task ID	Operations
config-services	read

This example shows how to enter the XML schema browser and the available commands:

```
RP/0/RP0/CPU0:router# show xml schema

Username: xxxx
Password:
Enter 'help' or '?' for help
xml-schema[config]:> ?

config            oper              action
adminoper         adminaction       cd
pwd               classinfo         list
ls                datalist          walk
walkdata          get               hierarchy
quit              exit              help
xml-schema[config]:>
```

streaming

To configure XML response streaming, use the **streaming** command in one of the XML agent configuration modes. To disable XML response streaming, use the **no** form of this command.

streaming on size *size*

Syntax Description

on	Turns on XML streaming.
size *size*	Specifies the size of the stream in Kbytes.

Command Default

XML streaming is disabled.

Command Modes

XML agent

XML agent ssl

XML agent tty

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read, write

This example illustrates how to set the XML response streaming size to 5000 Kbytes.

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml-agent)# streaming on size 5000
```

throttle

To configure the XML agent processing capabilities, use the **throttle** command in XML agent configuration mode.

throttle {**memory** *size* | **process-rate** *tags*}

Syntax Description

memory	Specifies the XML agent memory size.
size	Maximum memory usage of XML agent per session in MB. Values can range from 100 to 600. In IOS XR 64 bit, the values range from 100 to 1024. The default is 300.
process-rate	Specifies the XML agent processing rate.
tags	Number of tags that the XML agent can process per second. Values can range from 1000 to 30000.

Command Default

The process rate is not throttled; memory size is 300 MB.

Command Modes

XML agent configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **throttle** command to control CPU time used by the XML agent when it handles large data.

Task ID

Task ID	Operation
config-services	read, write

Example

This example illustrates how to configure the number of tags that the XML agent can process to 1000:

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml-agent)# throttle process-rate 1000
```

xml agent

To enable Extensible Markup Language (XML) requests over a dedicated TCP connection and enter XML agent configuration mode, use the **xml agent** command in

global configuration

mode. To disable XML requests over the dedicated TCP connection, use the **no** form of this command.



Note This command enables a new, enhanced-performance XML agent. The **xml agent tty** command enables the legacy XML agent and is supported for backward compatibility.

xml agent
no xml agent

Command Default XML requests are disabled.

Command Modes Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines There are two XML agents: a legacy XML agent and an enhanced-performance XML agent. We recommend that you use the enhanced-performance agent. The legacy agent is supported for backward compatibility. Use the **xml agent** command to enable the enhanced-performance XML agent. Use the **xml agent tty** command to enable the legacy XML agent.

Use the **no** form of the **xml agent** command to disable the enhanced-performance XML agent.

Task ID

Task ID	Operations
config-services	read, write

This example shows how to enable XML requests over a dedicated TCP connection:

```
RP/0/RP0/CPU0:router(config)# xml agent
```

xml agent ssl

To enable Extensible Markup Language (XML) requests over Secure Socket Layer (SSL) and enter SSL XML agent configuration mode, use the **xml agent ssl** command in

global configuration

mode. To disable XML requests over SSL, use the **no** form of this command.

xml agent ssl
no xml agent ssl

Command Default

SSL agent is disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The k9sec package is required to use the SSL agent. The configuration is rejected during commit when the security software package is not active on the system. When the security software package is deactivated after configuring SSL agent, the following syslog message is displayed to report that the SSL agent is no longer available.

```
xml_dedicated_ssl_agent[420]:
%MGBL-XML_TTY-7-SSLINIT : K9sec pie is not active, XML service over
SSL is not available.
```

Task ID

Task ID	Operations
config-services	read, write

This example shows how to enable XML requests over SSL:

```
RP/0/RP0/CPU0:router(config)# xml agent ssl
```

xml agent tty

To enable Extensible Markup Language (XML) requests over Secure Shell (SSH) and Telnet and enter TTY XML agent configuration mode, use the **xml agent tty** command in

global configuration

mode. To disable XML requests over SSH and Telnet, use the **no** form of this command.



Note This command enables a legacy XML agent that has been superceded by an enhanced performance XML agent and is supported only for backward compatibility. To enable the enhanced-performance XML agent, use the **xml agent** command.

xml agent tty
no xml agent tty

Command Default

XML requests over SSH and Telnet are disabled.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

There are two XML agents: a legacy XML agent and an enhanced-performance XML agent. We recommend that you use the enhanced-performance agent. The legacy agent is supported for backward compatibility. The **xml agent tty** command enables the legacy XML agent. Use the **xml agent** command to enable the enhanced-performance XML agent.

Use the **no** form of the **xml agent tty** command to disable the legacy XML agent.

Task ID

Task ID	Operations
config-services	read, write

This example shows how to enable XML requests over Secure Shell (SSH) and Telnet:

```
RP/0/RP0/CPU0:router(config)# xml agent tty
```



Network Configuration Protocol Commands

- clear netconf-yang agent rate-limit, on page 218
- clear netconf-yang agent session, on page 219
- netconf-yang agent rate-limit, on page 220
- netconf-yang agent session, on page 221
- netconf-yang agent ssh , on page 222
- netconf-yang agent yfw idle-timeout , on page 223
- show gribi aft, on page 224
- show netconf-yang clients, on page 226
- show netconf-yang rate-limit, on page 227
- show netconf-yang statistics, on page 228
- ssh server capability netconf-xml, on page 230
- ssh server netconf port, on page 231

clear netconf-yang agent rate-limit

To clear the set rate-limit statistics, use the **clear netconf-yang agent rate-limit** command in the appropriate mode.

clear netconf-yang agent rate-limit

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	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 example shows how to use the **clear netconf-yang agent rate-limit** command:

```
RP/0/RP0/CPU0:router # clear netconf-yang agent rate-limit
```

clear netconf-yang agent session

To clear the specified netconf agent session, use the **clear netconf-yang agent session** in EXEC mode.

clear netconf-yang agent session *session-id*

Syntax Description

session-id	The session-id which needs to be cleared.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The **show netconf-yang clients** command can be used to get the required session-id(s).

Task ID

Task ID	Operation
config-services	read, write

Example

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

```
RP/0/RP0/CPU0:router (config) #  clear netconf-yang agent session 32125
```

netconf-yang agent rate-limit

To set the rate-limit for the netconf yang agent, use the **netconf-yang agent rate-limit** command in the appropriate mode. To delete the set rate-limit, use the **no** form of the command.

netconf-yang agent rate-limit *bytes*
no netconf-yang agent rate-limit *bytes*

Syntax Description

bytes	The number of bytes to process per second. Range is 4096-4294967295. It is based on the size of the request(s) from the client to the netconf server.

Command Default

By default, no limit is set

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Use the **show netconf-yang rate-limit** command to check if the set limit is adequate.

Task ID

Task ID	Operation
config-services	read, write

Example

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

```
RP/0/RP0/CPU0:router # netconf-yang agent rate-limit 5000
```

netconf-yang agent session

To set the session details (limits and timeouts) for a netconf-yang agent, use the **netconf-yang agent session** command in the appropriate mode. To remove the configured session limits and timeouts, use the **no** form of the command.

netconf-yang agent session { **limit** *value* | **absolute-timeout** *value* | **idle-timeout** *value* }
no netconf-yang agent session { **limit** *value* | **absolute-timeout** *value* | **idle-timeout** *value* }

Syntax Description

limit *value*	Sets the maximum count for concurrent netconf-yang sessions. Range is 1 to 1024.
absolute-timeout *value*	Enables session absolute timeout and sets the absolute session lifetime. Range is 1 to 1440. Unit is minutes.
idle-timeout *value*	Enables session idle timeout and sets the idle session lifetime. Range is 1 to 1440. Unit is minutes.

Command Default

By default, no limits are set

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	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 session** command:

```
RP/0/RP0/CPU0:router (config) #   netconf-yang agent session limit
```

netconf-yang agent ssh

To enable netconf agent over SSH (Secure Shell) , use the **netconf-yang agent ssh** command in the global configuration mode. To disable netconf, use the **no** form of the command.

netconf-yang agent ssh
no netconf-yang agent ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SSH is currently the supported transport method for Netconf.

Task ID

Task ID	Operation
config-services	read, write

Example

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

```
RP/0/RP0/CPU0:router (config) #  netconf-yang agent ssh
```

netconf-yang agent yfw idle-timeout

To configure idle timeout value for the operational yang model use the **netconf-yang agent yfw idle-timeout** command. Idle timeout indicates the duration for which there is no netconf process activity. If the idle timeout value is configured, all the operational yang models that are not being used for the specified duration, are released from the memory.

netconf-yang agent yfw idle-timeout *time in seconds*

Syntax Description

Specify the time in seconds. The valid value must be between the range of 1 to 4294967295 seconds

Command Default

If this command is not configured, the operational yang models are not released from the memory. To manually release the yang models, the Netconf process should be restarted.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Example

This example shows how to use the **netconf-yang agent yfw idle-timeout** command:

```
RP/0/RP0/CPU0:router (config) #  netconf-yang agent yfw idle-timeout 60
```

show gribi aft

To display Routing Information Base (RIB) data modified using gRPC Routing Information Base Interface (gRIBI) RPCs, use the **show gribi aft** command in XR EXEC mode.

show gribi aft { **next-hops** | | **next-hop-groups** | | **ipv4-unicast** } **vrf all**

Syntax Description

next-hops	Specifies registered next-hop notification addresses.
next-hop-groups	Specifies registered next-hop-groups notification addresses.
ipv4-unicast	Specifies IPv4 unicast address prefixes.
vrf all	Specifies all the VRF associated with the source interface.

Command Default

No default behavior or values.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.9.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
ipv4	read

Examples

The following example illustrates the **show gribi aft next-hops** command to view the registered next hop entries:

```
Router#show gribi aft next-hops
Thu Feb 02 17:01:19.548 UTC
100: 192.0.2.40
200: 192.0.2.42
1000: 192.0.2.6
1100: 192.0.2.10
1111: (vrf REPAIR)
1200: 192.0.2.14
2000: 192.0.2.18
2100: 192.0.2.22
3000: 192.0.2.26
4000: Decapsulate IPv4(vrf DEFAULT
```

The following example shows the **show gribi aft next-hop-groups** command to view the registered next hop group entries:

```
Router#show gribi aft next-hop-groups
Thu Feb 02 17:01:24.736 UTC
```

```
100, Backup NHG: 1111
  [100, 2]: 192.0.2.40
  [200, 2]: 192.0.2.42
  [1111, 100]: (vrf REPAIR) (!)
1000
  [1100, 30]: 192.0.2.10
  [1200, 10]: 192.0.2.14
  [1000, 60]: 192.0.2.6
1111
  [1111, 100]: (vrf REPAIR)
2000
  [2000, 50]: 192.0.2.18
  [2100, 50]: 192.0.2.22
3000
  [3000, 10]: 192.0.2.26
4000
  [4000, 10]: Decapsulate IPv4(vrf DEFAULT)
```

The following example shows the **show gribi aft ipv4-unicast** command to view the IPv4 address family configured in the RIB:

```
Router#show gribi aft ipv4-unicast vrf all
Thu Feb 02 17:01:24.736 UTC
VRF: DEFAULT
10.1.0.1/22 via NHG 3000
192.0.2.40/22 via NHG 1000
192.0.2.42/22 via NHG 2000
```

show netconf-yang clients

To display the client details for netconf-yang, use the **show netconf-yang clients** command in EXEC mode.

show netconf-yang clients

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang clients** command:

```
RP/0/RP0/CPU0:router (config) #  sh netconf-yang clients
Netconf clients
client session ID|   NC version|    client connect time|        last OP time|        last
OP type|    <lock>|
 22969|                    1.1|          0d  0h  0m  2s|            11:11:24|
close-session|        No|
 15389|                    1.1|          0d  0h  0m  1s|            11:11:25|
get-config|        No|
```

Table 19: Field descriptions

Field name	Description
Client session ID	Assigned session identifier
NC version	Version of the Netconf client as advertised in the hello message
Client connection time	Time elapsed since the client was connected
Last OP time	Last operation time
Last OP type	Last operation type
Lock (yes or no)	To check if the session holds a lock on the configuration datastore

show netconf-yang rate-limit

To display the statistics of the total data dropped , due to the set rate-limit, use the **show netconf-yang rate-limit** command in the appropriate mode.

show netconf-yang rate-limit

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang rate-limit** command:

```
RP/0/RP0/CPU0:router # show netconf-yang rate-limit
rate-limit statistics
Total data dropped: 0 Bytes
```

show netconf-yang statistics

To display the statistical details for netconf-yang, use the **show netconf-yang statistics** command in EXEC mode.

show netconf-yang statistics

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

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

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang statistics** command:

```
RP/0/RP0/CPU0:router (config) #  sh netconf-yang statistics
Summary statistics
                        # requests|             total time|   min time per request|   max
 time per request|   avg time per request|
other                             0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
close-session                     4|       0h  0m  0s   3ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
kill-session                      0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-schema                        0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get                               0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-config                        1|       0h  0m  0s   1ms|       0h  0m  0s   1ms|
 0h  0m  0s   1ms|       0h  0m  0s   1ms|
edit-config                       3|       0h  0m  0s   2ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
commit                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
cancel-commit                     0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
lock                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
unlock                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
discard-changes                   0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
```

```
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
validate                          0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
xml parse                         8|       0h  0m  0s   4ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
netconf processor                 8|       0h  0m  0s   6ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
```

Table 20: Field descriptions

Field name	Description
Requests	Total number of processed requests of a given type
Total time	Total processing time of all requests of a given type
Min time per request	Minimum processing time for a request of a given type
Max time per request	Maximum processing time for a request of a given type
Avg time per request	Average processing time for a request type

ssh server capability netconf-xml

To enable NETCONF reach XML subsystem via port 22, use the **ssh server capability netconf-xml** command in in the XR Config mode. Use **no** form of this command to disable NETCONF reach XML subsystem.

ssh server capability netconf-xml

Syntax Description

This command has no keywords or arguments.

Command Default

Port 22 is the default port.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
crypto	read, write

ssh server netconf port

To configure a port for the netconf SSH server, use the **ssh server netconf port** command in the global configuration mode. To return to the default port, use the **no** form of the command.

ssh server netconf port *port number*
no ssh server netconf port*port number*

Syntax Description

port *port-number*	Port number for the netconf SSH server (default port number is 830).

Command Default

The default port number is 830.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must configure the **ssh server netconf** command for at least one VRF, in order to configure a netconf port to enable netconf subsystem support.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to use the ssh server netconf port command with port 831:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server netconf port 831
```

Related Commands

Command	Description
ssh server netconf	Configures the vrf(s), where netconf subsystem requests are to be received.
netconf-yang agent ssh	Configures the **ssh netconf-yang backend** for the netconf subsystem (Required to allow the system to service netconf-yang requests). For more information, see the *System Management Command Reference for Cisco 8000 Series Routers*.



Object Tracking Commands

- action, on page 234
- delay, on page 235
- interface (track), on page 236
- line-protocol track, on page 237
- object, on page 238
- route ipv4, on page 239
- show track, on page 240
- track, on page 241
- type line-protocol state, on page 242
- type list boolean, on page 243
- type route reachability, on page 245
- type rtr, on page 246
- vrf (track), on page 247

action

To configure a track to implement actions based on changes in the state of the track, use the **action** command in the track configuration mode. To delete the configuration of action tracking, use the **no** form of this command.

action {**track-up** | **track-down**} **error-disable interface** *interface-name* [**auto-recover**]
no action {**track-up** | **track-down**} **error-disable interface** *interface-name* [**auto-recover**]

Syntax Description

track-up	Configures action on the track when the track goes up.
track-down	Configures action on the track when the track goes down.
error-disable	Disables the specified interface when the track state changes.
interface *interface name*	Name of the interface to be disabled.
auto-recover	(Optional) Allows the interface that is error-disabled by object tracking to auto-recover. Autorecovery of the interface occurs when the track state changes to the pre-error-disabled state.

Command Default

No default behavior or values

Command Modes

Track configuration (config track)

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To use the **action** command, you must be in a user group that is associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Task ID

Task ID	Operation
sysmgr	read, write

This example shows how to configure the **action** command:

```
Router# configure
Router(config)# track t1
Router(config-track)# type route reachability route ipv4 192.2.0.1/24
Router(config)# action track-down error-disable interface GigabitEthernet 0/0/0/1 auto-recover
```

delay

To configure the delay, in seconds, before the track or interface state should be polled for a change in status, use the **delay** command in track configuration mode. To delete the configuration of delay tracking, use the **no** form of this command.

delay {**up** | **down**} *seconds*
no delay {**up** | **down**} [*seconds*]

Syntax Description

delay up	*seconds*	Sets delay of from 1 to 180 seconds before communication of up status of the tracked object or list of objects.
delay down	*seconds*	Sets delay of from 1 to 180 seconds before communication of down status of the tracked object or list of objects.

Command Default

No default behavior or values

Command Modes

Track configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **delay** command can be used in conjunction with all track types:

- type line-protocol state, on page 242
- type list boolean, on page 243
- type route reachability, on page 245

When using the **no** form of the command, the use of the *seconds* argument is optional.

Task ID

Task ID	Operations
sysmgr	read, write

The following example shows that the tracking process is configured to notify the network administrator that the interface should be polled for its up state in five-second intervals:

```
Router# configuration
Router(config)# track name1
Router(config-track)# delay up 5
```

interface (track)

To select an interface object type for tracking purposes, use the **interface** command in interface configuration mode. To delete the configuration of a track based on a particular interface object type, use the **no** form of this command.

interface *type interface-path-id*
no interface *type interface-path-id*

Syntax Description

type	(Optional) Interface type. For more information, use the question mark (?) online help function.
interface-path-id	(Optional) Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

No default behavior or values

Command Modes

Interface configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To access the **interface** command, you must be in line protocol tracking configuration submode.

For information about interface keywords, see *Interface and Hardware Component Command Reference for Cisco 8000 Series Routers*.

Task ID

Task ID	Operations
sysmgr	read, write

The following example shows the **interface** command in the context of object tracking:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track track12
RP/0/RP0/CPU0:router(config-track)# type line-protocol state
RP/0/RP0/CPU0:router(config-track-line-prot)# interface atm 0/2/0/0.1
```

line-protocol track

To associate a specific track with an IPsec or GRE interface object, use the **line-protocol track** command in interface configuration mode. To delete the association between the track and the IPsec or GRE interface object, use the **no** form of this command.

line-protocol track *object-name*
no line-protocol track *object-name*

Syntax Description

object-name	Name of object being tracked.

Command Default

No default behavior or values

Command Modes

Interface configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read, write

The following example shows how the **line-protocol track** command is used:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track PREFIX1
RP/0/RP0/CPU0:router(config-track)# type route reachability
RP/0/RP0/CPU0:router(config-track-route)# route ipv4 7.0.0.0/24
RP/0/RP0/CPU0:router(config-track-route)# interface service-ipsec 1
RP/0/RP0/CPU0:router(config-if)# vrf 1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 70.0.0.2 255.25.255.0
RP/0/RP0/CPU0:router(config-if)# line-protocol track PREFIX1
```

object

To configure an object for tracking, use the **object** command in list tracking configuration mode. To delete a previously configured track based on an object, use the **no** form of this command.

object *object-name* [**not**]
no object *object-name*

Syntax Description

object-name	Name of the object to be tracked.
not	(Optional) Deletes a preivously configured track based on whether an interface object is not up or down.

Command Default

No default behavior or values

Command Modes

List tracking configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To delete a previously configured track based on whether an interface object is *not* up or down, use the **not** keyword together with the **object** command in a list of tracked objects based on a Boolean expression.

The **object** command can be used only for a track based on a Boolean expression.

Task ID

Task ID	Operations
sysmgr	read, write

The following example shows how to configure an object, using the optional **not** keyword, in a tracked list of objects based on a Boolean calculation:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track connection100
RP/0/RP0/CPU0:router(config-track-list)# type list boolean and
RP/0/RP0/CPU0:router(config-track-list)# object obj3 no
```

route ipv4

To configure that an IP prefix and subnet mask should be used as the basis to track route reachability, use the **route ipv4** command in route tracking configuration mode. To remove this configuration, use the **no** form of the command.

route ipv4 *IP prefix and subnet mask*
no route ipv4

Syntax Description

IP prefix and subnet mask	Network and subnet mask; for example, 10.56.8.10/16.

Command Default

No default behavior or values

Command Modes

Route tracking configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The *IP prefix* and *subnet mask* arguments are optional for the **no** form of this command.

Task ID

Task ID	Operations
sysmgr	read, write

The following example displays use of the **route ipv4** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track track22
RP/0/RP0/CPU0:router(config-track)# type route reachability
RP/0/RP0/CPU0:router(config-track-route)# route ipv4 10.56.8.10/16
```

show track

To display information about objects that were tracked and to specify the format of the report, use the **show track** command in EXEC mode.

show track [*track-name* | **interface** | **ipv4 route**] [**brief**]

Syntax Description

track-name	(Optional) Name of track used for tracking objects; for example, track1.
brief	(Optional) Displays a single line of information related to the preceding argument or keyword.
interface	(Optional) Displays tracked interface objects.
ipv4 route	(Optional) Displays the tracked IPv4 route objects.

Command Default

No default behavior or values

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show track** command to display information about objects that are tracked by the tracking process. When no arguments or keywords are specified, information for all objects is displayed.

Task ID

Task ID	Operations
sysmgr	read

The following sample output illustrates use of the **show track** command:

```
RP/0/RP0/CPU0:router# show track Track_name3

  Track_name3
          List boolean and is DOWN
          1 change, last change 10:26:20 SJC Sun Aug 05 2007
                  object name2 not UP
                  object name1 UP
```

track

To initiate or identify a tracking process used to track the status of an object or list of objects, use the **track** command in global configuration mode. To remove the tracking process, use the **no** form of this command.

track *track-name*
no track *track-name*

Syntax Description

track *track-name*	Name of track used for tracking objects; for example, track1. **Note** Special characters are not allowed in a *track-name*.

Command Default

No default behavior or values

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When you use the **track** command, you enter track configuration mode.

Task ID

Task ID	Operations
sysmgr	read, write

This example shows that the tracking process is configured to notify the network administrator about the up state of the tracked object list every five seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track LIST2
RP/0/RP0/CPU0:router# track LIST2 delay up 5
```

type line-protocol state

To configure tracking of the line protocol state of an interface object, use the **type line-protocol** command in track configuration mode. To delete the configuration of line-protocol tracking, use the **no** form of this command.

type line-protocol state
no type line-protocol state

Command Default

No default behavior or values

Command Modes

Track configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **type line-protocol state** command can be used in conjunction with the **delay** command to configure the delay, in seconds, before the track or interface state should be polled for a change in its status.

The **type line-protocol state** command enters line-protocol tracking configuration mode.

Task ID

Task ID	Operations
sysmgr	read, write

This example shows how to use the **type line-protocol state** command:

```
Router# configure
Router(config)# track track12
Router(config-track)# type line-protocol state
```

type list boolean

To configure a tracked list of objects based on a Boolean calculation, use the **type list boolean** command in track configuration mode. To remove an object tracking list based on a Boolean calculation, use the **no** form of the command.

type list boolean {**and** | **or**}
no type list boolean {**and** | **or**}

Syntax Description

and	Specifies that the list is up if all objects are up, or down if one or more objects are down. For example, when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down.
or	Specifies that the list is up if at least one object is up. For example, when tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.

Command Default

No default behavior or values

Command Modes

Track configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **type list boolean** command enters the list tracking configuration mode, and can be used in conjunction with the **delay** command to configure the delay, in seconds, before the track or interface state should be polled for a change in its status.

To remove a track based on whether an interface object is *not* up or down, use the **not** keyword together with the **object** command as shown in the example that follows.

Task ID

Task ID	Operations
sysmgr	read, write

This example shows how to use the **type list boolean** command in creating a list of objects to be tracked:

```
Router# configure
Router(config)# track LIST2
Router(config-track)# type list boolean and
Router(config-track-list)# object IPSec1 not
Router(config-track-list)# object IPSec2
Router(config-track-list)# object PREFIX1
Router(config-track-list)# exit
Router(config)# track IPSec1
Router(config-track)# type line-protocol state
```

```
Router(config-track-line-prot)# interface tengige 0/0/0/3
Router(config-track-line-prot)# exit
Router(config-track)# track IPSec2
Router(config-track)# type line-protocol state
Router(config-track-line-prot)# interface ATM0/2/0.1
Router(config-track-line-prot)# exit
Router(config)# track PREFIX1
Router(config-track)# type route reachability
Router(config-track-route)# route ipv4 7.0.0.0/24
Router(config-track-route)# exit
Router(config-track)# interface service-ipsec 1
Router(config-if)# vrf 1
Router(config-if)# ipv4 address 70.0.0.2 255.255.255.0
Router(config-if)# profile vrf_1_ipsec
Router(config-if)# line-protocol track LIST2
Router(config-if)# tunnel source 80.0.0.2
Router(config-if)# tunnel destination 80.0.0.1
Router(config-if)# service-location preferred-active 0/2/0
Router(config-if)# commit
```

type route reachability

To configure the routing process to notify the tracking process when the route state changes due to a routing update, use the **type route reachability** command in track configuration mode. To remove a track based on route reachability, use the **no** form of this command.

type route reachability
no type route reachability

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values

Command Modes

Track configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

A tracked IP-route object is considered up and reachable when a routing-table entry exists for the route and the route is not inaccessible.

The **type route reachability** command can be used in conjunction with the **delay** command to configure the delay, in seconds, before the track or interface state should be polled for a change in its status.

The route reachability tracking process is based on either of the following, depending on your router type:

- **vrf**—A VRF table name.
- **route**—An IPv4 prefix consisting of the network and subnet mask (for example, 10.56.8.10/16).

Task ID

Task ID	Operations
sysmgr	read, write

This example shows how to track for route reachability:

```
Router# configure
Router(config)# track track22
Router(config-track)# type route reachability
```

type rtr

To configure the router to track the return code of IP service level agreement (SLA) operations, use the **type rtr** command in track configuration mode. To remove a track based on IP SLA return code, use the **no** form of this command.

type rtr *ipsla-no* **reachability**
no type rtr

Syntax Description

ipsla-no	IP SLA operation number. Values can range from 1 to 2048.
reachability	Tracks whether the route is reachable or not.

Command Default

None

Command Modes

Track configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **type rtr** command in conjunction with a configuration that uses:

- The

  track keyword in the **permit** command within an ACL definition. For example:

```
ipv4 access-list abf-track
  10 permit any any nexthop track track1 1.2.3.4
```

- An IP service level agreement configuration.

Task ID

Task ID	Operation
sysmgr	read, write

This example shows how to configure IPSLA object tracking:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track track22
RP/0/RP0/CPU0:router(config-track)# type rtr 1 reachability
```

vrf (track)

To configure a VRF table to be used as the basis to track route reachability, use the **vrf** command in route tracking configuration mode. To delete the configuration of a VRF table for the purpose of IP route tracking purposes, use the **no** form of the command.

vrf *vrf-table-name*
no vrf [*vrf-table-name*]

Syntax Description

vrf-table-name	Network and subnet; for example, 10.56.8.10/16.

Command Default

No default behavior or values

Command Modes

Route tracking configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
sysmgr	read, write

The following example displays the use of the **vrf** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# track track22
RP/0/RP0/CPU0:router(config-track)# type route reachability
RP/0/RP0/CPU0:router(config-track-route)# vrf vrf1
```



Process Memory Management Commands

- clear context, on page 250
- dumpcore, on page 251
- exception disk-usage-limit, on page 254
- exception filepath, on page 255
- follow, on page 259
- process, on page 266
- process core, on page 268
- process mandatory, on page 270
- show context, on page 272
- show memory, on page 274
- show memory compare, on page 277
- show memory heap, on page 280
- show processes, on page 284

clear context

To clear core dump context information, use the **clear context** command in the appropriate mode.

clear context location {*node-id* | **all**}

Syntax Description

location {*node-id* | **all**}	(Optional) Clears core dump context information for a specified node. The *node-id* argument is expressed in the *rack/slot* notation. Use the **all** keyword to indicate all nodes.

Command Default

No default behavior or values

Command Modes

Administration EXEC

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **clear context** command to clear core dump context information. If you do not specify a node with the **location** *node-id* keyword and argument, this command clears core dump context information for all nodes.

Use the **show context** command to display core dump context information.

Task ID

Task ID	Operations
diag	execute

The following example shows how to clear core dump context information:

```
RP/0/RP0/CPU0:router# clear context
```

dumpcore

To manually generate a core dump, use the **dumpcore** command in XR EXEC mode .

dumpcore {**running** | **suspended**} *job-id* **location** *node-id*

Syntax Description

running	Generates a core dump for a running process.
suspended	Suspends a process, generates a core dump for the process, and resumes the process.
job-id	Process instance identifier.
location *node-id*	Generates a core dump for a process running on the specified node. The *node-id* argument is expressed in the *rack/slot* notation.

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When a process crashes on the Cisco IOS XR software, a core dump file of the event is written to a designated destination without bringing down the router. Upon receiving notification that a process has terminated abnormally, the Cisco IOS XR software then respawns the crashed process. Core dump files are used by Cisco Technical Support Center engineers and development engineers to debug the Cisco IOS XR software.

Core dumps can be generated manually for a process, even when a process has not crashed. Two modes exist to generate a core dump manually:

- **running** —Generates a core dump for a running process. This mode can be used to generate a core dump on a critical process (a process whose suspension could have a negative impact on the performance of the router) because the core dump file is generated independently, that is, the process continues to run as the core dump file is being generated.
- **suspended** —Suspends a process, generates a core dump for the process, and resumes the process. Whenever the process is suspended, this mode ensures data consistency in the core dump file.

Core dump files contain the following information about a crashed process:

- Register information
- Thread status information
- Process status information
- Selected memory segments

Task ID

Task ID	Operations
diag	read, write

The following example shows how to generate a core dump in suspended mode for the process instance 52:

```
RP/0/RP0/CPU0:router# dumpcore suspended 52

RP/0/RP0/CPU0:Sep 22 01:40:26.982 : sysmgr[71]: process in stop/continue state 4104
RP/0/RP0/CPU0Sep 22 01:40:26.989 : dumper[54]: %DUMPER-4-CORE_INFO : Core for pid = 4104
 (pkg/bin/devc-conaux) requested by pkg/bin/dumper_gen@node0_RP0_CPU0
RP/0/RP0/CPU0Sep 22 01:40:26.993 : dumper[54]: %DUMPER-6-SPARSE_CORE_DUMP :
 Sparse core dump as configured dump sparse for all
RP/0/RP0/CPU0Sep 22 01:40:26.995 : dumper[54]: %DUMPER-7-DLL_INFO_HEAD : DLL path
 Text addr.  Text size   Data addr.  Data size   Version
RP/0/RP0/CPU0Sep 22 01:40:26.996 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libplatform.dll  0xfc0d5000  0x0000a914  0xfc0e0000  0x00002000        0
RP/0/RP0/CPU0Sep 22 01:40:26.996 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libsysmgr.dll  0xfc0e2000  0x0000ab48  0xfc0c295c  0x00000368        0
RP/0/RP0/CPU0Sep 22 01:40:26.997 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libinfra.dll  0xfc0ed000  0x00032de0  0xfc120000  0x00000c90        0
RP/0/RP0/CPU0Sep 22 01:40:26.997 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libios.dll  0xfc121000  0x0002c4bc  0xfc14e000  0x00002000        0
RP/0/RP0/CPU0Sep 22 01:40:26.997 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libc.dll  0xfc150000  0x00077ae0  0xfc1c8000  0x00002000        0
RP/0/RP0/CPU0Sep 22 01:40:26.998 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libsyslog.dll  0xfc1d2000  0x0000530c  0xfc120c90  0x00000308        0
RP/0/RP0/CPU0Sep 22 01:40:26.998 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libbackplane.dll  0xfc1d8000  0x0000134c  0xfc0c2e4c  0x000000a8        0
RP/0/RP0/CPU0Sep 22 01:40:26.999 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libnodeid.dll  0xfc1e5000  0x00009114  0xfc1e41a8  0x00000208        0
RP/0/RP0/CPU0Sep 22 01:40:26.999 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttyserver.dll  0xfc1f1000  0x0003dfcc  0xfc22f000  0x00002000        0
RP/0/RP0/CPU0Sep 22 01:40:27.000 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttytrace.dll  0xfc236000  0x00004024  0xfc1e44b8  0x000001c8        0
RP/0/RP0/CPU0Sep 22 01:40:27.000 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libdebug.dll  0xfc23b000  0x0000ef64  0xfc1e4680  0x00000550        0
RP/0/RP0/CPU0Sep 22 01:40:27.001 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/lib_procfs_util.dll  0xfc24a000  0x00004e2c  0xfc1e4bd0  0x000002a8        0
RP/0/RP0/CPU0Sep 22 01:40:27.001 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libsysdb.dll  0xfc24f000  0x000452e0  0xfc295000  0x00000758        0
RP/0/RP0/CPU0Sep 22 01:40:27.001 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libsysdbutils.dll  0xfc296000  0x0000ae08  0xfc295758  0x000003ec        0
RP/0/RP0/CPU0Sep 22 01:40:27.002 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/lib_tty_svr_error.dll  0xfc2a1000  0x0000172c  0xfc1e4e78  0x00000088        0
RP/0/RP0/CPU0Sep 22 01:40:27.002 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/lib_tty_error.dll  0xfc2a3000  0x00001610  0xfc1e4f00  0x00000088        0
RP/0/RP0/CPU0Sep 22 01:40:27.003 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libwd_evm.dll  0xfc2a5000  0x0000481c  0xfc295b44  0x00000188        0
RP/0/RP0/CPU0Sep 22 01:40:27.003 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttydb.dll  0xfc2aa000  0x000051dc  0xfc295ccc  0x00000188        0
RP/0/RP0/CPU0Sep 22 01:40:27.004 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttydb_error.dll  0xfc23a024  0x00000f0c  0xfc295e54  0x00000088        0
RP/0/RP0/CPU0Sep 22 01:40:27.004 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/librs232.dll  0xfc2b0000  0x00009c28  0xfc2ba000  0x00000470        0
RP/0/RP0/CPU0Sep 22 01:40:27.005 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/lib_rs232_error.dll  0xfc2bb000  0x00000f8c  0xfc295edc  0x00000088        0
RP/0/RP0/CPU0Sep 22 01:40:27.005 : dumper[54]: %DUMPER-7-DLL_INFO :
```

```
 /pkg/lib/libst16550.dll  0xfc2bc000  0x00008ed4  0xfc2ba470  0x00000430        0
RP/0/RP0/CPU0Sep 22 01:40:27.006 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libconaux.dll  0xfc2c5000  0x00001dc0  0xfc2ba8a0  0x000001a8        0
RP/0/RP0/CPU0Sep 22 01:40:27.006 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/lib_conaux_error.dll  0xfc1ee114  0x00000e78  0xfc295f64  0x00000088        0
RP/0/RP0/CPU0Sep 22 01:40:27.007 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttyutil.dll  0xfc2c7000  0x00003078  0xfc2baa48  0x00000168        0
RP/0/RP0/CPU0Sep 22 01:40:27.007 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libbag.dll  0xfc431000  0x0000ee98  0xfc40cc94  0x00000368        0
RP/0/RP0/CPU0Sep 22 01:40:27.008 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libchkpt.dll  0xfc474000  0x0002ecf8  0xfc4a3000  0x00000950        0
RP/0/RP0/CPU0Sep 22 01:40:27.008 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libsysdbbackend.dll  0xfc8ed000  0x0000997c  0xfc8d3aa8  0x0000028c        0
RP/0/RP0/CPU0Sep 22 01:40:27.008 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttymgmtconnection.dll  0xfce85000  0x00004208  0xfce8a000  0x00000468
0
RP/0/RP0/CPU0Sep 22 01:40:27.009 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttymgmt.dll  0xfcea4000  0x0000e944  0xfce8abf0  0x000003c8        0
RP/0/RP0/CPU0Sep 22 01:40:27.009 : dumper[54]: %DUMPER-7-DLL_INFO :
 /pkg/lib/libttynmspc.dll  0xfcec7000  0x00004a70  0xfcec6644  0x000002c8        0
RP/0/RP0/CPU0Sep 22 01:40:28.396 : dumper[54]: %DUMPER-5-CORE_FILE_NAME :
 Core for process pkg/bin/devc-conaux at harddisk:/coredump/devc-conaux.by.
 dumper_gen.sparse.20040922-014027.node0_RP0_CPU0.ppc.Z
RP/0/RP0/CPU0Sep 22 01:40:32.309 : dumper[54]: %DUMPER-5-DUMP_SUCCESS : Core dump success
```

exception disk-usage-limit

To set the core dump folder `/misc/disk1/coredumps` storage limit, use the **exception disk-usage-limit** command in the appropriate configuration mode.

exception disk-usage-limit <*usage limit percent*>

Syntax Description

disk-usage-limit <*usage limit percent*>	Set the core dump folder storage limit between 20 to 50 percent.

Command Default

None

Command Modes

Global configuration

XR Config

Command History

Release	Modification
Release 25.2.1	The keyword **disk-usage-limit** was introduced.
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **exception disk-usage-limit** command to modify core dump folder storage limit.

Task ID

Task ID	Operations
diag	read, write

This example displays the core dump folder storage limit set to 29 percent of the total hard disk space.

```
Router# configuration
Router(config)# exception disk-usage-limit 29
Router(config)# commit
```

exception filepath

To modify core dump settings, use the **exception filepath** command in the appropriate configuration mode. To remove the configuration, use the **no** form of this command.

exception [**choice** *preference*] [**compress** {**on** | **off**}] **filename** *filename lower-limit−higher-limit* **filepath** *filepath-name*
no exception [**choice** *preference*] [**compress** {**on** | **off**}] **filename** *filename lower-limit−higher-limit* **filepath** *filepath-name*

Syntax Description

choice *preference*	(Optional) Configures the order of preference for the destination of core dump files. Up to the three destinations can be defined. Valid values are 1 to 3.
compress {**on** | **off**}	(Optional) Specifies whether or not the core dump file should be sent compressed. By default, core dump files are sent compressed. If you specify the **compress** keyword, you must specify one of the following required keywords: • **on** —Compresses the core dump file before sending it. • **off** —Does not compress the core dump file before sending it.
filename *filename lower-limit-higher-limit*	(Optional) Specifies the filename to be appended to core dump files and the lower and higher limit range of core dump files to be sent to a specified destination before being recycled by the circular buffer. **filename** *filename lower-limit-higher-limit* See exception filepath, on page 255 for a description of the default core dump file naming convention. Vali **filename** *filename lower-limit-higher-limit* d values for the *lower-limit* argument are 0 to 4. Valid values for the *higher-limit* argument are 5 to 64. A hyphen (−) must immediately follow the *lower-limit* argument. **Note** To uniquely identify each core dump file, a value is appended to each core dump file, beginning with the lower limit value configured for the *lower-limit* argument and continuing until the higher limit value configured for the *higher-limit* argument has been reached. After the higher limit value has been reached, the Cisco IOS XR software begins to recycle the values appended to core dump files, beginning with the lower limit value.
filepath-name	Local file system or network protocol, followed by the directory path. All local file systems are supported. The following network protocols are supported: TFTP and FTP.

Command Default

If you do not specify the order of preference for the destination of core dump files using the **choice** *preference* keyword and argument, the default preference is the primary location or 1.

Core dump files are sent compressed.

Command Modes

Global configuration

XR Config

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **exception filepath** command to modify core dump settings, such as the destination file path to store core dump files, file compression, and the filename appended to core dumps.

Up to three user-defined locations may be configured as the preferred destinations for core dump files:

- Primary location—The primary destination for core dump files. Enter the **choice** keyword and a value of **1** (that is, **choice 1**) for the *preference* argument to specify a destination as the primary location for core dump files.
- Secondary location—The secondary fallback choice for the destination for core dump files, if the primary location is unavailable (for example, if the hard disk is set as the primary location and the hard disk fails). Enter the **choice** keyword and a value of **2** (that is, **choice 2**) for the *preference* argument to specify a destination as the secondary location for core dump files.
- Tertiary location—The tertiary fallback choice as the destination for core dump files, if the primary and secondary locations fail. Enter the **choice** keyword and a value of 3 (that is, **choice 3**) for the *preference* argument to specify a destination as the tertiary location for core dump files.

When specifying a destination for a core dump file, you can specify an absolute file path on a local file system or on a network server. The following network protocols are supported: TFTP and FTP.

Note We recommend that you specify a location on the hard disk as the primary location.

In addition to the three preferred destinations that can be configured, Cisco IOS XR software provides three default fallback destinations for core dump files in the event that user-defined locations are unavailable.

The default fallback destinations are:

- harddisk:/dumper
- disk1:/dumper
- disk0:/dumper

Note If a default destination is a boot device, the core dump file is not sent to that destination.

We recommend that you configure at least one preferred destination for core dump files as a preventive measure if the default fallback paths are unavailable. Configuring at least one preferred destination also ensures that core dump files are archived because the default fallback destinations store only the first and last core dump files for a crashed process.

Note Cisco IOS XR software does not save a core file on a local storage device if the size of the core dump file creates a low-memory condition.

By default, Cisco IOS XR software assigns filenames to core dump files according to the following format:

process [.by. *requester* |.abort][.sparse]. *date-time* . *node* . *processor-type* [.Z]

For example:

```
packet.by.dumper_gen.20040921-024800.node0_RP0_CPU0.ppc.Z
```

Table 21: Default Core Dump File Naming Convention Description

Field	Description
process	Name of the process that generated the core dump.
.by. *requester* | .abort	If the core dump was generated because of a request by a process (requester), the core filename contains the string ".by.*requester*" where the *requester* variable is the name or process ID (PID) of the process that requested the core dump. If the core dump was due to a self-generated abort call request, the core filename contains the string ".abort" instead of the name of the requester.
.sparse	If a sparse core dump was generated instead of a full core dump, "sparse" appears in the core dump filename.
.date-time	Date and time the dumper process was called by the process manager to generate the core dump. The *.date-time* time-stamp variable is expressed in the *yyyy.mm.dd-hh.mm.ss* format. Including the time stamp in the filename uniquely identifies the core dump filename.
. node	Node ID, expressed in the *rack/module* notation, where the process that generated the core dump was running.
.processor-type	Type of processor (mips or ppc).
.Z	If the core dump was sent compressed, the filename contains the .Z suffix.

You can modify the default naming convention by specifying a filename to be appended to core dump files with the optional **filename** *filename* keyword and argument and by specifying a lower and higher limit ranges of values to be appended to core dump filenames with the *lower-limit* and *higher-limit* arguments, respectively. The filename that you specify for the *filename* argument is appended to the core dump file and the lower and higher limit ranges of core dump files to be sent to a specified destination before the filenames are recycled. Valid values for the *lower-limit* argument are 0 to 4. Valid values for the *higher-limit* argument are 5 to 64. A hyphen (-) must immediately follow the *lower-limit* argument. In addition, to uniquely identify each core dump file, a value is appended to each core dump file, beginning with the lower-limit value specified with the *lower-limit* argument and continuing until the higher-limit value specified with the *higher-limit* argument has been reached. When the configured higher-limit value has been reached, Cisco IOS XR software begins to recycle the values appended to core dump files, beginning with the lower-limit value.

Task ID

Task ID	Operations
diag	read, write

The following example shows how to configure the core dump setting for the primary user-defined preferred location. In this example, core files are configured to be sent uncompressed; the filename of core dump files is set to “core” (that is, all core filenames will be named core); the range value is set from 0 to 5 (that is, the values 0 to 5 are appended to the filename for the first five generated core dump files, respectively, before being recycled); and the destination is set to a directory on the hard disk.

```
RP/0/RP0/CPU0:router(config)# exception choice 1 compress off
  filename core 0-5 filepath /harddisk:/corefile
```

follow

To unobtrusively debug a live process or a live thread in a process, use the **follow** command in XR EXEC mode.

follow {**job** *job-id* | **process** *pid* | **location** *node-id*} [**all**] [**blocked**] [**debug** *level*] [**delay** *seconds*] [**dump** *address size*] [**iteration** *count*] [**priority** *level*] [**stackonly**] [**thread** *tid*] [**verbose**]

Syntax Description

job *job-id*	Follows a process by job ID.
process *pid*	Follows the process with the process ID (PID) specified for the *pid* argument.
location *node-id*	Follows the target process on the designated node. The *node-id* argument is expressed in the *rack/slot* notation.
all	(Optional) Follows all threads.
blocked	(Optional) Follows the chain of thread IDs (TIDs) or PIDs that are blocking the target process.
debug *level*	(Optional) Sets the debug level for the following operation. Valid values for the level argument are 0 to 10.
delay *seconds*	(Optional) Sets the delay interval between each iteration. Valid values for the *seconds* argument are 0 to 255 seconds.
dump *address size*	(Optional) Dumps the memory segment starting with the specified memory address and size specified for the *address* and *size* arguments.
iteration *count*	(Optional) Specifies the number of times to display information. Valid values for the *count* argument are 0 to 255 iterations.
priority *level*	(Optional) Sets the priority level for the following operation. Valid values for the *level* argument are 1 to 63.
stackonly	(Optional) Displays only stack trace information.
thread *tid*	(Optional) Follows the TID of a process or job ID specified for the *tid* argument.
verbose	(Optional) Displays register and status information pertaining to the target process.

Command Default

Entering the **follow** command without any optional keywords or arguments performs the operation for five iterations from the local node with a delay of 5 seconds between each iteration. The output includes information about all live threads. This command uses the default scheduling priority from where the command is being run.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use this command to unintrusively debug a live process or a live thread in a process. This command is particularly useful for debugging deadlock and livelock conditions, for examining the contents of a memory location or a variable in a process to determine the cause of a corruption issue, or in investigating issues where a thread is stuck spinning in a loop. A livelock condition is one that occurs when two or more processes continually change their state in response to changes in the other processes.

The following actions can be specified with this command:

- Follow all live threads of a given process or a given thread of a process and print stack trace in a format similar to core dump output.
- Follow a process in a loop for a given number of iterations.
- Set a delay between two iterations while invoking the command.
- Set the priority at which this process should run while this command is being run.
- Dump memory from a given virtual memory location for a given size.
- Display register values and status information of the target process.

Take a snapshot of the execution path of a thread asynchronously to investigate performance-related issues by specifying a high number of iterations with a zero delay.

Task ID

Task ID	Operations
basic-services	read

The following example shows how to use the **follow** command to debug the process associated with job ID 257 for one iteration:

```
RP/0/RP0/CPU0:router# follow job 257 iteration 1

Attaching to process pid = 28703 (pkg/bin/packet)
No tid specified, following all threads

DLL Loaded by this process
------------------------------

DLL path                 Text addr. Text size  Data addr. Data size  Version
/pkg/lib/libovl.dll      0xfc0c9000 0x0000c398 0xfc0c31f0 0x0000076c        0
/pkg/lib/libplatform.dll 0xfc0d6000 0x0000aa88 0xfc0e1000 0x00002000        0
/pkg/lib/libsysmgr.dll   0xfc0e3000 0x0000aeac 0xfc0c395c 0x00000388        0
/pkg/lib/libinfra.dll    0xfc0ee000 0x000332ec 0xfc122000 0x00000c70        0
/pkg/lib/libios.dll      0xfc123000 0x0002c4bc 0xfc150000 0x00002000        0
/pkg/lib/libc.dll        0xfc152000 0x00077ae0 0xfc1ca000 0x00002000        0
/pkg/lib/libsyslog.dll   0xfc1d4000 0x0000530c 0xfc122c70 0x00000308        0
/pkg/lib/libbackplane.dll 0xfc1da000 0x0000134c 0xfc0c3e6c 0x000000a8        0
/pkg/lib/libnodeid.dll   0xfc1e7000 0x000091fc 0xfc1e61a8 0x00000208        0
/pkg/lib/libdebug.dll    0xfc23e000 0x0000ef64 0xfc1e6680 0x00000550        0
/pkg/lib/lib_procfs_util.dll 0xfc24d000 0x00004e2c 0xfc1e6bd0 0x000002a8        0
/pkg/lib/libsysdb.dll    0xfc252000 0x00046224 0xfc299000 0x0000079c        0
```

```
/pkg/lib/libsysdbutils.dll 0xfc29a000 0x0000ae04 0xfc29979c 0x000003ec        0
/pkg/lib/libwd_evm.dll   0xfc2a9000 0x0000481c 0xfc299b88 0x00000188        0
/pkg/lib/lib_mutex_monitor.dll 0xfc35e000 0x00002414 0xfc340850 0x00000128        0
/pkg/lib/libchkpt.dll    0xfc477000 0x0002ee04 0xfc474388 0x00000950        0
/pkg/lib/libpacket_common.dll 0xfc617000 0x000130f0 0xfc6056a0 0x000007b0        0

Iteration 1 of 1
-------------------------------

Current process = "pkg/bin/packet", PID = 28703 TID = 1

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x48204410 [<N/A>]

ENDOFSTACKTRACE

Current process = "pkg/bin/packet", PID = 28703 TID = 2

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0xfc48d848 [chk_evm_thread]

ENDOFSTACKTRACE

Current process = "pkg/bin/packet", PID = 28703 TID = 3

trace_back: #0 0xfc17d54c [SignalWaitinfo]
trace_back: #1 0xfc161c64 [sigwaitinfo]
trace_back: #2 0xfc10302c [event_signal_thread]

ENDOFSTACKTRACE

Current process = "pkg/bin/packet", PID = 28703 TID = 4

trace_back: #0 0xfc1106c4 [MsgReceivePulse]
trace_back: #1 0xfc0fc604 [msg_receive_async]
trace_back: #2 0xfc0ffa70 [event_dispatch]
trace_back: #3 0xfc0ffc5c [event_block_async]
trace_back: #4 0xfc35e36c [receive_events]

ENDOFSTACKTRACE

Current process = "pkg/bin/packet", PID = 28703 TID = 5

trace_back: #0 0xfc17d564 [SignalWaitinfo_r]
trace_back: #1 0xfc161c28 [sigwait]
trace_back: #2 0x48203928 [<N/A>]

ENDOFSTACKTRACE
```

The following example shows how to use the **follow** command to debug TID 5 of the process associated with job ID 257 for one iteration:

```
RP/0/RP0/CPU0:router# follow job 257 iteration 1 thread 5
```

```
Attaching to process pid = 28703 (pkg/bin/packet)

DLL Loaded by this process
------------------------------

DLL path                 Text addr. Text size  Data addr. Data size  Version
/pkg/lib/libovl.dll      0xfc0c9000 0x0000c398 0xfc0c31f0 0x0000076c        0
/pkg/lib/libplatform.dll 0xfc0d6000 0x0000aa88 0xfc0e1000 0x00002000        0
/pkg/lib/libsysmgr.dll   0xfc0e3000 0x0000aeac 0xfc0c395c 0x00000388        0
/pkg/lib/libinfra.dll    0xfc0ee000 0x000332ec 0xfc122000 0x00000c70        0
/pkg/lib/libios.dll      0xfc123000 0x0002c4bc 0xfc150000 0x00002000        0
/pkg/lib/libc.dll        0xfc152000 0x00077ae0 0xfc1ca000 0x00002000        0
/pkg/lib/libsyslog.dll   0xfc1d4000 0x0000530c 0xfc122c70 0x00000308        0
/pkg/lib/libbackplane.dll 0xfc1da000 0x0000134c 0xfc0c3e6c 0x000000a8        0
/pkg/lib/libnodeid.dll   0xfc1e7000 0x000091fc 0xfc1e61a8 0x00000208        0
/pkg/lib/libdebug.dll    0xfc23e000 0x0000ef64 0xfc1e6680 0x00000550        0
/pkg/lib/lib_procfs_util.dll 0xfc24d000 0x00004e2c 0xfc1e6bd0 0x000002a8        0
/pkg/lib/libsysdb.dll    0xfc252000 0x00046224 0xfc299000 0x0000079c        0
/pkg/lib/libsysdbutils.dll 0xfc29a000 0x0000ae04 0xfc29979c 0x000003ec        0
/pkg/lib/libwd_evm.dll   0xfc2a9000 0x0000481c 0xfc299b88 0x00000188        0
/pkg/lib/lib_mutex_monitor.dll 0xfc35e000 0x00002414 0xfc340850 0x00000128        0
/pkg/lib/libchkpt.dll    0xfc477000 0x0002ee04 0xfc474388 0x00000950        0
/pkg/lib/libpacket_common.dll 0xfc617000 0x000130f0 0xfc6056a0 0x000007b0        0

Iteration 1 of 1
------------------------------

Current process = "pkg/bin/packet", PID = 28703 TID = 5

trace_back: #0 0xfc17d564 [SignalWaitinfo_r]
trace_back: #1 0xfc161c28 [sigwait]
trace_back: #2 0x48203928 [<N/A>]

ENDOFSTACKTRACE
```

The following example shows how to use the **follow** command to debug the chain of threads blocking thread 2 associated with the process assigned PID 139406:

```
RP/0/RP0/CPU0:router# follow process 139406 blocked iteration 1 thread 2

Attaching to process pid = 139406 (pkg/bin/lpts_fm)

DLL Loaded by this process
------------------------------

DLL path                 Text addr. Text size  Data addr. Data size  Version
/pkg/lib/libplatform.dll 0xfc0d6000 0x0000aa88 0xfc0e1000 0x00002000        0
/pkg/lib/libsysmgr.dll   0xfc0e3000 0x0000aeac 0xfc0c395c 0x00000388        0
/pkg/lib/libinfra.dll    0xfc0ee000 0x000332ec 0xfc122000 0x00000c70        0
/pkg/lib/libios.dll      0xfc123000 0x0002c4bc 0xfc150000 0x00002000        0
/pkg/lib/libc.dll        0xfc152000 0x00077ae0 0xfc1ca000 0x00002000        0
/pkg/lib/libltrace.dll   0xfc1cc000 0x00007f5c 0xfc0c3ce4 0x00000188        0
/pkg/lib/libsyslog.dll   0xfc1d4000 0x0000530c 0xfc122c70 0x00000308        0
/pkg/lib/libbackplane.dll 0xfc1da000 0x0000134c 0xfc0c3e6c 0x000000a8        0
/pkg/lib/libnodeid.dll   0xfc1e7000 0x000091fc 0xfc1e61a8 0x00000208        0
/pkg/lib/libdebug.dll    0xfc23e000 0x0000ef64 0xfc1e6680 0x00000550        0
/pkg/lib/lib_procfs_util.dll 0xfc24d000 0x00004e2c 0xfc1e6bd0 0x000002a8        0
/pkg/lib/libsysdb.dll    0xfc252000 0x00046224 0xfc299000 0x0000079c        0
/pkg/lib/libsysdbutils.dll 0xfc29a000 0x0000ae04 0xfc29979c 0x000003ec        0
/pkg/lib/libwd_evm.dll   0xfc2a9000 0x0000481c 0xfc299b88 0x00000188        0
/pkg/lib/libbag.dll      0xfc40c000 0x0000ee98 0xfc41b000 0x00000368        0
/pkg/lib/libwd_notif.dll 0xfc4f8000 0x00005000 0xfc4fd000 0x00001000        0
```

```
/pkg/lib/libifmgr.dll    0xfc665000 0x00029780 0xfc68f000 0x00003000        0
/pkg/lib/libnetio_client.dll 0xfca6a000 0x000065c8 0xfca2c4f8 0x000001b4        0
/pkg/lib/libpa_client.dll 0xfcec5000 0x00006e9c 0xfcecc000 0x00003000        0
/pkg/lib/libltimes.dll   0xfcecf000 0x00002964 0xfcdc4f20 0x000000a8        0

Iteration 1 of 1
--------------------------------

Current process = "pkg/bin/lpts_fm", PID = 139406 TID = 2

trace_back: #0 0xfc110744 [MsgSendv]
trace_back: #1 0xfc0fbf04 [msg_sendv]
trace_back: #2 0xfc0fbbd8 [msg_send]
trace_back: #3 0xfcec7580 [pa_fm_close]
trace_back: #4 0xfcec78b0 [pa_fm_process_0]

ENDOFSTACKTRACE

REPLY (node node0_RP1_CPU0, pid 57433)

No specific TID, following all threads of 57433 (pkg/bin/lpts_pa)
---------------------------------------------------------------------

DLL Loaded by this process
--------------------------------

DLL path                 Text addr. Text size  Data addr. Data size  Version
/pkg/lib/libplatform.dll 0xfc0d6000 0x0000aa88 0xfc0e1000 0x00002000        0
/pkg/lib/libsysmgr.dll   0xfc0e3000 0x0000aeac 0xfc0c395c 0x00000388        0
/pkg/lib/libinfra.dll    0xfc0ee000 0x000332ec 0xfc122000 0x00000c70        0
/pkg/lib/libios.dll      0xfc123000 0x0002c4bc 0xfc150000 0x00002000        0
/pkg/lib/libc.dll        0xfc152000 0x00077ae0 0xfc1ca000 0x00002000        0
/pkg/lib/libltrace.dll   0xfc1cc000 0x00007f5c 0xfc0c3ce4 0x00000188        0
/pkg/lib/libsyslog.dll   0xfc1d4000 0x0000530c 0xfc122c70 0x00000308        0
/pkg/lib/libbackplane.dll 0xfc1da000 0x0000134c 0xfc0c3e6c 0x000000a8        0
/pkg/lib/libnodeid.dll   0xfc1e7000 0x000091fc 0xfc1e61a8 0x00000208        0
/pkg/lib/libdebug.dll    0xfc23e000 0x0000ef64 0xfc1e6680 0x00000550        0
/pkg/lib/lib_procfs_util.dll 0xfc24d000 0x00004e2c 0xfc1e6bd0 0x000002a8        0
/pkg/lib/libsysdb.dll    0xfc252000 0x00046224 0xfc299000 0x0000079c        0
/pkg/lib/libsysdbutils.dll 0xfc29a000 0x0000ae04 0xfc29979c 0x000003ec        0
/pkg/lib/libwd_evm.dll   0xfc2a9000 0x0000481c 0xfc299b88 0x00000188        0
/pkg/lib/lrdlib.dll      0xfc2f6000 0x0000a900 0xfc2f551c 0x00000610        0
/pkg/lib/liblrfuncs.dll  0xfc30e000 0x00001998 0xfc2ebd80 0x000001ec        0
/pkg/lib/libdscapi.dll   0xfc310000 0x0000457c 0xfc2f5b2c 0x0000035c        0
/pkg/lib/liblrdshared.dll 0xfc315000 0x00005fec 0xfc31b000 0x00002000        0
/pkg/lib/libbag.dll      0xfc40c000 0x0000ee98 0xfc41b000 0x00000368        0
/pkg/lib/libchkpt.dll    0xfc477000 0x0002ee04 0xfc474388 0x00000950        0
/pkg/lib/libwd_notif.dll 0xfc4f8000 0x00005000 0xfc4fd000 0x00001000        0
/pkg/lib/libltrace_sdt.dll 0xfc65c000 0x000034fc 0xfc65b73c 0x00000568        0
/pkg/lib/libfabhandle.dll 0xfc6be000 0x00003354 0xfc65bca4 0x00000248        0
/pkg/lib/libfsdb_ltrace_util_rt.dll 0xfc6ea000 0x00001b74 0xfc605e50 0x00000108        0
/pkg/lib/libbcdl.dll     0xfc6fb000 0x0000f220 0xfc6fa6e8 0x0000045c        0
/pkg/lib/liblpts_pa_fgid.dll 0xfc8d7000 0x00006640 0xfc7acd5c 0x00000208        0
/pkg/lib/libfgid.dll     0xfc910000 0x0001529c 0xfc926000 0x00002000        0
/pkg/lib/libltimes.dll   0xfcecf000 0x00002964 0xfcdc4f20 0x000000a8        0

Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 1

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x48201904 [<N/A>]
```

```
trace_back: #6 0x48201e3c [<N/A>]

ENDOFSTACKTRACE

Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 2

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x4821e978 [<N/A>]

ENDOFSTACKTRACE

Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 3

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x482064c4 [<N/A>]

ENDOFSTACKTRACE
```

The following example shows how to use the **follow** command to debug the chain of threads blocking thread 2 associated with the process assigned PID 139406:

```
RP/0/RP0/CPU0:router# follow process 139406 blocked iteration 1 stackonly thread 2

Attaching to process pid = 139406 (pkg/bin/lpts_fm)

Iteration 1 of 1
------------------------------

Current process = "pkg/bin/lpts_fm", PID = 139406 TID = 2

trace_back: #0 0xfc110744 [MsgSendv]
trace_back: #1 0xfc0fbf04 [msg_sendv]
trace_back: #2 0xfc0fbbd8 [msg_send]
trace_back: #3 0xfcec7580 [pa_fm_close]
trace_back: #4 0xfcec78b0 [pa_fm_process_0]

ENDOFSTACKTRACE

REPLY (node node0_RP1_CPU0, pid 57433)

No specific TID, following all threads of 57433 (pkg/bin/lpts_pa)
---------------------------------------------------------------

Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 1

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x48201904 [<N/A>]
trace_back: #6 0x48201e3c [<N/A>]

ENDOFSTACKTRACE
```

```
Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 2

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x4821e978 [<N/A>]

ENDOFSTACKTRACE

Current process = "pkg/bin/lpts_pa", PID = 57433 TID = 3

trace_back: #0 0xfc1106dc [MsgReceivev]
trace_back: #1 0xfc0fc840 [msg_receivev]
trace_back: #2 0xfc0fc64c [msg_receive]
trace_back: #3 0xfc0ffa70 [event_dispatch]
trace_back: #4 0xfc0ffc2c [event_block]
trace_back: #5 0x482064c4 [<N/A>]

ENDOFSTACKTRACE
```

process

To start, terminate, or restart a process, use the **process** command in admin EXEC mode.

process {**crash** | **restart** | **shutdown** | **start**} {*executable-namejob-id*} **location** {*node-id* | **all**}

Syntax Description

crash	Crashes a process.
restart	Restarts a process.
shutdown	Stops a process. The process is not restarted (even if considered "mandatory?).
start	Starts a process.
executable-name	Executable name of the process to be started, terminated, or restarted. Supplying an executable name for the executable-name argument performs the action for all the simultaneously running instances of the process, if applicable.
job-id	Job ID of the process instance to be started, terminated, or restarted. Supplying a job ID for the *job-id* argument performs the action for only the process instance associated with the job ID.
location { *node-id*	**all**}	Starts, terminates, or restarts a process on the designated node. The *node-id* argument is entered in the *rack/slot* notation. The **all** keyword specifies all nodes.

Command Default

None

Command Modes

Admin EXEC

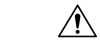
Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Under normal circumstances, processes are started and restarted automatically by the operating system as required. If a process crashes, it is automatically restarted.

Use this command to manually start, stop, or restart individual processes.



Caution Manually stopping or restarting a process can seriously impact the operation of a router. Use these commands only under the direction of a Cisco Technical Support representative.

process shutdown

The **process shutdown** command shuts down (terminates) the specified process and copies associated with the specified process. The process is not restarted, even if considered "mandatory.? Use the **show processes** command to display a list of executable processes running on the system.

Caution Stopping a process can result in an RP switchover, system failure or both. This command is intended for use only under the direct supervision of a Cisco Technical Support representative.

process restart

The **process restart** command restarts a process, such as a process that is not functioning optimally.

process start

The **process start** command starts a process that is not currently running, such as a process that was terminated using the **process kill** command. If multiple copies are on the system, all instances of the process are started simultaneously.

Task ID

Task ID	Operations
root-lr	execute

The following example shows how to restart a process. In this example, the IS-IS process is restarted:

```
RP/0/RSP0/CPU0:router# process restart isis

RP/0/RSP0/CPU0:router#RP/0/RSP0/CPU0:Mar 30 15:24:41 : isis[343]: %ISIS-6-INFO_ST
RTUP_START : Cisco NSF controlled start beginning
RP/0/RSP0/CPU0:router#RP/0/RSP0/CPU0:Mar 30 15:24:52 : isis[352]: %ISIS-6-INFO_ST
RTUP_FINISH : Cold controlled start completed
```

The following example shows how to terminate a process. In this example, the IS-IS process is stopped:

```
RP/0/RP0/CPU0:router# process shutdown isis
RP/0/RP0/CPU0:router#
```

The following example shows how to start a process. In this example, the IS-IS process is started:

```
RP/0/RSP0/CPU0:router# process start isis

RP/0/RSP0/CPU0:router#RP/0/RSP0/CPU0:Mar 30 15:27:19 : isis[227]:
   %ISIS-6-INFO_STARTUP_START : Cold controlled start beginning
RP/0/RSP0/CPU0:Mar 30 15:27:31 : isis[352]: %ISIS-6-INFO_STARTUP_FINISH :
   Cold controlled start completed
```

process core

To modify the core dump options for a process, use the **process core** command in administration EXEC mode.

process {*executable-namejob-id*} **core** {**context** | **copy** | **fallback** | **iomem** | **mainmem** | **off** | **sharedmem** | **sparse** | **sync** | **text**} [**maxcore** *value*] **location** *node-id*

Syntax Description

executable-name	Executable name of the process for which you want to change core dump options. Specifying a value for the *executable-name* argument changes the core dump option for multiple instances of a running process.
job-id	Job ID associated with the process instance. Specifying a *job-id* value changes the core dump option for only a single instance of a running process.
context	Dumps only context information for a process.
copy	Copies a core dump locally before performing the core dump.
fallback	Sets the core dump options to use the fallback options (if needed).
iomem	Dumps the I/O memory of a process.
mainmem	Dumps the main memory of a process.
off	Indicates that a core dump is not taken on the termination of the specified process.
sharedmem	Dumps the shared memory of a process.
sparse	Enables sparse core dumps of a process.
sync	Enables only synchronous core dumping.
text	Dumps the text of a process.
maxcore *value*	(Optional) Specifies the maximum number of core dumps allowed for the specified process on its creation.
location *node-id*	Sets the core dump options for a process on a designated node. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

By default, processes are configured to dump shared memory, text area, stack, data section, and heap information.

Command Modes

Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The modular architecture of Cisco IOS XR software allows core dumps for individual processes. By default, processes are configured to dump shared memory, text area, stack, data section, and heap information.

Specifying an executable name for the *executable-name job-id* argument changes the core dump option for all instances of the process. Specifying a job ID for the value changes the core dump option for a single instance of a running process.

Task ID

Task ID	Operations
root-lr	execute

The following example shows how to enable the collection of shared memory of a process:

```
RP/0/RP0/CPU0:router# process ospf core sharedmem
```

The following example shows how to turn off core dumping for a process:

```
RP/0/RP0/CPU0:router# process media_ether_config_di core off
```

process mandatory

To set the mandatory reboot options for a process, use the **process mandatory** command in the appropriate mode.

process mandatory
process mandatory {**on** | **off**} {*executable-namejob-id*} **location** *node-id*

process mandatory reboot
process mandatory reboot {**enable** | **disable**}

process mandatory toggle
process mandatory toggle {*executable-namejob-id*} **location** *node-id*

Syntax Description

on	Turns on mandatory process attribute.
off	Turns off the mandatory process attribute. The process is not considered mandatory.
reboot { **enable** | **disable**}	Enables or disables the reboot action when a mandatory process fails.
toggle	Toggles a mandatory process attribute.
executable-name	Executable name of the process to be terminated. Specifying an executable name for the *executable-name* argument terminates the process and all the simultaneously running copies, if applicable.
job-id	Job ID associated with the process to be terminated. Terminates only the process associated with the job ID.
location *node-id*	Sets the mandatory settings for a process on a designated node. The node-id argument is expressed in the *rack/slot* notation.

Command Default

No default behavior or values

Command Modes

Administration EXEC

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

If a process unexpectedly goes down, the following action occurs based on whether the process is considered mandatory.

- If the process is mandatory and the process cannot be restarted, the node automatically reboots.
- If the process is not mandatory and cannot be restarted, it stays down and the node does not reboot.

Task ID

Task ID	Operations
root-lr	execute

The following example shows how to turn on a mandatory attribute. In this example, the mandatory attribute is turned on for the media_ether_config_di process.

```
RP/0/RP0/CPU0:router# process mandatory on media_ether_config_di
```

The following example shows how to turn the reboot option on. In this example, the router is set to reboot the node if a mandatory process goes down and cannot be restarted.

```
RP/0/RP0/CPU0:router# process mandatory reboot enable

RP/0/0/CPU0:Mar 19 19:28:10 : sysmgr[71]: %SYSMGR-4-MANDATORY_REBOOT_ENABLE :
mandatory reboot option enabled by request
```

The following example shows how to turn off the reboot option. In this example, the router is set *not* to reboot the node if a mandatory process goes down and cannot be restarted. In this case, the mandatory process is restarted, but the node is not rebooted.

```
RP/0/RP0/CPU0:router# process mandatory reboot disable

RP/0/0/CPU0:Mar 19 19:31:20 : sysmgr[71]: %SYSMGR-4-MANDATORY_REBOOT_OVERRIDE
: mandatory reboot option overridden by request
```

show context

To display core dump context information, use the **show context** command in

administration EXEC mode or in EXEC

mode.

show context [*coredump-occurrence* | **clear**] [**location** {*node-id* | **all**}]

Syntax Description

coredump-occurrence	(Optional) Core dump context information to be displayed based on the occurrence of the core dump. Valid values are 1 to 10.
clear	(Optional) Clears the current context information.
location { *node-id* | **all**}	Displays core dump information that occurred on the designated node. The *node-id* argument is expressed in the *rack/slot* notation. The **all** keyword specifies to display information for all nodes.

Command Default

If no *coredump-occurrence* value is specified, core dump context information for all core dumps is displayed.

Command Modes

EXEC, Administration EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show context** command to display core dump context information. This command displays context information for the last ten core dumps. Cisco Technical Support Center engineers and development engineers use this command for post-analysis in the debugging of processes.

Use the clear context, on page 250 command to clear core dump context information.

Task ID

Task ID	Operations
diag	read

The following example shows sample output from the **show context** command:

```
RP/0/RP0/CPU0:router# show context

Crashed pid = 20502 (pkg/bin/mbi-hello)
Crash time: Thu Mar 25, 2004: 19:34:14
Core for process at disk0:/mbi-hello.20040325-193414.node0_RP0_CPU0

    Stack Trace
#0 0xfc117c9c
#1 0xfc104348
#2 0xfc104154
```

```
#3 0xfc107578
#4 0xfc107734
#5 0x482009e4
                Registers info
             r0       r1       r2       r3
  R0   0000000e  481ffa80  4820c0b8  00000003
             r4       r5       r6       r7
  R4   481ffb18  00000001  481ffa88  48200434
             r8       r9      r10      r11
  R8   00000000  00000001  00000000  fc17ac58
            r12      r13      r14      r15
  R12  481ffb08  4820c080  481ffc10  00000001
            r16      r17      r18      r19
  R16  481ffc24  481ffc2c  481ffcb4  00000000
            r20      r21      r22      r23
  R20  00398020  00000000  481ffb6c  4820a484
            r24      r25      r26      r27
  R24  00000000  00000001  4820efe0  481ffb88
            r28      r29      r30      r31
  R28  00000001  481ffb18  4820ef08  00000001
            cnt       lr      msr       pc
  R32  fc168d58  fc104348  0000d932  fc117c9c
            cnd      xer
  R36  24000022  00000004

                      DLL Info
DLL path    Text addr.  Text size   Data addr. Data size   Version
/pkg/lib/libinfra.dll  0xfc0f6000  0x00032698  0xfc0f5268  0x00000cb4
```

The following example shows sample output from the **show context** command. The output displays information about a core dump from a process that has not crashed.

```
RP/0/RP0/CPU0:router# show context

node:      node0_RP0_CPU0
------------------------------------------------------------------

Crashed pid = 28703 (pkg/bin/packet)
Crash time: Tue Sep 21, 2004: 02:48:00
Core for process at harddisk:/packet.by.dumper_gen.20040921-024800.node0_RP0_CPU0.ppc.Z
```

The following table describes the significant fields shown in the display.

Table 22: show context Field Descriptions

Field	Description
Crashed pid	Process ID (PID) of the crashed process followed by the executable path.
Crash time	Time and date the crash occurred.
Core for process at	File path to the core dump file.
Stack Trace	Stack trace information.
Registers Info	Register information related to crashed threads.
DLL Info	Dynamically loadable library (DLL) information used to decode the stack trace.

show memory

To display the available physical memory and memory usage information of processes on the router, use the **show memory** command in EXEC or administration EXEC System Admin EXEC mode.

show memory [*jobid* | **summary** [**bytes** | **detail**]] **location** *node-id*

Syntax Description

job id	(Optional) Job ID associated with a process instance. Specifying a job ID for the *job-id* argument displays the memory available and memory usage information for only the process associated with the specified job ID. If the *job-id* argument is not specified, this command displays information for all running processes.
summary	(Optional) Displays a summary of the physical memory and memory usage information.
bytes	(Optional) Displays numbers in bytes for an exact count.
detail	(Optional) Displays numbers in the format "nnn.dddM" for more detail.
location *node-id*	Displays the available physical memory from the designated node. The *node-id* argument is entered in the *rack/slot* notation.

Command Default

None

Command Modes

Administration EXEC

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To display detailed memory information for the entire router, enter the **show memory** command without any parameters.

Task ID

Task ID	Operations
basic-services	read

This example shows partial sample output from the **show memory** command entered without keywords or arguments. This command displays details for the entire router.

```
RP/0/RP0/CPU0:router# show memory

  Physical Memory:2048M total
   Application Memory :1802M (1636M available)
   Image:116M (bootram:116M)
   Reserved:128M, IOMem:0, flashfsys:0
   Total shared window:0
```

```
  kernel:jid 1
  Address         Bytes           What
  0008f000        12288           Program Stack
  000b2000        12288           Program Stack
  Total Allocated Memory:0
  Total Shared Memory:0

  sbin/devc-pty:jid 68
  Address         Bytes           What
  4817f000        4096            Program Stack (pages not allocated)
  48180000        516096          Program Stack (pages not allocated)
  481fe000        8192            Program Stack
  48200000        28672           Physical Mapped Memory
  48207000        4096            ANON FIXED ELF SYSRAM
  48208000        4096            ANON FIXED ELF SYSRAM
```

This example shows sample output from the **show memory** command entered with the job ID 7 to show the memory usage information for the process associated with this job identifier:

```
RP/0/RP0/CPU0:router# show memory 7

  Physical Memory: 256M total
   Application Memory : 249M (217M available)
   Image: 2M (bootram: 2M)
   Reserved: 4M, IOMem: 0, flashfsys: 0

  sbin/pipe: jid 7
  Address         Bytes           What
  07f7c000        126976          Program Stack (pages not allocated)
  07f9b000        4096            Program Stack
  07f9d000        126976          Program Stack (pages not allocated)
  07fbc000        4096            Program Stack
  07fbe000        126976          Program Stack (pages not allocated)
  07fdd000        4096            Program Stack
  07fdf000        126976          Program Stack (pages not allocated)
  07ffe000        4096            Program Stack
  08000000        122880          Program Stack (pages not allocated)
  0801e000        8192            Program Stack
  08020000        12288           Physical Mapped Memory
  08023000        4096            Program Text or Data
  08024000        4096            Program Text or Data
  08025000        16384           Allocated Memory
  08029000        16384           Allocated Memory
  7c001000        319488          DLL Text libc.dll
  7e000000        8192            DLL Data libc.dll
```

This example shows how to display a detailed summary of memory information for the router:

```
RP/0/RP0/CPU0:router# show memory summary detail

  Physical Memory: 256.000M total
   Application Memory : 140.178M (15.003M available)
   Image: 95.739M (bootram: 95.739M)
   Reserved: 20.000M, IOMem: 0, flashfsys: 0
   Shared window fibv6: 257.980K
   Shared window PFI_IFH: 207.925K
   Shared window aib: 8.972M
   Shared window infra_statsd: 3.980K
   Shared window ipv4_fib: 1.300M
   Shared window atc_cache: 35.937K
```

```
Shared window qad: 39.621K
Total shared window: 10.805M
Allocated Memory: 49.933M
Program Text: 6.578M
Program Data: 636.000K
Program Stack: 4.781M
```

Table 23: show memory summary Field Descriptions

Field	Description
Physical Memory	Available physical memory on the router.
Application Memory	Current memory usage of all the processes on the router.
Image	Memory that is currently used by the image and available memory.
Reserved	Total reserved memory.
IOMem	Available I/O memory.
flashfsys	Total flash memory.
Shared window fibv6	Internal shared window information.
Shared window PFI_IFH	Internal shared window information.
Shared window aib	Internal shared window information.
Shared window infra_statsd	Internal shared window information.
Shared window ipv4_fib	Internal shared window information.
Shared window atc_cache	Internal shared window information.
Shared window qad	Internal shared window information.
Total shared window	Internal shared window information.
Allocated Memory	Amount of memory allocated for the specified node.
Program Text	Internal program test information.
Program Data	Internal program data information.
Program Stack	Internal program stack information.

show memory compare

To display details about heap memory usage for all processes on the router at different moments in time and compare the results, use the **show memory compare** command in EXEC or administration EXEC System Admin EXEC mode.

show memory compare {**start** | **end** | **report**}

Syntax Description

start	Takes the initial snapshot of heap memory usage for all processes on the router and sends the report to a temporary file named /tmp/memcmp_start.out.
end	Takes the second snapshot of heap memory usage for all processes on the router and sends the report to a temporary file named /tmp/memcmp_end.out. This snapshot is compared with the initial snapshot when displaying the heap memory usage comparison report.
report	Displays the heap memory comparison report, comparing heap memory usage between the two snapshots of heap memory usage.

Command Default

None

Command Modes

Administration EXEC

EXEC

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show memory compare** command to display details about the heap memory usage of all processes on the router at different moments in time and compare the results. This command is useful for detecting patterns of memory usage during events such as restarting processes or configuring interfaces.

Use the following steps to create and compare memory snapshots:

1. Enter the **show memory compare** command with the **start** keyword to take the initial snapshot of heap memory usage for all processes on the router.

Note The snapshot is similar to that resulting from entry of the show memory heap, on page 280 command with the optional **summary** keyword.

2. Perform the test you want to analyze.
3. Enter the **show memory compare** command with the **end** keyword to take the snapshot of heap memory usage to be compared with the initial snapshot.

4. Enter the **show memory compare** command with the **report** keyword to display the heap memory usage comparison report.

Task ID

Task ID	Operations
basic-services	read

This example shows sample output from the **show memory compare** command with the **report** keyword:

```
Router# show memory compare report

  JID   name                 mem before   mem after    difference   mallocs  restarted
  ---   ----                 ----------   ---------    ----------   -------  ---------
  84    driver_infra_partner 577828       661492       83664        65
  279   gsp                  268092       335060       66968        396
  236   snap_transport       39816        80816        41000        5
  237   mpls_lsd_agent       36340        77340        41000        5
  268   fint_partner         24704        65704        41000        5
  90    null_caps_partner    25676        66676        41000        5
  208   aib                  55320        96320        41000        5
  209   ipv4_io              119724       160724       41000        5
  103   loopback_caps_partne 33000        74000        41000        5
  190   ipv4_arm             41432        82432        41000        5
  191   ipv6_arm             33452        74452        41000        5
  104   sysldr               152164       193164       41000        5
  85    nd_partner           37200        78200        41000        5
  221   clns                 61520        102520       41000        5
  196   parser_server        1295440      1336440      41000        5
  75    bundlemgr_distrib    57424        98424        41000        5
  200   arp                  83720        124720       41000        5
  201   cdp                  56524        97524        41000        5
  204   ether_caps_partner   39620        80620        41000        5
  206   qosmgr               55624        96624        41000        5
  240   imd_server           92880        104680       11800        28
  260   improxy              77508        88644        11136        10
  111   nrssvr               29152        37232        8080         60
  275   sysdb_svr_local      1575532      1579056      3524         30
  205   cfgmgr               31724        33548        1824         25
  99    sysdb_svr_shared     1131188      1132868      1680         14
  51    mbus-rp              26712        27864        1152         4
  66    wdsysmon             298068       299216       1148         15
  168   netio                1010912      1012060      1148         6
  283   itrace_manager       17408        17928        520          3
  59    devc-conaux          109868       110300       432          4
  67    syslogd_helper       289200       289416       216          2
  117   fctl                 41596        41656        60           2
  54    sysmgr               171772       171076       -696         -5
  269   ifmgr                539308       530652       -8656        -196        *
```

Table 24: show memory compare report Field Descriptions

Field	Description
JID	Process job ID.
name	Process name.

Field	Description
mem before	Heap memory usage at start (in bytes).
mem after	Heap memory usage at end (in bytes).
difference	Difference in heap memory usage (in bytes).
mallocs	Number of unfreed allocations made during the test period.
restarted	Indicates if the process was restarted during the test period.

show memory heap

To display information about the heap space for a process, use the **show memory heap** command in EXEC or administration EXEC System Admin EXEC mode.

show memory heap [**allocated**] [**dllname**] [**failure**] [**free**] {*jobid* | **all**}

Syntax Description

allocated	(Optional) Displays a list of all allocated heap blocks.
dllname	(Optional) Displays heaps with dynamic link library (DLL) names.
failure	(Optional) Displays a summary of heap failures.
free	(Optional) Displays a list of all free heap blocks.
summary	(Optional) Displays a summary of the information about the heap space.
job-id	Job ID associated with the process instance.
all	(Optional) Displays information about the heap space for all processes. The **all** keyword is only available when the **failure** or **summary** keywords are used.

Command Default

None

Command Modes

Administration EXEC

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
basic-services	read

This example shows sample output from the **show memory heap** command, specifying a job ID for the *job-id* argument:

```
RP/0/RP0/CPU0:router# show memory heap 111

  Malloc summary for pid 16433:
    Heapsize 16384: allocd 6328, free 8820, overhead 1236
    Calls: mallocs 144; reallocs 73; frees 5; [core-allocs 1; core-frees 0]
```

```
Block Allocated List
Total      Total      Block       Name/ID/Caller
Usize      Size       Count

0x000008c1 0x000008cc 0x00000001  0x7c018a10
0x000005ac 0x00000974 0x00000079  0x7c02b9e0
0x000004f0 0x000004f8 0x00000001  0x7c02b6fc
0x00000080 0x00000088 0x00000001  0x7c01936c
0x00000034 0x00000048 0x00000001  0x7c018954
0x00000024 0x00000030 0x00000001  0x7c019278
0x00000018 0x00000020 0x00000001  0x7c019b2c
0x00000008 0x00000010 0x00000001  0x7c017178
0x00000008 0x00000010 0x00000001  0x7c00fb54
0x00000008 0x00000010 0x00000001  0x7c00fb80
0x00000008 0x00000010 0x00000001  0x7c00fbb8
```

Table 25: show memory heap Field Descriptions

Field	Description
Malloc summary for pid	System-defined process ID (PID).
Heapsize	Size of the heap as allocated from the system by the malloc library.
allocd	Bytes allocated to the process.
free	Bytes available in the heap.
overhead	Malloc library overhead in bytes.
mallocs	Number of malloc calls.
reallocs	Number of realloc calls.
frees	Number of invocations to the caller interface provided in the malloc library for deallocating the memory.
[core-allocs 1; core-frees 0]	Number of core memory units, the memory units in the malloc library allocated by the system for the heap, allocated, and freed.

The following example shows sample output from the **show memory heap** command, specifying the **summary** *job-id* keyword and argument:

```
RP/0/RP0/CPU0:router# show memory heap summary 65

  Malloc summary for pid 20495 process pcmciad:
    Heapsize 65536: allocd 40332, free 16568, overhead 8636
    Calls: mallocs 883; reallocs 3; frees 671; [core-allocs 4; core-frees 0]
  Band size 16, element per block 48, nbuint 1
    Completely free blocks: 0
    Block alloced: 2, Block freed: 0
    allocs: 85, frees: 20
    allocmem: 1040, freemem: 496, overhead: 448
    blocks: 2, blknodes: 96
  Band size 24, element per block 34, nbuint 1
    Completely free blocks: 0
    Block alloced: 1, Block freed: 0
```

```
  allocs: 243, frees: 223
  allocmem: 480, freemem: 336, overhead: 168
  blocks: 1, blknodes: 34
Band size 32, element per block 26, nbuint 1
  Completely free blocks: 0
  Block alloced: 1, Block freed: 0
  allocs: 107, frees: 97
  allocmem: 320, freemem: 512, overhead: 136
  blocks: 1, blknodes: 26
Band size 40, element per block 22, nbuint 1
  Completely free blocks: 0
  Block alloced: 2, Block freed: 0
  allocs: 98, frees: 74
  allocmem: 960, freemem: 800, overhead: 240
  blocks: 2, blknodes: 44
Band size 48, element per block 18, nbuint 1
  Completely free blocks: 0
  Block alloced: 1, Block freed: 0
  allocs: 53, frees: 42
  allocmem: 528, freemem: 336, overhead: 104
  blocks: 1, blknodes: 18
Band size 56, element per block 16, nbuint 1
  Completely free blocks: 0
  Block alloced: 1, Block freed: 0
  allocs: 8, frees: 4
  allocmem: 224, freemem: 672, overhead: 96
  blocks: 1, blknodes: 16
Band size 64, element per block 14, nbuint 1
  Completely free blocks: 0
  Block alloced: 1, Block freed: 0
  allocs: 6, frees: 2
  allocmem: 256, freemem: 640, overhead: 88
  blocks: 1, blknodes: 14
Band size 72, element per block 12, nbuint 1
  Completely free blocks: 0
  Block alloced: 1, Block freed: 0
  allocs: 1, frees: 0
  allocmem: 72, freemem: 792, overhead: 80
  blocks: 1, blknodes: 12
```

Table 26: show memory heap summary Field Descriptions

Field	Description
Malloc summary for pid	System-defined process ID (pid).
Heapsize	Size of the heap as allocated from the system by the malloc library.
allocd	Bytes allocated to the process.
free	Bytes available in the heap.
overhead	Malloc library overhead in bytes.
mallocs	Number of malloc calls.
reallocs	Number of realloc calls.

Field	Description
frees	Number of invocations to the caller interface provided in the malloc library for deallocating the memory.
[core-allocs 1; core-frees 0]	Number of core memory units, the memory units in the malloc library allocated by the system for the heap, allocated and freed.
Band size	Small memory elements are arranged in bands. The band size specifies the size of elements within the band.
element per block	Number of elements per block in the band.
nbunit	Number of memory unit one block consists of. Any block in any band should be of a size that is an integer multiple of this basic unit.
Completely free blocks	Number of blocks in the band completely free (available for allocation).
Block alloced	Number of blocks currently allocated for the band.
allocs	Number of allocations currently performed from the band.
frees	Number of free calls that resulted in memory being returned to the band.
allocmem	Amount of memory currently allocated from the band.
overhead	Amount of memory in bytes as overhead for managing the band.
blocks	Number of blocks currently in the band.
blknodes	Number of nodes (elements) in all the blocks in the band.

show processes

To display information about active processes, use the **show processes** command in EXEC or administration EXEC System Admin EXEC mode.

show processes {*job-idprocess-name* | **aborts** | **all** | **blocked** | **boot** | **cpu** | **distribution** *process-name* | **dynamic** | **failover** | **family** | **files** | **location** *node-id* | **log** | **mandatory** | **memory** | **pidin** | **searchpath** | **signal** | **startup** | **threadname**} [**location** *node-id*] [**detail**] [**run**]

Syntax Description

job-id	Job identifier for which information for only the process instance associated with the *job-id* argument is displayed.
process-name	Process name for which all simultaneously running instances are displayed, if applicable.
aborts	Displays process abort information.
all	Displays summary process information for all processes.
blocked	Displays details about reply, send, and mutex blocked processes.
boot	Displays process boot information.
cpu	Displays CPU usage for each process.
distribution	Displays the distribution of processes.
dynamic	Displays process data for dynamically created processes.
failover	Displays process switchover information.
family	Displays the process session and family information.
files	Displays information about open files and open communication channels.
location *node-id*	Displays information about the active processes from a designated node. The *node-id* argument is entered in the *rack/slot* notation.
log	Displays process log.
mandatory	Displays process data for mandatory processes.
memory	Displays information about the text, data, and stack usage for processes.
pidin	Displays all processes using the QNX command.
searchpath	Displays the search path.
signal	Displays the signal options for blocked, pending, ignored, and queued signals.
startup	Displays process data for processes created at startup.
threadname	Displays thread names.

detail	(Optional) Displays more detail. This option is available only with the *process-name* argument.
run	(Optional) Displays information for only running processes. This option is available only with the *process-name* argument.

Command Default

None

Command Modes

Administration EXEC

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

- Use the **show processes** command to display general information about the active processes. To display more detailed information for a process, specify a job ID or process for the *job-id* argument or *process-name* argument, respectively.
- You can also use the **monitor processes** command to determine the top processes and threads based on CPU usage.
- If you execute the **show processes blocked** <> command when multiple show techs are being collected, a transient and an intermittent error would occur for a few seconds. You can handle this issue in one of the following ways:
  - Ignore the error and retry the **show processes blocked** <> command.
  - Avoid executing the **show processes blocked** <> command when multiple **show tech-support** <> commands are running.

Task ID

Task ID	Operations
basic-services	read

The **show processes** command with the *process-name* argument displays detailed information about a process:

```
RP/0/RSP0/CPU0:router# show processes ospf

Tue Jul 28 09:23:17.212 DST
                  Job Id: 338
                     PID: 336152
         Executable path: /disk0/asr9k-rout-3.9.0.14I/bin/ospf
              Instance #: 1
              Version ID: 00.00.0000
                 Respawn: ON
           Respawn count: 1
  Max. spawns per minute: 12
            Last started: Tue Jul 14 15:26:26 2009
           Process state: Run
```

```
           Package state: Normal
       Started on config: cfg/gl/ipv4-ospf/proc/100/ord_z/config
                    core: MAINMEM
            Max. core: 0
               Placement: Placeable
             startup_path: /pkg/startup/ospf.startup
                   Ready: 1.312s
               Available: 1.334s
        Process cpu time: 93.382 user, 13.902 kernel, 107.284 total
JID   TID CPU Stack pri state        TimeInState    HR:MM:SS:MSEC   NAME
338    1    0  116K  10 Receive        0:00:00:0375    0:00:47:0139 ospf
338    2    0  116K  10 Receive        0:00:05:0734    0:00:00:0029 ospf
338    3    1  116K  10 Receive        0:00:06:0765    0:00:00:0056 ospf
338    4    1  116K  10 Receive        0:00:00:0096    0:00:00:0698 ospf
338    5    1  116K  10 Receive        0:49:33:0609    0:00:00:0129 ospf
338    6    1  116K  10 Sigwaitinfo  329:56:49:0531    0:00:00:0000 ospf
338    7    0  116K  10 Receive        0:00:00:0816    0:00:58:0676 ospf
338    8    1  116K  10 Receive        0:00:06:0765    0:00:00:0043 ospf
338    9    1  116K  10 Condvar       82:30:01:0311    0:00:00:0029 ospf
338    10   1  116K  10 Receive       82:30:05:0188    0:00:00:0478 ospf
338    11   0  116K  10 Receive      329:54:49:0318    0:00:00:0005 ospf
------------------------------------------------------------------------------
```

Table 27: show processes Field Descriptions

Field	Description
Job id	Job ID. This field remains constant over process restarts.
PID	Process ID. This field changes when process is restarted.
Executable path	Path for the process executable.
Instance	There may be more than one instance of a process running at a given time (each instance may have more than one thread).
Version ID	API version.
Respawn	ON or OFF. The field indicates if this process restarts automatically in case of failure.
Respawn count	Number of times this process has been started or restarted (that is, the first start makes this count 1).
Max. spawns per minute	Number of respawns not to be exceeded in 1 minute. If this number is exceeded, the process stops restarting.
Last started	Date and time the process was last started.
Process state	Current state of the process.
Started on config	Configuration command that started (or would start) this process.
core	Memory segments to include in core file.
Max. core	Number of times to dump a core file. 0 = infinity.

The **show processes** command with the **memory** keyword displays details of memory usage for a given process or for all processes, as shown in the following example:

```
RP/0/RP0/CPU0:router# show processes memory

  JID    Text     Data     Stack    Dynamic  Process
  55     28672    4096     69632    17072128 eth_server
  317    167936   4096     45056    10526720 syslogd
  122    512000   4096     77824    9797632  bgp
  265    57344    4096     57344    5877760  parser_server
  254    40960    4096     143360   3084288  netio
  63     8192     4096     24576    2314240  nvram
  314    4096     4096     36864    1699840  sysdb_svr_local
  341    495616   4096     40960    1576960  wdsysmon
  259    53248    4096     28672    1490944  nvgen_server
  189    32768    4096     32768    1425408  hd_drv
  69     77824    4096     110592   1421312  qnet
  348    323584   4096     40960    1392640  ospf
  347    323584   4096     40960    1392640  ospf
  346    323584   4096     40960    1392640  ospf
  345    323584   4096     40960    1392640  ospf
  344    323584   4096     40960    1392640  ospf
  261    323584   4096     40960    1392640  ospf
   --More--
```

Table 28: show processes memory Field Descriptions

Field	Description
JID	Job ID.
Text	Size of text region (process executable).
Data	Size of data region (initialized and uninitialized variables).
Stack	Size of process stack.
Dynamic	Size of dynamically allocated memory.
Process	Process name.

The **show processes** command with the **all** keyword displays summary information for all processes, as shown in the following example:

```
RP/0/RP0/CPU0:router# show processes all

  JID    LAST STARTED            STATE    RE-      PLACE-  MANDA-  MAINT- NAME(IID) ARGS
                                          START    MENT    TORY    MODE
  ------------------------------------------------------------------------------------
  82     03/16/2007 14:54:52.488 Run      1               M       Y      wd-mbi(1)
  58     03/16/2007 14:54:52.488 Run      1               M       Y      dllmgr(1)-r 60 -u
30
  74     03/16/2007 14:54:52.488 Run      1               M       Y      pkgfs(1)
  57     03/16/2007 14:54:52.488 Run      1                       Y      devc-conaux(1) -h
-d
                                                                          librs232.dll -m
                                                                          libconaux.dll -u
                                                                          libst16550.dll
```

```
  76     03/16/2007 14:54:52.488 Run      1                    Y      devc-pty(1) -n 32
  56     Not configured          None     0                    Y      clock_chip(1) -r
-b
   --More--
```

Table 29: show processes all Field Description

Field	Description
JID	Job ID.
Last Started	Date when the process was last started.
State	State of the process.
Restart	Number of times the process has restarted since the node was booted. If a node is reloaded, the restart count for all processes is reset. Normally, this value is 1, because usually processes do not restart. However, if you restart a process using the **process restart** command, the restart count for the process increases by one.
Placement	Indicates whether the process is a placeable process or not. Most processes are not placeable, so the value is blank. ISIS, OSPF, and BGP are examples of placeable processes.
Mandatory	M indicates that the process is mandatory. A mandatory process must be running. If a mandatory process cannot be started (for example, sysmgr starts it but it keeps crashing), after five attempts the sysmgr causes the node to reload in an attempt to correct the problem. A node cannot function properly if a mandatory process is not running.
Maint Mode	Indicates processes that should be running when a node is in maintenance mode. Maintenance mode is intended to run as few processes as possible to perform diagnostics on a card when a problem is suspected. However, even the diagnostics require some services running.
Name (IID)	Name of the process followed by the instance ID. A process can have multiple instances running, so the IID is the instance ID.
Args	Command-line arguments to the process.



SNMP Server Commands Chapter Map

- clear snmp counters, on page 292
- index persistence, on page 293
- notification linkupdown, on page 294
- script snmp, on page 295
- show snmp, on page 296
- show snmp context, on page 299
- show snmp context-mapping, on page 300
- show snmp engineid, on page 301
- show snmp group, on page 302
- show snmp host, on page 304
- show snmp informs details, on page 306
- show snmp interface, on page 308
- show snmp interface notification, on page 310
- show snmp interface regular-expression, on page 312
- show snmp mib, on page 313
- show snmp mib ifmib cache, on page 316
- show snmp mib ifmib general, on page 318
- show snmp mib ifmib statsd, on page 320
- show snmp request drop summary, on page 322
- show snmp request duplicates, on page 324
- show snmp request incoming-queue detail, on page 325
- show snmp request overload stats, on page 327
- show snmp request type detail, on page 328
- show snmp request type summary, on page 329
- show snmp statistics oid group, on page 330
- show snmp statistics pdu, on page 332
- show snmp statistics poll oid all, on page 334
- Show snmp statistics poll oid nms, on page 336
- show snmp statistics slow oid, on page 337
- show snmp statistics slow oid [after/before] hh:mm:ss day mday year, on page 339
- show snmp traps details, on page 341
- show snmp users, on page 343
- show snmp view, on page 345

- snmp-server trap authentication vrf disable, on page 346
- snmp-server chassis-id, on page 347
- snmp-server community, on page 348
- snmp-server community-map, on page 350
- snmp-server contact, on page 352
- snmp-server context, on page 353
- snmp-server context mapping, on page 354
- snmp-server drop report acl, on page 356
- snmp-server drop unknown-user, on page 357
- snmp-server engineid local, on page 358
- snmp-server engineid remote, on page 359
- snmp-server entityindex persist, on page 360
- snmp-server group, on page 361
- snmp-server host, on page 364
- snmp-server ifindex persist, on page 368
- snmp-server ifmib internal cache max-duration, on page 369
- snmp-server ifmib stats cache, on page 370
- snmp-server inform, on page 371
- snmp-server interface, on page 372
- snmp-server interface subset, on page 374
- snmp-server ipv4 dscp, on page 376
- snmp-server ipv4 precedence , on page 377
- snmp-server location, on page 379
- snmp-server mibs cbqosmib persist, on page 380
- snmp-server mibs cbqosmib cache, on page 381
- snmp-server mibs eventmib congestion-control, on page 382
- snmp-server mibs eventmib packet-loss, on page 384
- snmp-server mibs sensormib cache, on page 386
- snmp-server mibs subscriber threshold access-if, on page 387
- snmp-server mibs subscriber threshold, on page 388
- snmp-server notification-log-mib, on page 390
- snmp-server packetsize, on page 392
- snmp-server queue-length, on page 393
- snmp-server script, on page 394
- snmp-server target list, on page 395
- snmp-server trap throttle-time, on page 396
- snmp-server trap-timeout, on page 397
- snmp-server traps, on page 398
- snmp-server traps bgp, on page 404
- snmp-server traps frequency synchronization, on page 406
- snmp-server traps mpls l3vpn, on page 407
- snmp-server traps ospf errors, on page 408
- snmp-server traps ospf lsa, on page 410
- snmp-server traps ospf retransmit, on page 411
- snmp-server traps ospf state-change, on page 412
- snmp-server traps ospfv3 errors, on page 414

- snmp-server traps ospfv3 state-change, on page 415
- snmp-server traps pim interface-state-change, on page 417
- snmp-server traps pim invalid-message-received, on page 418
- snmp-server traps pim neighbor-change, on page 419
- snmp-server traps pim rp-mapping-change, on page 420
- snmp-server traps rsvp, on page 421
- snmp-server traps selective-vrf-download role-change, on page 422
- snmp-server traps snmp, on page 423
- snmp-server traps subscriber session-aggregation, on page 425
- snmp-server traps syslog, on page 426
- snmp-server user, on page 427
- snmp-server view, on page 430
- snmp-server vrf, on page 432

clear snmp counters

To clear the Simple Network Management Protocol (SNMP) packet statistics shown by the **show snmp** command, use the **clear snmp counters** command in XR EXEC mode.

clear snmp counters

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **clear snmp counters** command provides the ability to clear all SNMP counters used in the **show snmp** command without restarting any processes.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to clear the SNMP counters:

```
RP/0/RP0/CPU0:router# clear snmp counters
```

index persistence

To enable index persistence on an Simple Network Management Protocol (SNMP) interface, use the **index persistence** command in SNMP interface configuration mode. To restore the default conditions with respect to this command, use the **no** form of this command.

index persistence
no index persistence

Syntax Description

This command has no keywords or arguments.

Command Default

Index persistence is disabled.

Command Modes

SNMP interface configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **index persistence** command to enable ifIndex persistence for individual entries (corresponding to individual interfaces) in the ifIndex table of the IF-MIB. IfIndex persistence retains the mapping between the ifName object values and the ifIndex object values (generated from the IF-MIB) across reboots, allowing for consistent identification of specific interfaces using SNMP.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to assign ifIndex persistence on interface 0/0/1/0:

```
RP/0/RP0/CPU0:router(config)# snmp-server interface tengige 0/0/1/0
RP/0/RP0/CPU0:router(config-snmp-if)# index persistence
```

notification linkupdown

To enable or disable linkUp and linkDown trap notifications on a Simple Network Management Protocol (SNMP) interface, use the **notification linkupdown** command in SNMP interface configuration mode. To revert to the default setting, use the **no** form of this command.

notification linkupdown disable
no notification linkupdown disable

Syntax Description

disable	Disables linkUp and linkDown trap notifications on an SNMP interface.

Syntax Description

This command has no keywords or arguments.

Command Default

By default, for all main interfaces the linkUp and linkDown trap notifications are enabled; for all subinterfaces they are disabled.

Command Modes

SNMP interface configuration

SNMP interface subset configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Enabling of linkUp and linkDown notifications is performed globally using the **snmp-server traps snmp** command. Issue the **notification linkupdown** command to disable linkUp and linkDown notifications on an interface.

Use the **no** form of this command to enable linkUp and linkDown notifications on an interface, if linkUp and linkDown notifications have been disabled.

You can also use the **snmp-server interface subset** command to enable or disable groups of interfaces.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to disable linkUp and linkDown trap notifications on interface 0/0/1/0:

```
RP/0/RP0/CPU0:router(config)# snmp-server interface tengige 0/0/1/0
RP/0/RP0/CPU0:router(config-snmp-if)# notification linkupdown disable
```

script snmp

To configure the checksum of the newly added script file, use the **script snmp** command in XR Config mode.

script snmp *file-name* **checksum sha256** *checksum-value*

no script snmp *file-name* **checksum sha256** *checksum-value*

Syntax Description

file-name	The name of the script file.
checksum sha256 *checksum-value*	The checksum of the script file.

Command Default

None

Command Modes

XR Config

Command History

Release	Modification
Release 7.5.3	This command was introduced.

Usage Guidelines

To use commands of this module, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using any command, contact your AAA administrator for assistance.

Task ID

Task ID	Operation
snmp	read,write

Examples

This example shows how to configure the checksum of the newly added script file.

```
Router(config)#script snmp show_lldp_string.py checksum sha256
156345c2cbfc1a2725b5f5ecdfb23d30d9a25e894604890d88929d724946e7b3
```

show snmp

To display the status of Simple Network Management Protocol (SNMP) communications, use the **show snmp** command in

EXEC

mode.

show snmp

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the `show snmp` command to show counter information for SNMP operations. It also displays the chassis ID string defined with the **snmp-server chassis-id** command.

Task ID

Task ID	Operations
snmp	read

This example shows sample output from the `show snmp` command:

```
RP/0/RP0/CPU0:router# show snmp

  Chassis: 01506199
  37 SNMP packets input
  0 Bad SNMP version errors
  4 Unknown community name
  0 Illegal operation for community name supplied
  0 Encoding errors
  24 Number of requested variables
  0 Number of altered variables
  0 Get-request PDUs
  28 Get-next PDUs
  0 Set-request PDUs
  78 SNMP packets output
  0 Too big errors (Maximum packet size 1500)
  0 No such name errors
  0 Bad values errors
  0 General errors
  24 Response PDUs
  13 Trap PDUs
  SNMP logging: enabled
```

```
Logging to 172.25.58.33.162, 0/10, 13 sent, 0 dropped.
```

The following table describes the significant fields shown in the display.

Table 30: show snmp Field Descriptions

Field	Description
Chassis	Chassis ID string.
SNMP packets input	Total number of SNMP packets input.
Bad SNMP version errors	Number of packets with an invalid SNMP version.
Unknown community name	Number of SNMP packets with an unknown community name.
Illegal operation for community name supplied	Number of packets requesting an operation not allowed for that community.
Encoding errors	Number of SNMP packets that were improperly encoded.
Number of requested variables	Number of variables requested by SNMP managers.
Number of altered variables	Number of variables altered by SNMP managers.
Get-request PDUs	Number of get requests received
Get-next PDUs	Number of get-next requests received.
Set-request PDUs	Number of set requests received.
SNMP packets output	Total number of SNMP packets sent by the device.
Too big errors	Number of SNMP packets that were larger than the maximum packet size.
Maximum packet size	Maximum size of SNMP packets.
No such name errors	Number of SNMP requests that specified a MIB object that does not exist.
Bad values errors	Number of SNMP set requests that specified an invalid value for a MIB object.
General errors	Number of SNMP set requests that failed due to some other error. (It is not a noSuchName error, badValue error, or any of the other specific errors.)
Response PDUs	Number of responses sent in reply to requests.
Trap PDUs	Number of SNMP traps sent.
SNMP logging	Enabled or disabled logging.
sent	Number of traps sent.

Field	Description
dropped	Number of traps dropped. Traps are dropped when the trap queue for a destination exceeds the maximum length of the queue, as set by the **snmp-server queue-length** command.

show snmp context

To display the enhanced SNMP context mappings, use the **show snmp context** command in EXEC mode.

show snmp context

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show snmp context** command to display the protocol instance, topology and VRF mappings associated with an SNMP context.

Task ID

Task ID	Operation
snmp	read

This example illustrates sample output from the **show snmp context** command:

```
RP/0/RP0/CPU0:router# show snmp context

Tue Dec 21 03:41:08.065 PST
Context-name       Vrf-name      Topology-Name   Instance-Name             Feature
con5                  vf5           tp5             in5                     OSPF
con6                  vf6           tp6             in6                     OSPF
con7                  vf7           tp7             in7                     OSPF
con8                  vf8           tp8             in8                     OSPF
```

show snmp context-mapping

To display the SNMP context mapping table, use the **show snmp context-mapping** command in

EXEC

mode.

show snmp context-mapping

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The SNMP agent handles queries based on SNMP contexts created by client features. Use the **show snmp context-mapping** command to display the SNMP context mapping table. Each entry in the table includes the name of an SNMP context created by a client instance and the name of the client that created the context.

Task ID

Task ID	Operations
snmp	read

The following example shows sample output from the **show snmp context-mapping** command:

```
RP/0/RP0/CPU0:router# show snmp context-mapping

Wed Aug 6 01:42:35.227 UTC
Context-name                 Feature-name                 Feature
ControlEthernet0_RP0_CPU0_S0  ControlEthernet0_RP0_CPU0_S0  BRIDGEINST
ControlEthernet0_RP1_CPU0_S0     ControlEthernet0_RP1_CPU0_S0     BRIDGEINST
```

Table 31: show snmp context-mapping Field Descriptions

Field	Definition
Context-name	Name of an SNMP context.
Feature-name	Name of the instance that created the context.
Feature	Name of the client whose instance created the context.

show snmp engineid

To display the identification of the local Simple Network Management Protocol (SNMP) engine that has been configured on the router, use the **show snmp engineid** command in XR EXEC mode.

show snmp engineid

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

An *SNMP engine* is a copy of SNMP that can reside on a local device.

Task ID

Task ID	Operations
snmp	read

The following example shows sample output from the **show snmp engineid** command:

```
RP/0/RP0/CPU0:router# show snmp engineid

  Local SNMP engineID: 000000090200000000C025808
```

show snmp group

To display the names of groups on the router, security model, status of the different views, and storage type of each group, use the **show snmp group** command in

EXEC

mode.

show snmp group

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

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

Task ID

Task ID	Operations
snmp	read

This example shows sample output from the **show snmp group** command:

```
RP/0/RP0/CPU0:router# show snmp group

  groupname: public security model:snmpv1
  readview : v1default writeview: -
  notifyview: v1default
  row status: nonVolatile

  groupname: public security model:snmpv2c
  readview : v1default writeview: -
  notifyview: v1default
  row status: nonVolatile
```

Table 32: show snmp group Field Descriptions

Field	Definition
groupname	Name of the Simple Network Management Protocol (SNMP) group or collection of users that have a common access policy.
readview	String identifying the read view of the group.

Field	Definition
security model	Security model used by the group, either v1, v2c, or v3.
writeview	String identifying the write view of the group.
notifyview	String identifying the notify view of the group.
row status	Settings that are set in volatile or temporary memory on the device, or in nonvolatile or persistent memory where settings remain after the device is turned off and on again.

show snmp host

To display the configured Simple Network Management Protocol (SNMP) notification recipient host, User Datagram Protocol (UDP) port number, user, and security model, use the **show snmp host** command in

EXEC

mode.

show snmp host

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read

The following example shows sample output from the **show snmp host** command:

```
RP/0/RP0/CPU0:router# show snmp host

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3auth security model: v3 auth

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3noauth security model: v3 noauth

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3priv security model: v3 priv

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userv2c security model: v2c
```

Table 33: show snmp host Field Descriptions

Field	Definition
Notification host	Name or IP address of target host.

Field	Definition
udp-port	UDP port number to which notifications are sent.
type	Type of notification configured.
user	Security level of the user.
security model	Version of SNMP used to send the trap, either v1, v2c, or v3.

show snmp informs details

To show the details about the informs generated for each host, the drop and retry count and the timestamp, use the **show snmp informs details** command in XR EXEC mode.

show snmp informs details

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp informs details** command shows these details:

Field	Description
InformOID	Generated inform.
Sent	Number of times the Inform is sent from the inform host and port configiured.
Drop	Number of times the Inform is sent from the inform host and port configiured.
Retry	Number of times the Inform retries from the inform host and port configiured
Last-sent	Time stamp when the last inform was sent from the host and port.
Last-drop	Time stamp when the last inform dropped from the host and port.
Host	Configured address of the host to receive traps.
udp-port	Configured port to receive traps.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp informs details
Mon Apr  7 17:14:17.212 UTC
HOST:9.22.24.150, udp-port:5555
```

```
------------------------------
InformOID                          Sent    Drop    Retry   Last-sent
Last-drop
ciscoConfigManMIB.2.0.1            8       2       6       Mon Apr 07 14 17:12:54      Mon
 Apr 07 14 17:12:42
ciscoFlashDeviceInsertedNotif      4       1       3       Mon Apr 07 14 17:12:55      Mon
 Apr 07 14 17:12:42
ciscoFlashDeviceRemovedNotif       4       1       3       Mon Apr 07 14 17:12:54      Mon
 Apr 07 14 17:12:42
ciscoMgmt.117.2.0.1                8       2       6       Mon Apr 07 14 17:12:53      Mon
 Apr 07 14 17:12:42
ciscoMgmt.117.2.0.2                4       1       3       Mon Apr 07 14 17:12:52      Mon
 Apr 07 14 17:12:42|
```

show snmp interface

To display the interface index identification numbers (ifIndex values) for all the interfaces or a specified interface, use the **show snmp interface** command in the appropriate mode.

show snmp interface [*type interface-path-id* **ifindex**]

Syntax Description

type	(Optional) Interface type. For more information, use the question mark (?) online help function.
interface-path-id	(Optional) Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
ifindex	(Optional) Displays the ifIndex value for the specified interface.

Command Default

Enter the **show snmp interface** command without keywords or arguments to display the ifIndex value for all interfaces.

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read

This example displays the ifIndex value for a specific interface:

```
RP/0/RP0/CPU0:router# show snmp interface pos 0/1/0/1 ifindex

ifName : POS0/1/0/1            ifIndex : 12
```

The following example displays the ifIndex value for all interfaces:

```
RP/0/RP0/CPU0:router# show snmp interface

  ifName : Loopback0             ifIndex : 1
```

```
ifName : POS0/1/0/1            ifIndex : 12
ifName : POS0/1/4/2            ifIndex : 14
ifName : POS0/1/4/3            ifIndex : 15
ifName : POS0/6/0/1            ifIndex : 2
ifName : POS0/6/4/4            ifIndex : 18
ifName : POS0/6/4/5            ifIndex : 19
ifName : POS0/6/4/6            ifIndex : 20
ifName : Bundle-POS24          ifIndex : 4
ifName : Bundle-Ether28        ifIndex : 5
ifName : Bundle-Ether28.1      ifIndex : 7
ifName : Bundle-Ether28.2      ifIndex : 8
ifName : Bundle-Ether28.3      ifIndex : 9
ifName : MgmtEth0/RP0/CPU0/0   ifIndex : 6
ifName : MgmtEth0/RP1/CPU0/0   ifIndex : 10
ifName : GigabitEthernet0/1/5/0  ifIndex : 11
ifName : GigabitEthernet0/1/5/1  ifIndex : 13
ifName : GigabitEthernet0/1/5/2  ifIndex : 3
ifName : GigabitEthernet0/6/5/1  ifIndex : 16
ifName : GigabitEthernet0/6/5/2  ifIndex : 17
ifName : GigabitEthernet0/6/5/7  ifIndex : 21
```

Table 34: show snmp interface Field Descriptions

Field	Definition
ifName	Interface name.
ifIndex	ifIndex value.

show snmp interface notification

To display the linkUp and linkDown notification status for a subset of interfaces, use the **show snmp interface notification** command in EXEC mode.

show snmp interface notification {**subset** *subset-number* | **regular-expression** *expression* | [*type interface-path-id*]}

Syntax Description

subset *subset-number*	Specifies the identifier of the interface subset. The subset-number argument is configured using the **snmp-server interface subset** command.
regular-expression *expression*	Specifies a subset of interfaces matching a regular expression, for which to display information.
type	(Optional) Interface type. For more information, use the question mark (?) online help function.
interface-path-id	(Optional) Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Regular expressions have two constraints:

- Regular expressions must always be entered within double quotes to ensure that the CLI interprets each character correctly.
- All characters that are part of a regular expression are considered regular characters with no special meaning. In order to enter special characters, such as "\" or "?," they must be preceded by the backslash character "\." For example, to enter the regular expression ([A-Z][A-Z0-9]*)\b[^>]*>(.*?)</\1, you would enter ([A-Z][A-Z0-9]*)\\b[^>]*>(.*\?)</\\1.

Refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in for more information regarding regular expressions.

When using the **subset** or **regular-expression** keywords, the actual display might not match the configuration if there are higher priority *subset-number* values that actually apply to the interface. This can happen for a

set of interfaces that are included in two or more configured regular expressions or where an individual interface configuration is enabled.

Task ID

Task ID	Operation
snmp	read

The following example illustrates how to display linkUp and linkDown notification status for a subset of interfaces identified by a specific *subset-number* :

```
RP/0/RP0/CPU0:router# show snmp interface notification subset 3
```

This example illustrates how to display linkUp and linkDown notification status for a subset of interfaces identified by a regular expression:

```
RP/0/RP0/CPU0:router# show snmp interface notification regular-expression
"^Gig[a-zA-Z]+[0-9/]+\."
```

show snmp interface regular-expression

To display interface names and indices assigned to interfaces that match a regular expression, use the **show snmp interface regular-expression** command in EXEC mode.

show snmp interface regular-expression *expression*

Syntax Description

expression	Specifies a subset of interfaces matching a regular expression, for which to display information.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

All characters that are part of a regular expression are considered regular characters with no special meaning. In order to enter special characters, such as "\" or "?," they must be preceded by the backslash character "\." For example, to enter the regular expression ([A-Z][A-Z0-9]*)\b[^>]*>(.*?)</\1, you would enter ([A-Z][A-Z0-9]*)\\b[^>]*>(.*\?)</\\1.

Refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in for more information regarding regular expressions.

Task ID

Task ID	Operation
snmp	read

This example illustrates how to display information for interfaces that match the given regular expression:

```
RP/0/RP0/CPU0:router# show snmp interface regular-expression "^Gig[a-zA-Z]+[0-9/]+\."
```

show snmp mib

To display a list of MIB module object identifiers (OIDs) registered on the system, use the **show snmp mib** command in

EXEC

mode.

show snmp mib [*object-name* | **dll**]

Syntax Description

object-name	(Optional) Specific MIB object identifier or object name.
dll	(Optional) Displays a list of all MIB DLL filenames and the OID supported by each DLL filename on the system.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show snmp mib** command to display a list of the MIB module instance identifiers registered on the system.

Although the **show snmp mib** command can be used to display a list of MIB OIDs registered on the system, the use of a Network Management System (NMS) application is the recommended alternative for gathering this information.

The **show snmp mib** command is intended only for network managers who are familiar with Abstract Syntax Notation One (ASN.1) syntax and the Structure of Management Information (SMI) of Open Systems Interconnection (OSI) Reference Model.

SNMP management information is viewed as a collection of managed objects residing in a virtual information store termed the *MIB*. Collections of related objects are defined in MIB modules. These modules are written using a subset of ASN.1 termed the *SMI.*

The definitions for the OIDs displayed by this command can be found in the relevant RFCs and MIB modules. For example, RFC 1907 defines the system.x, sysOREntry.x, snmp.x, and snmpTrap.x OIDs, and this information is supplemented by the extensions defined in the CISCO-SYSTEM-MIB.

Use the **detailed** keyword to display a list of the MIB module instance identifiers registered on the system. The output displays additional details, such as DLL and configuration information.

Use the **dll** keyword to display a list of the MIB modules loaded into the agent. This command can be used to find the supported MIBs.



Note This command produces a high volume of output if SNMP is enabled on the system. To exit from a --More-- prompt, press **Crtl-Z**.

Task ID

Task ID	Operations
snmp	read

The following example shows sample output from the **show snmp mib** command:

```
RP/0/RP0/CPU0:router# show snmp mib

1.3.6.1.2.1.47.1.1.1.1.2
1.3.6.1.2.1.47.1.1.1.1.3
1.3.6.1.2.1.47.1.1.1.1.4
1.3.6.1.2.1.47.1.1.1.1.5
1.3.6.1.2.1.47.1.1.1.1.6
1.3.6.1.2.1.47.1.1.1.1.7
1.3.6.1.2.1.47.1.1.1.1.8
1.3.6.1.2.1.47.1.1.1.1.9
1.3.6.1.2.1.47.1.1.1.1.10
1.3.6.1.2.1.47.1.1.1.1.11
1.3.6.1.2.1.47.1.1.1.1.12
1.3.6.1.2.1.47.1.1.1.1.13
1.3.6.1.2.1.47.1.1.1.1.14
1.3.6.1.2.1.47.1.1.1.1.15
1.3.6.1.2.1.47.1.1.1.1.16
1.3.6.1.2.1.47.1.2.1.1.2
1.3.6.1.2.1.47.1.2.1.1.3
1.3.6.1.2.1.47.1.2.1.1.4
1.3.6.1.2.1.47.1.2.1.1.5
1.3.6.1.2.1.47.1.2.1.1.6
1.3.6.1.2.1.47.1.2.1.1.7
1.3.6.1.2.1.47.1.2.1.1.8
1.3.6.1.2.1.47.1.3.1.1.1
--More-
```

This example shows sample output from the **show snmp mib** command with the **detailed** keyword:

```
RP/0/RP0/CPU0:router# show snmp mib detailed

Entitymib:dll=/pkg/lib/mib/libEntitymib.dll, config=Entity.mib, loaded
1.3.6.1.2.1.47.1.1.1.1.2
1.3.6.1.2.1.47.1.1.1.1.3
1.3.6.1.2.1.47.1.1.1.1.4
1.3.6.1.2.1.47.1.1.1.1.5
1.3.6.1.2.1.47.1.1.1.1.6
1.3.6.1.2.1.47.1.1.1.1.7
1.3.6.1.2.1.47.1.1.1.1.8
1.3.6.1.2.1.47.1.1.1.1.9
1.3.6.1.2.1.47.1.1.1.1.10
1.3.6.1.2.1.47.1.1.1.1.11
1.3.6.1.2.1.47.1.1.1.1.12
1.3.6.1.2.1.47.1.1.1.1.13
1.3.6.1.2.1.47.1.1.1.1.14
```

```
1.3.6.1.2.1.47.1.1.1.1.15
1.3.6.1.2.1.47.1.1.1.1.16
1.3.6.1.2.1.47.1.2.1.1.2
1.3.6.1.2.1.47.1.2.1.1.3
1.3.6.1.2.1.47.1.2.1.1.4
1.3.6.1.2.1.47.1.2.1.1.5
1.3.6.1.2.1.47.1.2.1.1.6
1.3.6.1.2.1.47.1.2.1.1.7
1.3.6.1.2.1.47.1.2.1.1.8
--More--
```

This example shows sample output from the **show snmp mib** command with the **dll** keyword:

```
RP/0/RP0/CPU0:router# show snmp mib dll

Entitymib:dll=/pkg/lib/mib/libEntitymib.dll, config=Entity.mib, loaded
bgp4mib:dll=/pkg/lib/mib/libbgp4mib.dll, config=bgp4.mib, loaded
cdpmib:dll=/pkg/lib/mib/libcdpmib.dll, config=cdp.mib, loaded
ciscoprocessmib:dll=/pkg/lib/mib/libciscoprocessmib.dll,
 config=ciscoprocess.mib, loaded
ciscosyslogmib:dll=/pkg/lib/mib/libciscosyslogmib.dll,
 config=ciscosyslog.mib, loaded
ciscosystemmib:dll=/pkg/lib/mib/libciscosystemmib.dll,
 config=ciscosystem.mib, loaded
confcopymib:dll=/pkg/lib/mib/libconfcopymib.dll, config=confcopy.mib,
 loaded
configmanmib:dll=/pkg/lib/mib/libconfigmanmib.dll, config=configman.mib,
 loaded
dot3admib:dll=/pkg/lib/mib/libdot3admib.dll, config=dot3ad.mib,
 loaded
fabhfrmib:dll=/pkg/lib/mib/libfabhfrmib.dll, config=fabhfr.mib,
 loaded
fabmcastapplmib:dll=/pkg/lib/mib/libfabmcastapplmib.dll,
 config=fabmcastappl.mib, loaded
fabmcastmib:dll=/pkg/lib/mib/libfabmcastmib.dll, config=fabmcast.mib,
 loaded
flashmib:dll=/pkg/lib/mib/libflashmib.dll, config=flash.mib,
 loaded
hsrpmib:dll=/pkg/lib/mib/libhsrpmib.dll, config=hsrp.mib, loaded
icmpmib:dll=/pkg/lib/mib/libicmpmib.dll, config=icmp.mib, loaded
ifmib:dll=/pkg/lib/mib/libifmib.dll, config=if.mib, loaded
ipmib:dll=/pkg/lib/mib/libipmib.dll, config=ip.mib, loaded
mempoolmib:dll=/pkg/lib/mib/libmempoolmib.dll, config=mempool.mib,
 loaded
mplsldpmib:dll=/pkg/lib/mib/libmplsldpmib.dll, config=mplsldp.mib,
 loaded
.
.
.
```

show snmp mib ifmib cache

To show the Ifindex that has exceeded the threshold time for processing, the request type and the time stamp, use the **show snmp mib ifmib cache** command in XR EXEC mode. The threshold time for the data to create an entry is 500 milli seconds.

show snmp mib ifmib cache

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This command displays the latest 500 entries. An entry will be added when the difference between Cache in and Cache out time is more than 500 milli seconds. The timeout value cannot be changed.

The **show snmp mib ifmib cache** command shows these details:

Field	Description
Index	Interface index.
MIB IN	Time stamp of the request when IFMIB starts processing.
Cache In	Time stamp in milli seconds when data retrieval from the cache starts for the request. It is offset from MIB IN time stamp.
Cache Out	Time stamp in milli seconds when data is retrieved from cache. It is offset from MIB IN time stamp.
MIB Out	Time stamp in milli seconds of the response from IF MIB. It is offset from MIB IN time stamp.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp mib ifmib cache
IFIndex    Type          MIB IN                    CACHE IN[ms]          CACHE OUT[ms]
       MIB OUT[ms]

2           NEXT          Mar 18 07:14:41.815           4                    701
          701
```

```
2           NEXT          Mar 18 07:15:36.815           0                     679
            679

2           NEXT          Mar 18 07:16:00.735           0                     684
            684
```

show snmp mib ifmib general

To show how many requests get data from internal cache and how many requests are sent to statsd to get data, use the **show snmp mib ifmib general** command in XR EXEC mode.

show snmp mib ifmib general

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Default IFMIB internal cache is 15 seconds. To change the duration, use the command:

```
snmp-server ifmib internal cache max-duration <0-60>
```

The default duration is 15 seconds, 0 seconds to disable the IFMIB internal cache.

To service the requests from Stats cache instead of Drivers, use the command:

```
snmp-server ifmib stats cache
```

The **show snmp mib ifmib general** command shows these details:

Field	Description
Cache Hit	Number of times the request retrieves data from IFMIB internal cache.
Cache Miss	Number of times the request processed from statsd, and not from IFMIB internal cache
Last Access Time	Latest time stamp of corresponding hit or miss.
Count	Number of times the data is retrieved.

The Cache Hit and Cache Miss are 32 bit counters. The maximum value is 2^31 and reset to 0 if the maximum value is exceeded.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# Show snmp mib ifmib general
 Fri Mar 14 05:05:50.408 PDT

Type                    Count            Last Access Time
```

```
Cache Hit               328             Mar 14 05:05:47.480
Cache Miss              2               Mar 14 05:05:47.386
```

show snmp mib ifmib statsd

To show the Ifindex that has exceeded the threshold time for processing, the request type and the time stamp, use the **show snmp mib ifmib statsd** command in XR EXEC mode. The threshold time for the data to create an entry is 500 milli seconds.

show snmp mib ifmib statsd

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This command displays the latest 500 entries. An entry will be added when the difference between Stats in and Stats out time is more than 500 milli seconds. The timeout value cannot be changed.

The **show snmp mib ifmib statsd** command shows these details:

Field	Description
Index	Interface index.
MIB IN	Time stamp of the request when IFMIB starts processing.
Stats In	Time stamp in milli seconds when data retrieval from the Statsd starts for the request. It is offset from MIB IN time stamp.
Stats Out	Time stamp in milli seconds when data is retrieved from Statsd. It is offset from MIB IN time stamp.
MIB Out	Time stamp in milli seconds of the response from IF MIB. It is offset from MIB IN time stamp.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp mib ifmib statsd
IFIndex    Type          MIB IN                    STATS IN[ms]      STATS OUT[ms]     MIB
 OUT[ms]

2          NEXT          Mar 18 07:14:41.815         4                   701
701
```

```
2           NEXT          Mar 18 07:15:36.815         0                   679
679

2           NEXT          Mar 18 07:16:00.735         0                   684
684
```

show snmp request drop summary

To show the summary of overall packet drop, use the **show snmp request drop summary** command in XR EXEC mode.

show snmp request drop summary

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp request drop summary** command shows these details:

Field	Description
NMS Address	Address of network management station from which request is received.
IN Q	Number of packets dropped in incoming queue as the dropped packets are not processed more than 10 seconds.
Encode	Number of packets dropped because of encode errors.
Duplicate	Number of requests dropped with duplicate request feature.
Stack	Numbers of requests are dropped in stack.
AIPC	Number of packets dropped at AIPC module.
Overload	Number of packets dropped because of overload control notification.
Timeout	Number of packets are dropped because of slow response from MIB.
Internal	Number of packets dropped because of internal failures.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp request drop summary
 Fri Mar 14 05:32:31.732 PDT
NMS Address          INQ   Encode    Duplicate   Stack    AIPC   Overload   Timeout
Internal
 1.2.3.4              0     0         0           0      0         218        0         0

NMS Address : 1:2:3::4
INQ      Encode     Duplicate    Stack    AIPC    Overload     Timeout     Internal
0         0          0            0       0       109          0           0
```

show snmp request duplicates

To display the number of duplicate protocol data unit (PDU) requests dropped by the SNMP agent, use the **show snmp request duplicates** command in

EXEC

mode.

show snmp request duplicates

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read

This example illustrates sample output from the **show snmp request duplicates** command:

```
RP/0/RP0/CPU0:router# show snmp request duplicates

No of Duplicate request received/Dropped : 0
```

show snmp request incoming-queue detail

To show the details of the queue of incoming SNMP requests, use the **show snmp request incoming-queue detail** command in XR EXEC mode.

show snmp request incoming-queue detail

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This command shows an output for maximum of 15 queues and an additional general queue. The entry will be deleted when any queue is not polled for 30 minutes.

This command shows these details:

Field	Description
NMS Address	Source address (IPv4 or IPv6) of network management system (NMS) queue. Specifies the NMS packet requests in this queue.
Q Depth	Number of packets to be processed in the queue.
Deque Count	Number of packets that are processed.
Priority	Priority of queue with packets to be processed. The priority ranges from 1 to 5, 1 indicates low priority and 5 indicates high priority.
Enque time	Time stamp of last request in the queue.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp request incoming-queue detail
 Wed Mar 12 05:16:59.505 PDT

NMS ADDRESS          Q Depth      Deque count          Priority              Enque time

4.5.6.7               0               1223             1                     Wed Mar 12
05:16:25
```

```
1.2.3.4               0                1193              1                     Wed Mar 12
05:15:06

General Q             0                 0                0                     Wed Mar 12
05:14:49

NMS ADDRESS       : 4:5:6::7

  Q Depth     Deque count           Priority               Enque time

  0                1220               1                    Wed Mar 12 05:16:02

NMS ADDRESS       : 1:2:3::4

  Q Depth     Deque count          Priority               Enque time

  0               1221               1                     Wed Mar 12 05:15:37
```

show snmp request overload stats

To show the number of packets dropped due to overload feature, use the **snmp request overload stats** command in XR EXEC mode.

show snmp request overload stats

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This command displays the latest 100 entries.

The show snmp request overload stats command shows these details:

Field	Description
StartTime	Time when overload control notification is received.
InQInDrop	Number of packet drops before inserting in incoming queue.
InQOutDrop	Number of packets dropped from incoming queue.
EndTime	Time when overload control notification ends.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp request overload stats
Thu Mar 13 07:00:45.575 UTC
StartTime                          InQInDrop        InQOutDrop                EndTime

Thu Mar 13 07:00:28                   1               0                        Thu Mar
13 07:00:38
```

show snmp request type detail

To shows the group that is polled frequently and from which network management system (NMS), use the **show snmp reqest type detail** command in XR EXEC mode.

show snmp request type detail

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp request type detail** command shows these details:

Field	Description
NMS Address	Address of Network Management Station from which the request is received.
Request	Number of requests from NMS.
SNMPD	Number of requests to snmpd.
Interface	Number of requests to mibd_interface.
Entity	Number of requests to mibd_entity.
Route	Number of requests to mibd_route.
Infra	Number of requests to mibd_infra.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp request type detail
Wed Mar 12 05:17:34.838 PDT
NMS Address      Request    AGENT    INTERFACE    ENTITY    ROUTE    INFRA
1.2.3.4          1193       52          742        70       267        123
4.5.6.7          1223       52          742        100      267        123
1:2:3::4         1221       52          742        100      265        123
4:5:6::7         1220       52          742        100      265        122
```

show snmp request type summary

To show the types of requests sent from each network management system (NMS), use the **show snmp request type summary** command in XR EXEC mode.

show snmp request type summary

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp request type summary** command shows these details:

Field	Description
NMS address	IP address of the NMS that sent the request.
Get	Number of requests of Get type.
Getnext	Number of requests of Getnext type.
Getbulk	Number of requests of Getbulk type.
Set	Number of requests of Set type.
Test	Number of requests of Test type that is part of Set request.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp request type summary
 Wed Mar 12 05:17:14.643 PDT
NMS Address           Get      GetNext      GetBulk      Set      Test
 1.2.3.4              0          1254         0          0          0
 4.5.6.7              0          5101         0          0          0

NMS Address : 1:2:3::4
 Get      GetNext      GetBulk      Set      Test
  0          2536         0          0          0
NMS Address : 4:5:6::7
 Get      GetNext      GetBulk      Set      Test
  0          3817         0          0          0
```

show snmp statistics oid group

To show the statistics of object ID (OID), use the **show snmp statistics oid group** command in XR EXEC mode.

show snmp statistics oid group {**interface** | **infra** | **route** | **entity**}

Syntax Description

interface	mibd_interface sub-agent process
infra	mibd_infra sub-agent process
route	mibd_route sub-agent process
entity	mibd_entity sub-agent process

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The latest 500 entries for each group is displayed and a maximum of 2000 entries is displayed for four groups.

The **show snmp statistics oid group** command shows these details:

Field	Description
SerNum	Unique serial number for each request processing in sub-agents.
Type	Request type.
NumObj	Number of OIDs processing in this request.
MIBMGR-IN	Time stamp of request received from AIPC.
PDU-IN	Time stamp of request sent to MIB for processing. This will be offset in milli seconds from MIBMGR_IN time stamp.
FROM-MIB	Time stamp of response sent from MIB after processing. This will be offset in milli seconds from MIBMGR_IN time stamp.
PDU-OUT	Time stamp of response sent to SNMP through AIPC . This will be offset in milli seconds from MIBMGR_IN.
OID	OID info processing this request.
MIB-IN	Time stamp of the request sent to MIB for each OID.

Field	Description
MIB-OUT	Time stamp of response sent from MIB after processing. This will be offset in milli seconds from MIB-IN.
ExpNext	Request Exp-Next.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp statistics oid group interface
Thu Mar 13 07:10:30.310 UTC
SerNum: 2489   Type: GETNEXT     NumObj: 1
  MIBMGR-IN    PDU-IN[ms]          PDU-OUT[ms]          MIBMGR-OUT[ms]
  Mar 13 07:00:49.933                1030                 1030
    OID: 1.3.6.1.2.1.10.32.4.2.0      Exp-Next: Yes
      MIB-IN : Mar 13 07:00:49.933   MIB-OUT[ms] : 1030
SerNum: 10203    Type: GETNEXT     NumObj: 1
  MIBMGR-IN      PDU-IN[ms]          PDU-OUT[ms]            MIBMGR-OUT[ms]
  Mar 13 06:36:16.976    0             1031                   1031
    OID: 1.3.6.1.2.1.10.32.4.2.0  Exp-Next: Yes
      MIB-IN : Mar 13 06:36:16.976   MIB-OUT[ms] : 1031
```

show snmp statistics pdu

To show if processing time of any protocol data unit (PDU) is more than threshold limit, use the **show snmp statistics pdu nms** command in XR EXEC mode.

show snmp statistics pdu nms[address]

Syntax Description

nms [address]	Address of Network Management Station from which request has arrived. The PDU statistics is filtered for each NMS.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The PDU processing time can exceed the threshold limit in these scenarios:

- SNMPD not able to dispatch the request to MIB because of any failures in snmpd.
- MIB response after threshold limit.
- MIB does not respond to SNMPD.

Default threshold limit is 2 seconds. To change the default threshold value, use the command:

```
Snmp-server timeouts pdu stats <1-10>
```

The maximum number of entries per network management system (NMS) is 500 and the maximum number of NMS is 30.

This command shows these details:

Field	Description
NMS	Address of Network Management Station from which request has arrived.
Port	Port number of application that requested the SNMP query.
REQID	Request ID for each PDU.
Type	Type of PDU.
SerNum	The unique number generated for every request and sent to all MIBDs.
Timeout	If the request was timeout out set to TRUE, else set to FALSE.
InputQ-In	Time stamp of the PDU when queued into input Q.

Field	Description
InputQ-Out	Time stamp of the PDU when queued into input Q, This will be in milliseconds, Offset from INPUT-IN time stamp.
ProcQ-In	Time stamp of the PDU when queued into Processing Q. This will be in milliseconds, Offset from INPUT-IN time stamp.
Response	Time stamp in milli seconds of the PDU when response is received from sub agents. Offset from INPUT-IN time stamp.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp statistics pdu nms
 Thu Mar 13 08:03:17.322 UTC
NMS: 64.103.222.6   PORT: 35028
  REQID:962974264  TYPE: 161  SerNum: 9428 TIMEOUT: No
  INPUTQ-IN                   INPUTQ-OUT[ms]       PROCQ-IN[ms]        RESPONSE[ms]
 Mar 13 08:03:15.269            0                   0                   1056
```

show snmp statistics poll oid all

To show all object IDs (OIDs) polled from all network management system (NMS) and how many times it has polled, use the **show snmp statistics poll oid all** command in XR EXEC mode.

For this command to work, the following configuration has to be committed:

(config)#snmp-server oid-poll-stats

show snmp statistics poll oid all

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The maximum number of entries equals the number of OIDs that were polled. The maximum number of NMS details for each OID is 15.

The **show snmp statistics poll oid all** command shows these details:

Field	Description
Object ID	OID requested from NMS.
NMS	List of NMS IP address requested for each OID.
Count	Number of times OID is polled for each NMS.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp statistics poll oid all
 Object ID   :   1.3.6.1.2.1.1.3
 NMS                 COUNT
 10.2.1.3              10
 10.3.1.2              30
 10.4.1.3              20
 10.12.1.3             5

Object ID    :    1.3.6.1.2.1.1.4
NMS                   COUNT
10.2.1.3               10
10.3.1.2               5
10.4.1.3               20
10.12.1.3              30
```

```
Object ID   :  1.3.6.1.2.1.1.5
NMS                  COUNT
10.2.1.3                10
10.3.1.2                3
10.4.1.3                2
```

Show snmp statistics poll oid nms

To show which object ID (OID) is polled from which network management system (NMS) and how many times it has polled, use the **show snmp statistics poll oid nms** command in XR EXEC mode.

show snmp statistics poll oid nms<V4 / V6 address>

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp statistics poll oid nms** command shows these details:

Field	Description
Object ID	OID requested from NMS.
NMS	List of NMS IP address requested for each OID.
Count	Number of times OID is polled for each NMS.

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp statistics poll nms 1.2.3.4
 NMS Address   : 1.2.3.4
Object ID                  Count
1.3.6.1.2.1.2.2.1.2        14
```

show snmp statistics slow oid

To show the object ID (OID) that has exceeded beyond the threshold time for processing and the number of times that the threshold limit is exceeded with the latest timestamp, use the **show snmp statistics slow oid** command in XR EXEC mode.

show snmp statistics slow oid

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Default threshold limit for this data as 500 milli seconds. To change the default value, use the command:

```
Snmp-server logging threshold oid-processing < <0-20000>
```

The latest 500 entries for each sub agent is displayed and a total of upto 2000 entries is maintained.

The **show snmp statistics slow oid** command shows these details:

Field	Description
Type	Request type for slow OID.
Exact OID	Requested OID from NMS.
Resp OID	Response OID for the Request type and EXACT OID.
Slow Count	Number of times OID is slow.
Slow Time	Time taken for processing the OID in milli seconds.
Time Stamp	Time stamp of the slow OID when MIB responded to MIBD.

Task ID

Task ID	Operations
snmp	read, write

This example shows a slow OIDs that exceeds the specified threshold time.

```
RP/0/RP0/CPU0:router# show snmp statistics slow oid
 Group:agent

  TYPE                : GETNEXT
```

```
  REQ_OID          : 1.3.6.1.2.1.1.1.0
  RESP_OID        : 1.3.6.1.2.1.1.1.2
  COUNT             : 2
  TIME[ms]          : 0
  TIME_STAMP    : Mar 13 05:36:52.279

Group:infra

Group:route

  TYPE              : GETNEXT
  REQ_OID        :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  RESP_OID      :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  COUNT           : 4
  TIME[ms]        : 14
  TIME_STAMP  : Mar 13 05:36:52.279
  TYPE              : GET
  REQ_OID        :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  RESP_OID      :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  COUNT           : 4
  TIME[ms]        : 14
  TIME_STAMP  : Mar 13 05:36:52.279

Group:entity

Group:interface

  TYPE               : GETNEXT
  REQ_OID         : 1.3.6.1.2.1.2.1
  RESP_OID       : 1.3.6.1.2.1.2.1.0
  COUNT            : 1
  TIME[ms]         : 0
  TIME_STAMP   : Mar 13 05:36:52.279
```

show snmp statistics slow oid [after/before] hh:mm:ss day mday year

To show the object ID (OID) that has exceeded beyond the threshold time for processing and the number of times that the threshold limit is exceeded with the latest timestamp, use the **show snmp statistics slow oid [after/before] hh:mm:ss day mday year** command in XR EXEC mode.

show snmp statistics slow oid[after/before] hh:mm:ss day mday year

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Default threshold limit for this data as 500 milli seconds. To change the default value, use the command:

```
Snmp-server logging threshold oid-processing < <0-20000>
```

The latest 500 entries for each sub agent is displayed and a total of upto 2000 entries is maintained.

The **show snmp statistics slow oid [after/before] hh:mm:ss day mday year** command shows these details:

Field	Description
Type	Request type for slow OID.
Exact OID	Requested OID from NMS.
Resp OID	Response OID for the Request type and EXACT OID.
Slow Count	Number of times OID is slow.
Slow Time	Time taken for processing the OID in milli seconds.
Time Stamp	Time stamp of the slow OID when MIB responded to MIBD.

Task ID

Task ID	Operations
snmp	read, write

This example shows a slow OIDs that exceeds the specified threshold time.

```
RP/0/RP0/CPU0:router# show snmp statistics slow oid
Group:agent
```

```
  TYPE                : GETNEXT
  REQ_OID          : 1.3.6.1.2.1.1.1.0
  RESP_OID        : 1.3.6.1.2.1.1.1.2
  COUNT              : 2
  TIME[ms]           : 0
  TIME_STAMP    : Mar 13 05:36:52.279

Group:infra
Group:route
  TYPE               : GETNEXT
  REQ_OID        :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  RESP_OID       :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  COUNT            : 4
  TIME[ms]         : 14
  TIME_STAMP  : Mar 13 05:36:52.279

  TYPE               : GET
  REQ_OID        :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  RESP_OID       :
1.3.6.1.2.1.4.34.1.3.4.20.254.128.0.0.0.0.0.0.254.8.255.254.203.38.197.0.0.0.2
  COUNT            : 4
  TIME[ms]         : 14
  TIME_STAMP  : Mar 13 05:36:52.279

Group:entity
Group:interface
  TYPE                : GETNEXT
  REQ_OID          : 1.3.6.1.2.1.2.1
  RESP_OID        : 1.3.6.1.2.1.2.1.0
  COUNT             : 1
  TIME[ms]          : 0
  TIME_STAMP   : Mar 13 05:36:52.279
```

show snmp traps details

To show the details about the traps generated for each host, the sent and drop count and the timestamp, use the **show snmp traps details** command in XR EXEC mode.

show snmp traps details

This command has no keywords or arguments.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **show snmp traps details** command shows these details:

Field	Description
TrapOID	Generated trap.
Sent	Number of times the trap sent from the host and port configured.
Drop	Number of times the trap dropped from the host and port configured.
Last-sent	Time stamp when the last trap was sent from the host and port.
Last-drop	Time stamp when the last trap dropped from the host and port.
Host	Configured address of the host to receive traps
udp-port	Configured port to receive traps

Task ID

Task ID	Operations
snmp	read, write

```
RP/0/RP0/CPU0:router# show snmp traps details
Mon Apr  7 17:14:07.241 UTC
HOST:9.22.24.150, udp-port:3333
------------------------------
TrapOID                          Sent    Drop     Last-sent                     Last-drop
ciscoConfigManMIB.2.0.1            2         0      Mon Apr 07 14 17:12:29          ~
```

```
ciscoFlashDeviceInsertedNotif       1         0      Mon Apr 07 14 17:12:28          ~
ciscoFlashDeviceRemovedNotif        1         0      Mon Apr 07 14 17:12:28          ~
```

show snmp users

To display information about the configured characteristics of Simple Network Management Protocol (SNMP) users, use the **show snmp users** command in

EXEC

mode.

show snmp users

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

An SNMP user must be part of an SNMP group, as configured using the **snmp-server user** command.

Use the **show snmp users** command to display information about all configured users.

When configuring SNMP, you may see the logging message "Configuring snmpv3 USM user." USM stands for the User-Based Security Model (USM) for SNMP Version 3 (SNMPv3). For further information about USM, see RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*.

Task ID

Task ID	Operations
snmp	read

This example shows sample output from the **show snmp users** command:

```
RP/0/RP0/CPU0:router# show snmp users

  User name:user1
  Engine ID:localSnmpID
  storage-type:nonvolatile active
```

Table 35: show snmp users Field Descriptions

Field	Definition
User name	String identifying the name of the SNMP user.
Engine ID	String identifying the name of the copy of SNMP on the device.

Field	Definition
storage-type	Settings that are set in volatile or temporary memory on the device, or in nonvolatile or persistent memory where settings remain after the device is turned off and on again.

show snmp view

To display the configured views and the associated MIB view family name, storage type, and status, use the **show snmp view** command in

EXEC

mode.

show snmp view

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read

This example shows sample output from the **show snmp view** command:

```
RP/0/RP0/CPU0:router# show snmp view

  view1 1.3 - included nonVolatile active
  v1default 1.3.6.1 - included nonVolatile active
```

snmp-server trap authentication vrf disable

To disable authentication traps on VPNs, use the **snmp-server trap authentication vrf disable** command in global configuration mode.

snmp-server trap authentication vrf disable

Syntax Description

This command has no keywords or arguments.

Command Default

Authentication traps are enabled on VPNs by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

This example illustrates how to disable authentication traps on VPNs:

```
RP/0/RP0/CPU0:router(config)# snmp-server trap authentication vrf disable
```

snmp-server chassis-id

To provide a message line identifying the Simple Network Management Protocol (SNMP) server serial number, use the **snmp-server chassis-id** command in

global configuration

mode. To restore the default value, if any, use the **no** form of this command.

snmp-server chassis-id *serial-number*
no snmp-server chassis-id

Syntax Description

serial-number	Unique identification string to identify the chassis serial number.

Command Default

On hardware platforms, where the serial number can be read by the device, the default is the serial number. For example, some Cisco devices have default chassis ID values of their serial numbers.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server chassis-id** command to provide a message line identifying the SNMP server serial number.

The chassis ID message can be displayed with the **show snmp** command.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to specify the chassis serial number 1234456:

```
RP/0/RP0/CPU0:router# snmp-server chassis-id 1234456
```

snmp-server community

To configure the community access string to permit access to the Simple Network Management Protocol (SNMP), use the **snmp-server community** command in

global configuration

mode. To remove the specified community string, use the **no** form of this command.

snmp-server community [**clear** | **encrypted**] *community-string* [**view** *view-name*] [**RO** | **RW**] [**SDROwner** | **SystemOwner**] [*access-list-name*]
no snmp-server community *community-string*

Syntax Description

clear	(Optional) Specifies that the entered *community-string* is clear text and should be encrypted when displayed by the **show running** command.
encrypted	(Optional) Specifies that the entered *community-string* is encrypted text and should be displayed as such by the **show running** command.
community-string	Community string that acts like a password and permits access to the SNMP protocol. The maximum length of the *community-string* argument is 32 alphabetic characters. If the **clear** keyword was used, *community-string* is assumed to be clear text. If the **encrypted** keyword was used, *community-string* is assumed to be encrypted. If neither was used, *community-string* is assumed to be clear text.
view *view-name*	(Optional) Specifies the name of a previously defined view. The view defines the objects available to the community.
RO	(Optional) Specifies read-only access. Authorized management stations are able only to retrieve MIB objects.
RW	(Optional) Specifies read-write access. Authorized management stations are able both to retrieve and to modify MIB objects.
SDROwner	(Optional) Limits access to the owner service domain router (SDR).
SystemOwner	(Optional) Provides system-wide access.
access-list-name	(Optional) Name of an access list of IP addresses allowed to use the community string to gain access to the SNMP agent.

Command Default

By default, an SNMP community string permits read-only access to all MIB objects.

By default, a community string is assigned to the SDR owner.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server community** command to configure the community access string to permit access to SNMP.

To remove the specified community string, use the **no** form of this command.

Use the **clear** keyword to specify that the clear text community string you enter is displayed encrypted in the **show running** command output. To enter an encrypted string, use the **encrypted** keyword. To enter a clear text community string that is not encrypted by the system, use neither of these keywords.

When the **snmp-server community** command is entered with the **SDROwner** keyword, SNMP access is granted only to the MIB object instances in the owner SDR.

When the **snmp-server community** command is entered with the **SystemOwner** keyword, SNMP access is granted to the entire system.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to assign the string comaccess to SNMP, allowing read-only access, and to specify that IP access list 4 can use the community string:

```
RP/0/RP0/CPU0:router(config)# snmp-server community comaccess ro 4
```

The following example shows how to assign the string mgr to SNMP, allowing read-write access to the objects in the restricted view:

```
RP/0/RP0/CPU0:router(config)# snmp-server community mgr view restricted rw
```

This example shows how to remove the community comaccess:

```
RP/0/RP0/CPU0:router(config)#no snmp-server community comaccess
```

snmp-server community-map

To associate a Simple Network Management Protocol (SNMP) community with an SNMP context, security name, or a target-list use the **snmp-server community-map** command in

global configuration

mode. To change an SNMP community mapping to its default mapping, use the **no** form of this command.

snmp-server community-map [**clear** | **encrypted**] *community-string* [**context** *context-name*] [**security-name** *security-name*] [**target-list** *target*]
no snmp-server community-map [**clear** | **encrypted**] *community-string*

Syntax Description

clear	(Optional) Specifies that the *community-string* argument is clear text.
encrypted	(Optional) Specifies that the community-string argument is encrypted text.
community-string	Name of the community.
context *context-name*	(Optional) Name of the SNMP context to which this community name is to be mapped.
security-name *security-name*	(Optional) Security name for this community. By default, the *string* is the security name.
target-list *target*	(Optional) Name of the target list for this community.

Command Default

The value of the *community-string* argument is also the security name.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server community-map** command to map an SNMPv1 or SNMPv2c community name to one or more of the following:

- **context name**—Maps a community name to a specific SNMP context name. This allows MIB instances in an SNMP context to be accessed through SNMPv1 or SNMPv2c using this community name.
- **security name**—By default, the community name is used to authenticate SNMPv1 and SNMPv2c. Configure a security name for a community name to override the default and authenticate SNMP with the security name.
- **target**—Target list identifies a list of valid hosts from which SNMP access can be made using a specific security name. When such mapping is done for a particular community name, SNMP access is allowed only from hosts included in the target list.

Use the **clear** keyword to specify that the clear text community string you enter is displayed encrypted in the **show running** command output. To enter an encrypted string, use the **encrypted** keyword. To enter a clear text community string that is not encrypted by the system, use neither of these keywords.

Task ID

Task ID	Operations
snmp	read, write

This example maps the community name “sample 2” to the SNMP context name “sample1”:

```
RP/0/RP0/CPU0:router(config)# snmp-server community-map sample2 context sample1
```

snmp-server contact

To set the Simple Network Management Protocol (SNMP) system contact, use the **snmp-server contact** command in

global configuration

mode. To remove the system contact information, use the **no** form of this command.

snmp-server contact *system-contact-string*
no snmp-server contact

Syntax Description

system-contact-string	String that describes the system contact information. The maximum string length is 255 alphanumeric characters.

Command Default

No system contact is set.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server contact** command to set the system contact string. Use the **no** form of this command to remove the system contact information.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to specify a system contact string:

```
RP/0/RP0/CPU0:router(config)# snmp-server contact Dial System Operator at beeper # 27345
```

snmp-server context

To create a Simple Network Management Protocol (SNMP) context, use the **snmp-server context** command in

global configuration

mode. To remove an SNMP context, use the **no** form of this command.

snmp-server context *context-name*
no snmp-server context *context-name*

Syntax Description

context-name	Name of the SNMP context.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This command creates an SNMP context. By default, all the SNMP MIB instances are in a default context. Create an SNMP context and map it to a particular feature to enable similar instances of the same object to co-exist in different SNMP contexts.

Task ID

Task ID	Operations
snmp	read, write

This example creates a new SNMP context named “sample1:”

```
RP/0/RP0/CPU0:router(config)# snmp-server context sample1
```

snmp-server context mapping

To map an SNMP context with a protocol instance, topology or VRF entity, use the **snmp-server context mapping** command in global configuration mode.

snmp-server context mapping *context-name* [**feature** *feature-name*] [**instance** *instance-name*] [**topology** *topology-name*] [**vrf** *vrf-name*]

Syntax Description

context-name	Name of the SNMP context.
feature *feature-name*	Specifies the protocol for which to map the context. Available options are: • **bridge**—Layer 2 VPN bridge • **vrf**—Virtual Routing and Forwarding
instance *instance-name*	Maps the context to the specified protocol instance.
topology *topology-name*	Maps the context to the specified protocol topology.
vrf *vrf-name*	Maps the context to the specified VRF logical entity.

Command Default

No context mappings exist by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

A device can support multiple instances of a logical network entity, such as protocol instances or VRFs. Most existing MIBs cannot distinguish between these multiple logical network entities. For example, the original OSPF-MIB assumes a single protocol instance on a device, but you can now configure multiple OSPF instances on a device.

The **snmp-server context mapping** command maps a context to a protocol instance, topology or VRF logical entity.



Note The snmp-server context mapping command does not work for OSPF and OSPFv3. Refer to the **snmp context** commands.

Task ID

Task ID	Operation
snmp	read, write

This example illustrates how to map an snmp context to an OSPF instance:

```
RP/0/RP0/CPU0:router(config)# snmp-server context mapping con5 feature ospf instance in1
```

snmp-server drop report acl

To apply an ACL policy for restricting an SNMPv3 unknown engine-id report to be sent out to NMS, use the **snmp-server drop report acl** command in the configuration mode.

snmp-server drop report acl IPv4 *IPv4-acl-name* **IPv6** *IPv6-acl-name*

Syntax Description

acl	Specifies IP Access Control Lists (ACL) policy
IPv4 *IPv4-acl-name*	Defines an IPv4 ACL name.
IPv6 *IPv6-acl-name*	Defines an IPv6 ACL name.

Command Default

Unknown engine-id reports will be sent to all polling stations (even if other ACLs are configured).

Command Modes

Configuration mode

Command History

Release	Modification
Release 6.2.3	This command was introduced.

Usage Guidelines

To drop an unknown engine-id report, you can either configure IPv4/IPv6 ACL name or both. When router is polled with wrong engine-id or no engine-id during a snmpv3 packet exchange, the unknown engine-id report will be sent based on the ACL policy that is configured.

Unknown engine-id reports will be sent only to polling station addresses that are permitted by ACL.

Task ID

Task ID	Operation
snmp	read, write

Example

This example shows how to configure the SNMP server to drop the unknown engin-id report:

```
RP/0/RP0/CPU0:router (config) #  snmp-server drop report acl IPv4 nms-block IPv6
nms-block-ipv6
```

snmp-server drop unknown-user

To avoid error PDUs being sent out of router when polled with incorrect SNMPv3 user name, use the **snmp-server drop unknown-user** command in the appropriate mode. If the configuration is not set, by default it will respond with error PDUs.

snmp-server drop unknown-user

Syntax Description

drop unknown-user	Drop the error PDUs to be sent when router is polled with incorrect SNMPv3 user name.

Command Default

Unknown error PDUs will be sent when router is polled with incorrect SNMPv3 user name.

Command Modes

XR config

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

Example

This example shows how to configure the SNMP server to drop the error PDUs:

```
RP/0/RP0/CPU0:router (config) #  snmp-sever drop unknown-user
```

snmp-server engineid local

To specify Simple Network Management Protocol (SNMP) engine ID on the local device, use the **snmp-server engineid local** command in

global configuration

mode. To return the engine ID to the default, use the **no** form of this command.

snmp-server engineid local *engine-id*
no snmp-server engineid local *engine-id*

Syntax Description

engine-id	Character string that identifies the engine ID. Consists of up to 24 characters in hexadecimal format. Each hexadecimal number is separated by a colon (:).

Command Default

An SNMP engine ID is generated automatically.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure the SNMP engine ID on the local device:

```
RP/0/RP0/CPU0:router(config)# snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61
```

snmp-server engineid remote

To specify a Simple Network Management Protocol (SNMP) engine ID on a remote device, use the **snmp-server engineid remote** command in

global configuration

mode. To return the engine ID to the default, use the **no** form of this command.

snmp-server engineid remote *ip-address engine-id* **udp-port** *port*
no snmp-server engineid remote *ip-address engine-id* **udp-port** *port*

Syntax Description

ip-address	IP address of remote SNMP notification host
engine-id	Character string that identifies the engine ID. Consists of up to 24 characters in hexadecimal format. Each hexadecimal number is separated by a colon (:).
udp-port *port*	(Optional) Specifies the User Datagram Protocol (UDP) port of the host to use. Range is from 1 to 65535. The default UDP port is 161.

Command Default

An SNMP engine ID is generated automatically.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

This example shows how to configure the SNMP engine ID on the local device:

```
RP/0/RP0/CPU0:Router(config)# snmp-server engineID remote 172.16.4.1
00:00:00:09:00:00:00:a1:61:6c:20:61
```

snmp-server entityindex persist

To enable the persistent storage of ENTITY-MIB data across process restarts, switchovers, and device reloads, use the **snmp-server entityindex persist** command in

global configuration

mode. To disable the persistent storage of ENTITY-MIB data, use the **no** form of this command.

snmp-server entityindex persist
no snmp-server entityindex persist

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

Example

This example illustrates how to enable persistent storage of ENTITY-MIB indices:

```
RP/0/RP0/CPU0:router(config)# snmp-server entityindex persist
```

snmp-server group

To configure a new Simple Network Management Protocol (SNMP) group, or a table that maps SNMP users to SNMP views, use the **snmp-server group** command in

global configuration

XR Config

mode. To remove a specified SNMP group, use the **no** form of this command.

snmp-server group *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [**context** *context-name*] [*access-list-name*]
no snmp-server group *name*

Syntax Description

name	Name of the group.
v1	Specifies a group that uses the SNMPv1 security model. The SNMP v1 security model is the least secure of the possible security models.
v2c	Specifies a group that uses the SNMPv2c security model. The SNMPv2c security model is the second least secure of the possible security models.
v3	Specifies a group that uses the SNMPv3 security model. The SNMP v3 security is the most secure of the possible security models.
auth	Specifies authentication of a packet without encrypting it.
noauth	Specifies no authentication of a packet.
priv	Specifies authentication of a packet with encryption.
read *view*	(Optional) Specifies a read view string (not to exceed 64 characters) that is the name of the view that allows only the contents of the agent to be viewed.
write *view*	(Optional) Specifies a write view string (not to exceed 64 characters) that is the name of the view used to enter data and configure the contents of the agent.
notify *view*	(Optional) Specifies a notify view string (not to exceed 64 characters) that is the name of the view used to specify a notify or trap.
context *context-name*	(Optional) Specifies the SNMP context to associate with this SNMP group and associated views.
access-list-name	(Optional) Access list string (not to exceed 64 characters) that is the name of the access list.

Command Default

See snmp-server group, on page 361.

Command Modes

Global configuration

XR Config

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

This table describes the default values for the different views:

Table 36: snmp-server group Default Descriptions

Default	Definition
read *view*	Assumed to be every object belonging to the Internet (1.3.6.1) object identifier (OID) space, unless the user uses the **read** option to override this state.
write *view*	Nothing is defined for the write view (that is, the null OID). You must configure write access.
notify *view*	Nothing is defined for the notify view (that is, the null OID). If a view is specified, any notifications in that view that are generated are sent to all users associated with the group (provided an SNMP server host configuration exists for the user).

Configuring Notify Views

Do not specify a notify view when configuring an SNMP group for the following reasons:

- The **snmp-server host** command autogenerates a notify view for the user, and then adds it to the group associated with that user.
- Modifying the notify view of the group affects all users associated with that group.

The notify view option is available for two reasons:

- If a group has a notify view that is set using SNMP, you may need to change the notify view.
- The **snmp-server host** command may have been configured before the **snmp-server group** command. In this case, reconfigure the **snmp-server host** command or specify the appropriate notify view.

Instead of specifying the notify view for a group as part of the **snmp-server group** command, use the following commands in global configuration mode:

- **snmp-server user**—Configures an SNMP user.
- **snmp-server group**—Configures an SNMP group, without adding a notify view.
- **snmp-server host**—Autogenerates the notify view by specifying the recipient of a trap operation.

Working with Passwords and Digests

No default values exist for authentication or privacy algorithms when this command is configured. In addition, no default passwords exist. The minimum length for a password is one character, although we recommend using eight characters for security. A plain-text password or localized Message Digest 5 (MD5) password can be specified. Forgotten passwords cannot be recovered, and the user must be reconfigured.

SNMP Contexts

SNMP contexts provide Virtual Private Network (VPN) users with a secure way of accessing MIB data. When a VPN is associated with a context, that VPN's specific MIB data exists in that context. Associating a VPN with a context enables service providers to manage networks with multiple VPNs. Creating and associating

a context with a VPN enables a provider to prevent the users of one VPN from accessing information about users of other VPNs on the same networking device.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to configure an SNMP version 3 group named group1 that requires the authentication of packets with encryption:

```
Router(config)# snmp-server group group1 v3 priv
```

snmp-server host

To specify the recipient of a Simple Network Management Protocol (SNMP) notification operation, use the **snmp-server host** command in

Global configuration

To remove the specified host, use the **no** form of this command.

snmp-server host *address* [**clear** | **encrypted**] [**informs**] [**traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [**udp-port** *port*] [*notification-type*]
nosnmp-server host *address* [**clear** | **encrypted**] [**informs**] [**traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [**udp-port** *port*] [*notification-type*]

Syntax Description

address	Name or IP address of the host (the targeted recipient).
clear	(Optional) Specifies that the *community-string* argument is clear text.
encrypted	(Optional) Specifies that the *community-string* argument is encrypted text.
informs	(Optional) Specifies to send inform messages to this host.
traps	(Optional) Specifies that notifications should be sent as traps. This is the default.
version	(Optional) Specifies the version of the SNMP used to send the traps.
1	Specifies SNMPv1, the default.
2c	Specifies SNMPv2C.
3	Specifies SNMPv3. Version 3 is the most secure model because it allows packet encryption. If you specify the SNMPv3 keyword, you must specify the security level.
auth	Enables Message Digest 5 (MD5) algorithm and Secure Hash Algorithm (SHA) packet authentication.
noauth	Specifies that the noAuthNoPriv security level applies to this host. This is the default security level for SNMPv3.
priv	Enables Data Encryption Standard (DES) packet encryption (also called "privacy").
community-string	Password-like community string sent with the notification operation. We recommend defining this string using the **snmp-server community** command prior to using the **snmp-server host** command.
udp-port *port*	(Optional) Specifies the User Datagram Protocol (UDP) port of the host to use. Range is from 1 to 65535. The default UDP port is 161.

notification-type	(Optional) Type of notification to be sent to the host. If no type is specified, all available notifications are sent. The notification type can be one or more of these keywords: • **bgp** —Enables SNMP Border Gateway Protocol Version 4 (BGPv4) traps. • **config** —Controls configuration notifications, as defined in the CISCO-CONFIG-MAN-MIB (enterprise 1.3.6.1.4.1.9.9.43.2). The notification type is (1) ciscoConfigManEvent. • **copy-complete** —Enables CISCO-CONFIG-COPY-MIB ccCopyCompletion traps. • **entity** —Controls Entity MIB modification notifications. This notification type is defined in the ENTITY-MIB (enterprise 1.3.6.1.2.1.47.2) as: (1) entConfigChange. • **fabric** —Enables SNMP fabric traps. • **fru-ctrl** —Enables SNMP entity field-replaceable unit (FRU) control traps. • **mpls** —Enables SNMP Multiprotocol Label Switching (MPLS) traps. • **sensor** —Enables SNMP entity sensor traps. • **snmp** —Enables SNMP traps. • **syslog** —Controls error message notifications (Cisco-syslog-MIB). Specify the level of messages to be sent with the **logging history** command.

Command Default

This command is disabled by default. No notifications are sent.

The default UDP port is 161.

When this command is entered without keywords, the default is to send all trap types to the host.

If no version keyword is entered, the default is version 1.

If version 3 is specified, but the security level is not specified, the default security level is noauth.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps. Traps are unreliable because the receiver does not send acknowledgments when it receives traps. The sender cannot determine if the traps were received. Traps are discarded as soon as they are sent. Traps are also sent only once.

When the **snmp-server host** command is not entered, no notifications are sent. To configure the device to send SNMP notifications, configure at least one **snmp-server host** command. When the command is entered without keywords, all trap types are enabled for the host.

To enable multiple hosts, issue a separate **snmp-server host** command for each host. You can specify multiple notification types in the command for each host.

When multiple **snmp-server host** commands are given for the same host and kind of notification (trap), each succeeding **snmp-server host** command overwrites the previous command. Only the last **snmp-server host** command is in effect. For example, if an **snmp-server host** command with the **traps** keyword is entered for a host and then another command with the **traps** keyword is entered for the same host, the second command replaces the first.

Either a host name or IP address can be used to specify the host. Both IPv4 and IPv6 IP address formats are supported.

The **snmp-server host** command is used with the **snmp-server engineid** command. Use the **snmp-server traps** command to specify which SNMP notifications are sent globally. For a host to receive most notifications, at least one **snmp-server traps** command and the **snmp-server host** command for that host must be enabled.

However, some notification types cannot be controlled with the **snmp-server traps** command. For example, some notification types are always enabled. Other notification types are enabled by a different command.

The availability of a notification-type depends on the device type and Cisco software features supported on the device.

To display which notification types are available on the system, use the question mark (**?**) online help function at the end of the **snmp-server host** command.

Previously, the SNMP agent would drop traps after 5 minutes if the respective hosts were unreachable. In its new behavior, the SNMP agent now sends traps to all unreachable hosts instead of dropping them. There is no change in the trap delay timer behavior.

The **no snmp-server host** command used with no keywords disables traps.

Use the **clear** keyword to specify that the clear text community string you enter is displayed encrypted in the **show running** command output. To enter an encrypted string, use the **encrypted** keyword. To enter a clear text community string that is not encrypted by the system, use neither of these keywords.

If the **informs** keyword is used, the SNMP version can be only SNMPv2C or SNMPv3.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to send RFC 1157 SNMP traps to the host specified by the name myhost.cisco.com. Other traps are enabled, but only SNMP traps are sent because only the **snmp** keyword is specified in the **snmp-server host** command. The community string is defined as comaccess.

```
Router(config)# snmp-server traps
Router(config)# snmp-server host myhost.cisco.com comaccess snmp
```

This example shows how to send the SNMP traps to address 172.30.2.160:

```
Router(config)# snmp-server traps snmp
Router(config)# snmp-server host 172.30.2.160 public snmp
```

This example shows how to enable the router to send all traps to the host, myhost.cisco.com, using the community string public:

```
Router(config)# snmp-server traps
Router(config)# snmp-server host myhost.cisco.com public
```

This example shows how to prevent traps from being sent to any host. The BGP traps are enabled for all hosts, but only the configuration traps are enabled to be sent to a host.

```
Router(config)# snmp-server traps bgp
Router(config)# snmp-server host hostabc public config
```

This example shows how to send SNMPv3 informs to a host:

```
Router(config)# snmp-server host 172.30.2.160 informs version 3
```

snmp-server ifindex persist

To enable ifIndex persistence globally on all Simple Network Management Protocol (SNMP) interfaces, use the **snmp-server ifindex persist** command in global configuration mode. To disable global interface persistence, use the **no** form of this command.

snmp-server ifindex persist
no snmp-server ifindex persist

Syntax Description

This command has no keywords or arguments.

Command Default

Global interface persistence is disabled.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server ifindex persist** command to enable ifIndex persistence on all interfaces that have entries in the ifIndex table of the IF-MIB. When enabled, this command retains the mapping between the ifName object values and the ifIndex object values (generated from the IF-MIB) persistent during reloads, allowing for consistent identification of specific interfaces using SNMP. Applications such as device inventory, billing, and fault detection depend on this feature.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable ifIndex persistence globally:

```
RP/0/RP0/CPU0:router(config)# snmp-server ifindex persist
```

snmp-server ifmib internal cache max-duration

To configure the refresh interval for the IF-MIB statistics cache, use the **snmp-server ifmib internal cache max-duration** command in global configuration mode. To revert to the default cache interval, use the **no** form of this command.

snmp-server ifmib internal cache max-duration *timeout*

Syntax Description

timeout	Length of time before the cache is refreshed. Values can range from 0 to 60 seconds. The default is 15.

Command Default

timeout: 15 seconds

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **snmp-server ifmib internal cache max-duration** command controls the refresh interval of the cache. If the *timeout* value in the **snmp-server ifmib internal cache max-duration** command is set to zero, the cache is disabled. By default, the counters are cached for 15 secs in the ifmib internal cache, after which it will be discarded.

Task ID

Task ID	Operation
snmp	read, write

This example shows how to change the refresh interval for the IF-MIB statistics cache.

```
RP/0/RP0/CPU0:routerrouter(config)# snmp-server ifmib internal cache max-duration 60
```

snmp-server ifmib stats cache

To enable retrieval of cached statistics instead of real-time statistics, use the **snmp-server ifmib stats cache** command. To revert to the default, use the **no** form of this command.

snmp-server ifmib stats cache
no snmp-server ifmib stats cache

Syntax Description

This command has no keywords or arguments.

Command Default

Cached statistics are not enabled.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Cisco IOS XR statistics infrastructure maintains a cache of statistics for all interfaces. This cache is updated every 30 seconds. Use the **snmp-server ifmib stats cache** command to enable the IF-MIB to retrieve these cached statistics rather than real-time statistics. Accessing cached statistics is less CPU-intensive than accessing real-time statistics.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the IF-MIB caches statistics:

```
RP/0/RP0/CPU0:router(config)# snmp-server ifmib stats cache
RP/0/RP0/CPU0:router(config)# exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes
RP/0/RP0/CPU0:router#
```

snmp-server inform

To configure Simple Network Management Protocol (SNMP) inform message options, use the **snmp-server inform** command in global configuration mode. To revert to the default informs options, use the **no** form of this command.

snmp-server inform {**pending** *max-no* | **retries** *no-retries* | **timeout** *seconds*}
no snmp-server inform {**pending** *max-no* | **retries** *no-retries* | **timeout** *seconds*}

Syntax Description

pending *max-no*	Specifies the maximum number of inform messages to hold in the queue. The default is 25.
retries *no-retries*	Specifies the retry count for inform messages. Values can be from 1 to 100. The default is three.
timeout *seconds*	Specifies the inform message timeout value in seconds. The default is 15.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To enable the sending of SNMP inform messages, use the **snmp-server host** command with the **informs** keyword. When SNMP server informs are enabled, the SNMP version can be only SNMPv2C or SNMPv3.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure SNMP inform messages:

```
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com informs comaccess
RP/0/RP0/CPU0:router(config)# snmp-server inform pending 40
RP/0/RP0/CPU0:router(config)# snmp-server inform retries 10
```

snmp-server interface

To enable an interface to send Simple Network Management Protocol (SNMP) trap notifications and enter SNMP interface configuration mode, use the **snmp-server interface** command in global configuration mode. To disable the sending of SNMP trap notifications on an interface, use the **no** form of this command.

snmp-server interface *type interface-path-id*
no snmp-server interface *type interface-path-id*

Syntax Description

type	Interface type. For more information, use the question mark (**?**) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

Ethernet interfaces are enabled to send SNMP trap notifications. SNMP trap notifications are disabled on all other physical and logical interfaces.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **snmp-server interface** command enters SNMP interface configuration mode for you to configure the available SNMP options.

Note In references to a Management Ethernet interface located on a route processor card, the physical slot number is and the module is CPU0. Example: interface .

Task ID

Task ID	Operations
snmp	read, write

This example shows how to assign ifIndex persistence on Packet-over-SONET/SDH (POS) interface 0/0/1/0:

```
RP/0/RP0/CPU0:router(config)# snmp-server interface pos 0/0/1/0
```

```
RP/0/RP0/CPU0:router(config-snmp-if)#
```

snmp-server interface subset

To enter snmp-server interface subset configuration mode for a set of interfaces, use the **snmp-server interface subset** command in global configuration mode. To revert to the default interface settings, use the **no** form of this command.

snmp-server interface subset *subset-number* **regular-expression** *expression*
no snmp-server interface subset *subset-number*

Syntax Description

subset-number	Identifying number of the interface subset, which also indicates its relative priority.
regular-expression *expression*	Specifies for which subset of interfaces to enter snmp-server interface subset configuration mode. The *expression* argument must be entered surrounded by double quotes.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The *subset-number* argument is used to set the priority for an interface that matches more than one configured regular expressions. Lower values of the *subset-number* have a higher priority. If a single interface becomes part of a multiple-interface configured regular expression, the configuration with the lower *subset-number* value is applied.

Regular expressions have two constraints:

- Regular expressions must always be entered within double quotes to ensure that the CLI interprets each character correctly.
- All characters that are part of a regular expression are considered regular characters with no special meaning. In order to enter special characters, such as "\" or "?," they must be preceded by the backslash character "\." For example, to enter the regular expression ([A-Z][A-Z0-9]*)\b[^>]*>(.*?)</\1, you would enter ([A-Z][A-Z0-9]*)\\b[^>]*>(.*\?)</\\1.

Refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in for more information regarding regular expressions.

From the snmp-server interface mode of a subset of interfaces, SNMP linkUp and linkDown notifications can be enabled or disabled using the **notification linkupdown disable** command.

Task ID

Task ID	Operation
snmp	read, write

This example illustrates how to configure all Gigabit Ethernet interfaces:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# snmp-server int subset 2
    regular-expression "^Gig[a-zA-Z]+[0-9/]+\."
RP/0/RP0/CPU0:router(config-snmp-if-subset)#
```

snmp-server ipv4 dscp

To mark packets with a specific differentiated services code point (DSCP) value, use the **snmp-server ipv4 dscp** command in global configuration mode. To remove matching criteria, use the **no** form of this command.

snmp-server ipv4 dscp *value*
no snmp-server ipv4 dscp [*value*]

Syntax Description

value	Value of the DSCP. The DSCP value can be a number from 0 to 63, or it can be one of the following keywords: **default**, **ef**, **af11**, **af12**, **af13**, **af21**, **af22**, **af23**, **af31**, **af32**, **af33**, **af41**, **af42**, **af43**, **cs1**, **cs2**, **cs3**, **cs4**, **cs5**, **cs6**, **cs7**.

Command Default

The IP DSCP default value for SNMP traffic is 0.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server ipv4 dscp** command to specify an IP DSCP value to give SNMP traffic higher or lower priority in your network.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure the DSCP value to af32:

```
RP/0/RP0/CPU0:router(config)# snmp-server ipv4 dscp af32
```

snmp-server ipv4 precedence

To mark packets with a specific precedence level to use for packet matching, use the **snmp-server ipv4 precedence** command in global configuration mode. To restore the system to its default interval values, use the **no** form of this command.

snmp-server ipv4 precedence *value*
no snmp-server ipv4 precedence [*value*]

Syntax Description

value Value of the precedence. The precedence value can be a number from 0 to 7, or it can be one of the following keywords:

critical

Set packets with critical precedence (5)

flash

Set packets with flash precedence (3)

flash-override

Set packets with flash override precedence (4)

immediate

Set packets with immediate precedence (2)

internet

Set packets with internetwork control precedence (6)

network

Set packets with network control precedence (7)

priority

Set packets with priority precedence (1)

routine

Set packets with routine precedence (0)

Command Default

The IP Precedence default value for SNMP traffic is 0.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server ipv4 precedence** command to specify an IP Precedence value to give SNMP traffic higher or lower priority in your network.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to set the precedence to 2:

```
RP/0/RP0/CPU0:router(config)# snmp-server ipv4 precedence 2
```

snmp-server location

To specify the system location for Simple Network Management Protocol (SNMP), use the **snmp-server location** command in

global configuration

mode. To remove the location string, use the **no** form of this command.

snmp-server location *system-location*
no snmp-server location

Syntax Description

system-location	String indicating the physical location of this device. The maximum string length is 255 alphanumeric characters.

Command Default

No system location string is set.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to specify a system location string:

```
RP/0/RP0/CPU0:router(config)# snmp-server location Building 3/Room 214
```

snmp-server mibs cbqosmib persist

To enable persistent storage of the CISCO-CLASS-BASED-QOS-MIB data across process restarts, switchovers, and device reloads, use the **snmp-server mibs cbqosmib persist** command in global configuration mode. To disable persistent storage of the MIB data, use the **no** form of this command.

snmp-server mibs cbqosmib persist
no snmp-server mibs cbqosmib persist

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

Example

This example illustrates how to enable persistent storage of CISCO-CLASS-BASED-QOS-MIB data:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib persist
```

snmp-server mibs cbqosmib cache

To enable and configure caching of the QoS MIB statistics, use the **snmp-server mibs cbqosmib cache** command in global configuration mode. To disable caching, use the **no** form of this command.

snmp-server mibs cbqosmib cache {**refresh time** *time* | **service-policy count** *count*}
no snmp-server mibs cbqosmib cache [**refresh time** *time* | **service-policy count** *count*]

Syntax Description

refresh	Enables QoS MIB caching with a specified cache refresh time.
time *time*	Specifies the cache refresh time, in seconds. The *time* argument can be between 5 and 60. The default is 30.
service-policy	Enables QoS MIB caching with a limited number of service policies to cache.
count *count*	Specifies the maximum number of service policies to cache. The count argument can be between 1 and 5000.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

Example

This example illustrates how to enable QoS MIB caching with a refresh time:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache refresh time 45
```

This example illustrates how to enable QoS MIB caching with a service policy count limitation:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache service-policy count 10
```

snmp-server mibs eventmib congestion-control

To configure the generation of SNMP traps when congestion exceeds configured thresholds, use the **snmp-server mibs eventmib congestion-control** command in global configuration mode. To restore the default values, use the **no** form of this command.

snmp-server mibs eventmib congestion-control *type interface-path-id* **falling** *lower-threshold* **interval** *sampling-interval* **rising** *upper-threshold*
no snmp-server mibs eventmib congestion-control *type interface-path-id*

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the show interfaces command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
falling *lower-threshold*	Specifies the lower threshold for which to determine whether an mteTriggerFalling SNMP Trap is generated.
interval *sampling-interval*	Specifies how often the congestion statistics are polled. The *interval* argument, in minutes, can be between 5 and 1440; it must be a multiple of 5.
rising *upper-threshold*	Specifies the upper threshold for which to determine whether an mteTriggerRising SNMP Trap is generated.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Note A maximum of 100 interfaces can be monitored for congestion.

Congestion configurations using the **snmp-server mibs eventmib congestion-control** command cannot be modified using SNMP SET and vice versa.

When the congestion between two intervals increases above the *upper-threshold* argument, an mteTriggerRising SNMP trap is generated. This trap is not generated until the congestion drops below the lower threshold and then rises above the upper threshold.

When the congestion between two intervals falls below the *lower-threshold* argument, and an SNMP mteTriggerRising trap was generated previously, an SNMP mteTriggerFalling trap is generated. The mteTriggreRising trap is not generated until the congestion goes above the upper threshold and then falls back below the lower threshold.

The *lower-threshold* value (falling) should be set to a value less than or equal to the *upper-threshold* value (rising).

The **snmp-server mibs eventmib congestion-control** command is configured on a specific interface and is supported on the following cards:

- 8-port 10 Gigabit Ethernet PLIM
- 16-port OC-48c/STM-16 POS/DPT PLIM
- 1-port OC-768c/STM-256 POS PLIM
- 4-port OC-192c/STM-64 POS/DPT PLIM
- All Ethernet SPAs
- 2-port and 4-port OC-3c/STM-1 POS SPAs
- 2-port, 4-port, and 8-port OC-12c/STM-4 POS SPAs
- 2-port and 4-port OC-48c/STM-16 POS/RPR SPAs
- 1-port OC-192c/STM-64 POS/RPR SPA

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure the generation of SNMP traps in response to congestion:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs eventmib congestion-control pos 0/1/0/0
    falling 1 interval 5 rising 2
```

snmp-server mibs eventmib packet-loss

To configure the generation of SNMP traps when packet loss exceeds configured thresholds, use the **snmp-server mibs eventmib packet-loss** command in global configuration mode. To restore the default values, use the **no** form of this command.

snmp-server mibs eventmib packet-loss *type interface-path-id* **falling** *lower-threshold* **interval** *sampling-interval* **rising** *upper-threshold*
no snmp-server mibs eventmib packet-loss *type interface-path-id*

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the show interfaces command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
falling *lower-threshold*	Specifies the lower threshold for which to determine whether an mteTriggerFalling SNMP Trap is generated.
interval *sampling-interval*	Specifies how often the packet loss statistics are polled. The *interval* argument, in minutes, can be between 5 and 1440; it must be a multiple of 5.
rising *upper-threshold*	Specifies the upper threshold for which to determine whether an mteTriggerRising SNMP Trap is generated.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Note A maximum of 100 interfaces can be monitored for packet loss.

Packet loss configurations using the **snmp-server mibs eventmib packet-loss** command cannot be modified using SNMP SET and vice versa.

When the packet loss between two intervals increases above the *upper-threshold* argument, an mteTriggerRising SNMP trap is generated. This trap is not generated until the packet loss drops below the lower threshold and then rises above the upper threshold.

When the packet loss between two intervals falls below the *lower-threshold* argument, and an SNMP mteTriggerRising trap was generated previously, an SNMP mteTriggerFalling trap is generated. The mteTriggreRising trap is not generated until the packet loss goes above the upper threshold and then falls back below the lower threshold.

The *lower-threshold* value (falling) should be set to a value less than or equal to the *upper-threshold* value (rising).

The **snmp-server mibs eventmib packet-loss** command is configured on a specific interface and is supported on the following cards:

- 8-port 10 Gigabit Ethernet PLIM
- 16-port OC-48c/STM-16 POS/DPT PLIM
- 1-port OC-768c/STM-256 POS PLIM
- 4-port OC-192c/STM-64 POS/DPT PLIM
- All Ethernet SPAs
- 2-port and 4-port OC-3c/STM-1 POS SPAs
- 2-port, 4-port, and 8-port OC-12c/STM-4 POS SPAs
- 2-port and 4-port OC-48c/STM-16 POS/RPR SPAs
- 1-port OC-192c/STM-64 POS/RPR SPA

Task ID

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure the generation of SNMP traps in response to packet loss:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs eventmib packet-loss pos 0/1/0/0
    falling 1 interval 5 rising 2
```

snmp-server mibs sensormib cache

To enable and configure caching for sensor mib values, use **snmp-server mibs sensormib cache** command in global configuration mode. To restore the default values, use the **no** form of this command.

snmp-server mibs sensormib cache

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global configuration mode.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Example

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs sensormib cache
```

snmp-server mibs subscriber threshold access-if

To disable the per-session access notifications by the session monitoring process, use the **snmp-server mibs subscriber threshold access-if** in the global configuration mode. To enable notifications, use the **no** form of the command.

snmp-server mibs subscriber threshold access-if subset*number* **regular expression** *word* **notification rising-falling disable**
nosnmp-server mibs subscriber threshold access-if

Syntax Description

subset *number*	Subset number of the subscriber threshold. Lower the subset value, higher is the priority. Range is 1 to 255.
regular expression *word*	Regular expression to match the interface name. Traps on the corresponding access interface(s) are disabled.
notification	Name of the notification.
rising-falling	The rising and falling thresholds.
disable	Disables the access interface notifications.

Command Default

Session monitoring is enabled by default

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **regular expression** keyword disables notifications of the access-interface.

Task ID

Task ID	Operation
snmp	read, write

Example

```
RP/0/RP0/CPU0:router (config) # snmp-server mibs subscriber threshold access-if subset 100
 regular expression notification rising-falling disable
```

snmp-server mibs subscriber threshold

To set the snmp-server mibs server threshold parameters, use the **snmp-server mibs subscriber threshold** command in the global configuration mode. To delete any of the set parameters, use the **no** form of the command.

snmp-server mibs subscriber threshold [**rising** | **falling** | **delta-loss percent** | **delta-loss evaluation**] [**access-if location** *interface-path-id* **interval** *seconds*] [**session-count**]

nosnmp-server mibs subscriber threshold

Syntax Description

rising	Rising threshold value. The set value triggers the traps. Traps are generated when the number of sessions exceed the rising threshold value.
falling	Falling threshold value. The set value triggers the traps. Traps are generated when the number of sessions are lesser than the falling threshold value.
delta-loss percent	Delta-loss percentage.
delta-loss evaluation	The actual subscriber sessions (after delta-loss) . This is based on the set delta-loss percentage. If the number of sessions exceed the loss percentage, traps are generated.
access-if	Access-interface.
location *name*	Location name.
interval *seconds*	Interval between the rising and the falling thresholds (in seconds).
session-count	Subscriber-session count.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The minimum delta loss interval is 30 seconds and can be incremented by 10 seconds till the time period reaches 5 minutes.

Task ID

Task ID	Operation
snmp	read, write

Example

```
RP/0/RP0/CPU0:router (config) # snmp-server mibs subscriber threshold delta-loss evaluation
 access-if tengige 0/4/0/0 interval 100
```

snmp-server notification-log-mib

To configure the NOTIFICATION-LOG-MIB, use the **snmp-server notification-log-mib** command in global configuration

mode. To remove the specified configuration, use the **no** form of this command.

snmp-server notification-log-mib {**globalAgeOut** *time* | **globalSize** *size* | **default** | **disable** | **size** *size*}
no snmp-server notification-log-mib {**globalAgeOut** | **globalSize** | **default** | **disable** | **size**}

Syntax Description

globalAgeOut *time*	Specifies how much time, in minutes, a notification remains in the log. Values for the *time* argument can range from 0 to 4294967295; the default is 15.
globalSize *size*	Specifies the maximum number of notifications that can be logged in all logs. The default is 500.
default	Specifies to create a default log.
disable	Specifies to disable logging to the default log.
size *size*	Specifies the maximum number of notifications that the default log can hold. The default is 500.

Command Default

NOTIFICATION-LOG-MIB notifications are not logged.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Logging of NOTIFICATION-LOG-MIB notifications begins when the default log is created. Named logs are not supported, therefore only the default log can be created.

Task ID

Task ID	Operations
snmp	read, write

The following example creates a default log for notifications:

```
RP/0/RP0/CPU0:router(config)# snmp-server notification-log-mib default
```

This example removes the default log:

```
RP/0/RP0/CPU0:router(config)# no snmp-server notification-log-mib default
```

This example configures the size of all logs to be 1500:

```
RP/0/RP0/CPU0:router(config)# snmp-server notification-log-mib globalSize 1500
```

snmp-server packetsize

To establish control over the largest Simple Network Management Protocol (SNMP) packet size permitted when the SNMP server is receiving a request or generating a reply, use the **snmp-server packetsize** command in

global configuration

mode. To restore the default value, use the **no** form of this command.

snmp-server packetsize *size*
no snmp-server packetsize

Syntax Description

size	Packet size, in bytes. Range is from 484 to 65500. The default is 1500.

Command Default

size: 1500

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server packetsize** command to establish control over the largest SNMP packet size permitted when the SNMP server is receiving a request or generating a reply.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to set the maximum size of SNMP packets to 1024 bytes:

```
RP/0/RP0/CPU0:router(config)# snmp-server packetsize 1024
```

snmp-server queue-length

To establish the message queue length for each trap host for Simple Network Management Protocol (SNMP), use the **snmp-server queue-length** command in

global configuration

mode. To restore the default value, use the **no** form of this command.

snmp-server queue-length *length*
no snmp-server queue-length

Syntax Description

length	Integer that specifies the number of trap events that can be held before the queue must be emptied. Range is from 1 to 5000.

Command Default

length : 100

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server queue-length** command to define the length of the message queue for each trap host. After a trap message is successfully sent, Cisco IOS XR software continues to empty the queue at a throttled rate to prevent trap flooding.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to set the SNMP notification queue to 20 events:

```
RP/0/RP0/CPU0:router(config)# snmp-server queue-length 20
```

snmp-server script

To map the script file with custom OID, use the **snmp-server script** command in XR Config mode.

snmp-server script script-oid *oid-number* **script-filename** *file-name*

no snmp-server script script-oid *oid-number* **script-filename** *file-name*

Syntax Description

script-oid*oid-number*	The OID number to be added as custom OID. The custom OID number has to be followed by root OID 1.3.6.1.4.1.9.9.999998.
script-filename *file-name*	The name of the script file to be mapped.

Command Default

None

Command Modes

XR Config

Command History

Release	Modification
Release 7.5.3	This command was introduced.

Usage Guidelines

To use commands of this module, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using any command, contact your AAA administrator for assistance.

Task ID

Task ID	Operation
snmp	read,write

Examples

This example shows how to map the script file with custom OID.

```
Router(config)#snmp-server script script-oid 1.3.6.1.4.1.9.9.999998.10 script-filename
show_lldp_string.py
```

snmp-server target list

To create a Simple Network Management Protocol (SNMP) target list, use the **snmp-server target list** command in

global configuration

mode. To remove an SNMP target list, use the **no** form of this command.

snmp-server target list *target-list* {**vrf** *vrf-name* | **host** *hostname*}
no snmp-server target list *target-list*

Syntax Description

target-list	Name of the target list.
vrf *vrf-name*	Specifies the name of the VRF hosts included in the target list.
host *hostname*	Assigns a hostname to the target list. The *hostname* variable is a name or IP address.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use this command to create an SNMP target list and assign hosts to the list. When a target list is mapped to a community name using the **snmp-server community-map** command, SNMP access is restricted to the hosts in the target list (for that community name).

Task ID

Task ID	Operations
snmp	read, write

In this example, a new target list “sample3” is created and assigned to the vrf server “server2:”

```
RP/0/RP0/CPU0:router(config)# snmp-server target list sample3 vrf server2
```

snmp-server trap throttle-time

To specify the throttle time for handling more Simple Network Management Protocol (SNMP) traps, use the **snmp-server trap throttle-time** command in

global configuration

mode. To restore the throttle time to its default value, use the **no** form of this command.

snmp-server trap throttle-time *time*
no snmp-server trap throttle-time

Syntax Description

time	Throttle time in milliseconds. Values can be from 10 to 500.

Command Default

250

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read, write

In the following example, the trap throttle time is set to 500 milliseconds:

```
RP/0/RP0/CPU0:router(config)# snmp-server trap throttle-time 500
```

snmp-server trap-timeout

To define how often to try resending trap messages on the retransmission queue, use the **snmp-server trap-timeout** command in

global configuration

mode. To restore the default value, use the **no** form of this command.

snmp-server trap-timeout *seconds*
no snmp-server trap-timeout *seconds*

Syntax Description

seconds	Integer that sets the interval for resending the messages, in seconds). Value can be from 1 to 1000.

Command Default

seconds : 30

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Before Cisco IOS XR software tries to send a trap, it looks for a route to the destination address. If there is no known route, the trap is saved in a retransmission queue. Use the **snmp-server trap-timeout** command to determine the number of seconds between retransmission attempts.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to set an interval of 20 seconds to try resending trap messages on the retransmission queue:

```
RP/0/RP0/CPU0:router(config)# snmp-server trap-timeout 20
```

snmp-server traps

To enable Simple Network Management Protocol (SNMP) trap notifications, use the **snmp-server traps** command in

global configuration

mode. To disable SNMP notifications, use the **no** form of this command.

snmp-server traps *notification-type*
no snmp-server traps [*notification-type*]

Syntax Description *notification-type*

(Optional) Type of notification (trap) to enable or disable. If no type is specified, all notifications available on the device are enabled or disabled.

The notification type can be one or more of the following keywords:

bfd

Enables Bidirectional Forwarding Detection (BFD) traps.

bgp

Enables BGP4-MIB and CISCO-BGP4-MIB traps.

bridgemib

Enables SNMP traps for the Bridge MIB.

config

Controls configuration notifications, as defined in the CISCO-CONFIG-MAN-MIB (enterprise 1.3.6.1.4.1.9.9.43.2). The notification type is: (1) ciscoConfigManEvent.

copy-complete

Enables CISCO-CONFIG-COPY-MIB ccCopyCompletion traps.

ds1

Enables SNMP Cisco DS1 traps.

ds2

Enables SNMP Cisco DS2 traps.

entity

Controls Entity MIB modification notifications. This notification type is defined in the ENTITY-MIB (enterprise 1.3.6.1.2.1.47.2) as: (1) entConfigChange.

ethernet

Enables Ethernet link OAM and 802.1ag connectivity fault management traps.

flash insertion

Enables ciscoFlashDeviceInsertedNotif.

flash removal

Enables ciscoFlashDeviceRemovedNotif.

fru-ctrl

Enables SNMP entity field-replaceable unit (FRU) control traps.

hsrp

Enables SNMP HSRP traps.

ipsec tunnel start

Enables SNMP IPsec tunnel start traps.

ipsec tunnel stop

Enables SNMP IPsec tunnel stop traps.

isakmp

Enables ISAKMP traps.

l2vpn all

Enables all Layer 2 VPN traps.

l2vpn vc-down

Enables Layer 2 VPN VC down traps.

l2vpn vc-up

Enables Layer 2 VPN VC up traps.

mpls frr all

Enables all MPLS fast reroute MIB traps.

mpls frr protected

Enables MPLS fast reroute tunnel protected traps.

mpls ldp

Enables SNMP Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP) traps.

mpls traffic-eng

Enables SNMP MPLS traffic engineering traps.

msdp peer-state-change

Enables SNMP MSDP Peer state change traps.

ntp

Enables SNMP Cisco NTP traps.

otn

Enables SNMP Cisco optical transport network (OTN) traps.

pim

Enables SNMP PIM traps.

rf

Enables RF-MIB traps.

sensor

Enables SNMP entity sensor traps.

snmp

Enables SNMP traps.

sonet

Enables SONET traps.

syslog

Controls error message notifications (Cisco-syslog-MIB). Specify the level of messages to be sent with the **logging history** command.

system

Enables SNMP SYSTEMMIB-MIB traps.

vpls

Enables virtual private LAN service (VPLS) traps.

vrrp events

Enables Virtual Router Redundancy Protocol (VRRP) traps.

Note
To display the trap notifications supported on a platform, use the online help (**?**) function.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server traps** command to enable trap requests for the specified notification types. To configure the router to send SNMP notifications, specify at least one **snmp-server traps** command. When the command is entered with no keyword, all notification types are enabled. When a notification type keyword is specified, only the notification type related to that keyword is enabled. To enable multiple types of notifications, issue a separate **snmp-server traps** command for each notification type.

More information about individual MIBs can be found in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

Some SNMP trap notifications require additional Task IDs as indicated in the following table:

Notification Type	Task ID	Operations
bfd	bgp	read, write
	ospf	read, write
	isis	read, write
	mpls-te	read, write
	snmp	read, write
bgp	bgp	read, write
copy-complete	config-services	read, write
ipsec	crypto	read, write
isakmp	crypto	read, write
l2vpn	l2vpn	read, write
mpls frr	mpls-ldp	read, write
	mpls-te	read, write
mpls l3vpn	ipv4	read, write
	mpls-ldp	read, write
	mpls-te	read, write
mpls ldp	mpls-ldp	read, write
	mpls-te	read, write
mpls traffic-eng	mpls-ldp	read, write
	mpls-te	read, write
ospf	ospf	read, write
syslog	sysmgr	read, write
vpls	l2vpn	read, write

This example shows how to enable the router to send all traps to the host specified by the name myhost.cisco.com, using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com public
```

snmp-server traps bgp

To enable Border Gateway Protocol (BGP) state-change Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps bgp** command in global configuration mode. To disable BGP state-change SNMP notifications, use the **no** form of this command.

snmp-server traps bgp
no snmp-server traps bgp

Syntax Description

This command has no keywords or arguments.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

Use the **snmp-server traps bgp** command to enable or disable BGP server state-change notifications, as defined in the BGP4-MIB (enterprise 1.3.6.1.2.1.15.7). The notifications types are:

- bgpEstablished
- bgpBackwardTransition

The BGP notifications are defined in the BGP-4 MIB as follows:

```
bgpTraps                OBJECT IDENTIFIER ::= { bgp 7 }

bgpEstablished NOTIFICATION-TYPE
OBJECTS { bgpPeerLastError,
bgpPeerState      }
STATUS  current
DESCRIPTION
"The BGP Established event is generated when the BGP FSM enters the ESTABLISHED
state."
::= { bgpTraps 1 }

bgpBackwardTransition NOTIFICATION-TYPE
OBJECTS { bgpPeerLastError,
bgpPeerState      }
STATUS  current
DESCRIPTION
"The BGPBackwardTransition Event is generated when the BGP FSM moves from a higher
numbered state to a lower numbered state."
::= {bgpTraps 2}
```

For a complete description of these notifications and additional MIB functions, see the BGP4-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps bgp** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write
bgp	read, write

The following example shows how to enable the router to send BGP state-change notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps bgp
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps frequency synchronization

To enable frequency synchronization MIB traps, use the **snmp-server traps frequency synchronization** command in global configuration mode. To disable frequency synchronization MIB traps, use the **no** form of this command.

snmp-server traps frequency synchronization
no snmp-server traps frequency synchronization

Syntax Description

This command has no keywords or arguments.

Command Default

Frequency synchronization MIB traps are disabled.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
snmp	read, write

This example shows how to enable frequency synchronization MIB traps:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps frequency synchronization
```

snmp-server traps mpls l3vpn

To enable the sending of MPLS Layer 3 VPN Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps mpls l3vpn** command in global configuration mode. To disable MPLS Layer 3 VPN SNMP notifications, use the **no** form of this command.

snmp-server traps mpls l3vpn {**all** | **max-threshold-cleared** | **max-threshold-exceeded** | **max-threshold-reissue-notif-time** *seconds* | **mid-threshold-exceeded** | **vrf-down** | **vrf-up**}
no snmp-server traps mpls l3vpn

Syntax Description

all	Enables all MPLS Layer 3 VPN traps.
max-threshold-cleared	Enables maximum threshold cleared traps.
max-threshold-exceeded	Enables maximum threshold exceeded traps.
max-threshold-reissue-notif-time *seconds*	Specifies the time interval for reissuing a maximum threshold notification, in seconds.
mid-threshold-exceeded	Enables mid-threshold exceeded traps.
vrf-down	Enables VRF down traps.
vrf-up	Enables VRF up traps.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to enable the device to send MPLS Layer 3 VPN traps:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps mpls l3vpn all
```

snmp-server traps ospf errors

To enable Open Shortest Path First (OSPF) error Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps ospf errors** command in global configuration mode. To disable OSPF error SNMP notifications, use the **no** form of this command.

snmp-server traps ospf errors {**authentication-failure** | **bad-packet** | **config-error** | **virt-authentication-failure** | **virt-bad-packet** | **virt-config-error**}
no snmp-server traps ospf errors {**authentication-failure** | **bad-packet** | **config-error** | **virt-authentication-failure** | **virt-bad-packet** | **virt-config-error**}

Syntax Description

authentication-failure	Enables SNMP traps for authentication failure errors on physical interfaces.
bad-packet	Enables SNMP traps for bad packet errors on physical interfaces.
config-error	Enables SNMP traps for configuration errors on physical interfaces.
virt-authentication-failure	Enables SNMP traps for authentication failure errors on virtual interfaces.
virt-bad-packet	Enables SNMP traps for bad packet errors on virtual interfaces.
virt-config-error	Enables SNMP traps for configuration errors on virtual interfaces.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

For a complete description of OSPF error notifications and additional MIB functions, see the OSPF-TRAP-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps ospf errors** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the router to send OSPF error notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps ospf errors
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps ospf lsa

To enable Open Shortest Path First (OSPF) link-state advertisement Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps ospf lsa** command in global configuration mode. To disable OSPF link state SNMP notifications, use the **no** form of this command.

snmp-server traps ospf lsa {**lsa-maxage** | **lsa-originate**}
no snmp-server traps ospf lsa {**lsa-maxage** | **lsa-originate**}

Syntax Description

lsa-maxage	Enables SNMP traps for link-state advertisement maxage.
lsa-originate	Enables SNMP traps for new link-state advertisement origination.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

For a complete description of OSPF link-state advertisement notifications and additional MIB functions, see the OSPF-TRAP-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps ospf lsa** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the router to send OSPF link-state advertisement notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps ospf lsa lsa-maxage
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps ospf retransmit

To enable Open Shortest Path First (OSPF) retransmission Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps ospf retransmit** command in global configuration mode. To disable OSPF retransmission SNMP notifications, use the **no** form of this command.

snmp-server traps ospf retransmit {**packets** | **virt-packets**}
no snmp-server traps ospf retransmit {**packets** | **virt-packets**}

Syntax Description

packets	Enables SNMP traps for packet retransmissions on physical interfaces.
virt-packets	Enables SNMP traps for packet retransmissions on virtual interfaces.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

For a complete description of OSPF retransmission notifications and additional MIB functions, see the OSPF-TRAP-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps ospf retransmit** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the router to send OSPF retransmission notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
Router(config)# snmp-server traps ospf retransmit packets
Router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps ospf state-change

To enable Simple Network Management Protocol (SNMP) notifications for Open Shortest Path First (OSPF) neighbor state change, use the **snmp-server traps ospf state-change** command in global configuration mode. To disable OSPF state-change SNMP notifications, use the **no** form of this command.

snmp-server traps ospf state-change {**if-state-change** | **neighbor-state-change** | **virtif-state-change** | **virtneighbor-state-change**}
no snmp-server traps ospf state-change {**if-state-change** | **neighbor-state-change** | **virtif-state-change** | **virtneighbor-state-change**}

Syntax Description

if-state-change	Enables SNMP traps for OSPF non-virtual interface state chages.
neighbor-state-change	Enables SNMP traps for OSPF neighbor state changes
virtif-state-change	Enables SNMP traps for OSPF virtual interface state changes.
virtneighbor-state-change	Enables SNMP traps for OSPF virtual neighbor state changes.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

Use the **snmp-server traps ospf state-change** command to enable or disable OSPF server state-change notifications, as defined in the MIB. One notification type is ospfNbrStateChange.

For example, the OSPF ospfNbrStateChange notification is defined in the OSPF MIB as follows:

```
!     ospfNbrStateChange NOTIFICATION-TYPE
!         OBJECTS {
!                     ospfRouterId, -- The originator of the trap
!                     ospfNbrIpAddr,
!                     ospfNbrAddressLessIndex,
!                     ospfNbrRtrId,
!                     ospfNbrState  -- The new state
!                   }
!         STATUS             current
```

For a complete description of these notifications and additional MIB functions, see the OSPF-TRAP-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps ospf state-change** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to enable the router to send OSPF state-change notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps ospf state-change neighbor-state-change
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps ospfv3 errors

To enable Open Shortest Path First (OSPF) Version 3 error Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps ospfv3 errors** command in global configuration mode. To disable OSPFv3 error SNMP notifications, use the **no** form of this command.

snmp-server traps ospfv3 errors [**bad-packet** | **config-error** | **virt-bad-packet** | **virt-config-error**]
no snmp-server traps ospfv3 errors [**bad-packet** | **config-error** | **virt-bad-packet** | **virt-config-error**]

Syntax Description

bad-packet	Enables SNMP traps for bad packet errors on physical interfaces.
config-error	Enables SNMP traps for configuration errors on physical interfaces.
virt-bad-packet	Enables SNMP traps for bad packet errors on virtual interfaces.
virt-config-error	Enables SNMP traps for configuration errors on virtual interfaces.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

For a complete description of OSPFv3 error notifications and additional MIB functions, see the OSPFV3-MIB in the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

The **snmp-server traps ospfv3 errors** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the router to send OSPF error notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps ospfv3 errors
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps ospfv3 state-change

To enable Simple Network Management Protocol (SNMP) notifications for Open Shortest Path First (OSPF) Version 3 state changes, use the **snmp-server traps ospfv3 state-change** command in global configuration mode. To disable OSPFv3 state-change SNMP notifications, use the **no** form of this command.

snmp-server traps ospfv3 state-change [**if-state-change** | **neighbor-state-change** | **nssa-state-change** | **restart-helper-status-change** | **restart-status-change** | **restart-virtual-helper-status-change** | **virtif-state-change** | **virtneighbor-state-change**]
no snmp-server traps ospfv3 state-change [**if-state-change** | **neighbor-state-change** | **nssa-state-change** | **restart-helper-status-change** | **restart-status-change** | **restart-virtual-helper-status-change** | **virtif-state-change** | **virtneighbor-state-change**]

Syntax Description

if-state-change	Enables SNMP traps for OSPFv3 non-virtual interface state chages.
neighbor-state-change	Enables SNMP traps for OSPFv3 neighbor state changes
nssa-state-change	Enables SNMP traps for OSPFv3 not so stubby area (NSSA) status changes.
restart-helper-status-change	Enables SNMP traps for OSPFv3 restart helper status changes.
restart-status-change	Enables SNMP traps for OSPFv3 restart status changes.
restart-virtual-helper-status-change	Enables SNMP traps for OSPFv3 virtual helper restart status changes.
virtif-state-change	Enables SNMP traps for OSPFv3 virtual interface state changes.
virtneighbor-state-change	Enables SNMP traps for OSPFv3 virtual neighbor state changes.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

SNMP notifications can be sent as traps.

Use the **snmp-server traps ospfv3 state-change** command to enable or disable the various OSPFv3 server state-change notifications, as defined in the MIB.

The **snmp-server traps ospfv3 state-change** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to enable the router to send OSPFv3 NSSA state-change notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps ospfv3 state-change nssa-state-change
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps pim interface-state-change

To enable Protocol Independent Multicast (PIM) interface status notification, use the **snmp-server traps pim interface-state-change** command in global configuration mode. To disable this command so no notification is sent, use the **no** form of this command.

snmp-server traps pim interface-state-change
no snmp-server traps pim interface-state-change

Syntax Description

This command has no keywords or arguments.

Command Default

Simple Network Management Protocol (SNMP) notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Use the **snmp-server traps pim interface-state-change** command to send notifications when a PIM interface changes status from up to down. When the status is up, the notification signifies the restoration of a PIM interface. When the status is down, the notification signifies the loss of a PIM interface.

PIM notifications are defined in the CISCO-PIM-MIB.my and PIM-MIB.my files that can be accessed from the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to use the **snmp-server traps pim interface-state-change** command:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps pim interface-state-change
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps pim invalid-message-received

To enable notifications for monitoring invalid Protocol Independent Multicast (PIM) protocol operations, such as invalid register received and invalid join or prune received, use the **snmp-server traps pim invalid-message-received** command in global configuration mode. To disable this command so that no notification is sent, use the **no** form of this command.

snmp-server traps pim invalid-message-received
no snmp-server traps pim invalid-message-received

Syntax Description

This command has no keywords or arguments.

Command Default

Simple Network Management Protocol (SNMP) notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

A router can receive a join or prune message in which the RP specified in the packet is not the RP for the multicast group. Or a router can receive a register message from a multicast group in which it is not the RP.

PIM notifications are defined in the CISCO-PIM-MIB.my and PIM-MIB.my files that can be accessed from the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to use the **snmp-server traps pim invalid-message-received** command:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps pim invalid-message-received
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps pim neighbor-change

To enable Protocol Independent Multicast (PIM) neighbor status down notifications, use the **snmp-server traps pim neighbor-change** command in global configuration mode. To disable PIM neighbor down notifications, use the **no** form of this command.

snmp-server traps pim neighbor-change
no snmp-server traps pim neighbor-change

Syntax Description

This command has no keywords or arguments.

Command Default

PIM Simple Network Management Protocol (SNMP) notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server traps pim neighbor-change** command to send notifications when a PIM neighbor changes status from up to down on an interface. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

PIM notifications are defined in the CISCO-PIM-MIB.my and PIM-MIB.my files that can be accessed from the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the router to send PIM neighbor status down notifications to the host at the address myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps pim neighbor-change
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps pim rp-mapping-change

To enable notifications indicating a change in the rendezvous point (RP) mapping information due to either Auto-RP or bootstrap router (BSR) messages, use the **snmp-server traps pim rp-mapping-change** command in global configuration mode. To disable this command so no notification is sent, use the **no** form of this command.

snmp-server traps pim rp-mapping-change
no snmp-server traps pim rp-mapping-change

Syntax Description

This command has no keywords or arguments.

Command Default

PIM SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

PIM notifications are defined in the CISCO-PIM-MIB.my and PIM-MIB.my files that can be accessed from the SNMP Object Navigator, available through cisco.com at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to use the **snmp-server traps pim rp-mapping-change** command:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps pim rp-mapping-change
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server traps rsvp

To enable the sending of Resource Reservation Protocol (RSVP) notifications, use the **snmp-server traps rsvp** command in global configuration mode. To disable RSVP notifications, use the **no** form of this command.

snmp-server traps rsvp {**all** | **lost-flow** | **new-flow**}

Syntax Description

all	Enables the sending of both new flow lost flow traps.
lost-flow	Enables the sending of traps when a flow is deleted.
new-flow	Enables the sending of traps when a flow is created.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
mpls-te	read, write
ouni	read, write
snmp	read, write

This example illustrates how to enable all SNMP RSVP MIB traps.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# snmp-server traps rsvp all
```

snmp-server traps selective-vrf-download role-change

To attempt to download only those prefixes and labels to a physical entity required to forward traffic through the physical entity, use the **snmp-server trap selective-vrf-download role-change** command in global configuration mode.

snmp-server trap selective-vrf-download role-change

This command has no keywords or arguments.

Command Default

Selective VRF downloads are disabled.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The selective VRF download feature makes a best effort to download only those prefixes and labels to a physical entity required to forward traffic through the physical entity. This is accomplished by characterizing roles for physical entities based on their configuration.

From a network management point of view the CISCO-SELECTIVE-VRF-DOWNLOAD-MIB:

- Lists the state relating to the selective VRF download feature for each physical entity capable of forwarding packets.
- Lists the role change history per address family (ipv4 and ipv6) for each physical entity capable of forwarding packets.
- Lists the VRF tables selectively downloaded to each physical entity capable of forwarding packets.

Task ID

Task ID	Operation
snmp	read, write
basic-services	read, write

This example shows how to enable the selective VRF downloads:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps selective-vrf-download role-change
```

snmp-server traps snmp

To enable the sending of RFC 1157 Simple Network Management Protocol (SNMP) notifications, use the **snmp-server traps snmp** command in the appropriate configuration mode. To disable RFC 1157 SNMP notifications, use the **no** form of this command.

snmp-server traps snmp [**authentication** | **coldstart** | **linkdown** | **linkup** | **warmstart**]
no snmp-server traps snmp [**authentication** | **coldstart** | **linkdown** | **linkup** | **warmstart**]

Syntax Description

authentication	(Optional) Controls the sending of SNMP authentication failure notifications.
linkup	(Optional) Controls the sending of SNMP linkUp notifications
linkdown	(Optional) Controls the sending of SNMP linkDown notifications
coldstart	(Optional) Controls the sending of SNMP coldStart notifications.
warmstart	(Optional) Controls the sending of SNMP warmStart notifications.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **snmp-server traps snmp** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

The optional **authentication** keyword controls the sending of SNMP authentication failure notifications. In order to send notifications, you must configure at least one **snmp-server host** command. An authentication Failure (4) trap signifies that the sending device is the addressee of a protocol message that is not properly authenticated. The authentication method depends on the version of SNMP being used. For SNMPv1 or SNMPv2, authentication failure occurs for packets with an incorrect community string. For SNMPv3, authentication failure occurs for packets with an incorrect Secure Hash Algorithm (SHA) or Message Digest 5 (MD5) authentication key or for a packet that is outside the authoritative SNMP engine's window, for example, the packets that are configured outside access lists or time ranges. In such an instance, only a report Protocol Data Unit (PDU) is generated, and authentication failure traps are not generated.

The optional **linkup** keyword controls the sending of SNMP linkUp notifications. The linkUp(3) trap signifies that the sending device recognizes one of the communication links represented in the agent's configuration coming up.

The optional **linkdown** keyword controls the sending of SNMP linkDown notifications. The linkDown(2) trap signifies that the sending device recognizes a failure in one of the communication links represented in the agent's configuration.

The **snmp-server traps snmp** command with the **linkup** or **linkdown** keywords globally enables or disables SNMP linkUp and linkDown traps. After enabling either of these traps globally, you can enable or disable these traps on specific interfaces using the **no notification linkupdown disable** command in interface configuration mode. According to RFC 2863, linkUp and linkDown traps are enabled for interfaces that do not operate on top of any other interface (as defined in the ifStackTable), and are disabled otherwise. This means that you do not have to enable linkUp and linkdown notifications on such interfaces. However, linkUp and linkDown notifications will not be sent unless you enable them globally using the **snmp-server traps snmp** command.

The optional **coldstart** keyword controls the sending of SNMP coldStart notifications. The coldStart(0) trap signifies that the sending device is reinitializing itself such that the agent's configuration or the protocol entity implementation may be altered.

The optional **warmstart** keyword controls the sending of SNMP coldStart notifications. The warmStart(1) trap signifies that the sending device is reinitializing itself such that neither the agent configuration nor the protocol entity implementation is altered.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to enable the device to send all traps to the host myhost.cisco.com using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps snmp
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com public snmp
```

The following example shows how to enable only linkUp and linkDown traps:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps snmp linkup
RP/0/RP0/CPU0:router(config)# snmp-server traps snmp linkdown
```

snmp-server traps subscriber session-aggregation

To set the session aggregation parameters, use the **snmp-server traps subscriber session** command in global configuration mode. To delete the set parameters, use the no form of the command.

snmp-server traps subscriber session-agregation [**access-interface** | **node**]

no snmp-server traps subscriber session-agregation [**access-interface** | **node**]

Syntax Description

access-interface	Subscriber notification at access interface level.
node	Subscriber notification at node level.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **snmp-server traps subscriber session-agregation** command to enable CISCO-SUBSCRIBER-SESSION-MIB notifications (traps). Notifications will include MIB's asynchronous events.

Task ID

Task ID	Operation
snmp	read, write

Example

```
RP/0/RP0/CPU0:router (config)# snmp-server traps subscriber session-aggregation node
```

snmp-server traps syslog

To enable Simple Network Management Protocol (SNMP) notifications of Cisco-syslog-MIB error messages, use the **snmp-server traps syslog** command in the appropriate configuration mode. To disable these types of notifications, use the **no** form of this command.

snmp-server traps syslog
no snmp-server traps syslog

Syntax Description

This command has no keywords or arguments.

Command Default

SNMP notifications are disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **snmp-server traps syslog** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP notifications.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to enable Cisco-syslog-MIB error message notifications to the host at the address myhost.cisco.com, using the community string defined as public:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps syslog
RP/0/RP0/CPU0:router(config)# snmp-server host myhost.cisco.com version 2c public
```

snmp-server user

To configure a new user to a Simple Network Management Protocol (SNMP) group, use the **snmp-server user** command in

global configuration

mode. To remove a user from an SNMP group, use the **no** form of this command.

snmp-server user *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv** {**3des** | **aes** *aes-bit-encryption* | **des56**} {**clear** | **encrypted**} *priv-password*]]} [**SDROwner** | **SystemOwner**] [*access-list-name*]
no snmp-server user *username groupname*

Syntax Description

username	Name of the user on the host that connects to the agent. **Note** The recommended range for a user-defined username is 2-253 characters.
groupname	Name of the group to which the user belongs.
v1	Specifies that the SNMPv1 security model should be used.
v2c	Specifies that the SNMPv2c security model should be used.
v3	Specifies that the SNMPv3 security model should be used.
auth	(Optional) Specifies which authentication level should be used. If this keyword is used, you must specify an authentication level and an authorization password.
md5	Specifies the HMAC-MD5-96 authentication level.
sha	Specifies the HMAC-SHA-96 authentication level.
clear	Specifies that an unencrypted password follows.
encrypted	Specifies that an encrypted password follows.
auth-password	Authentication password, which is a string (not to exceed 64 characters) that enables the agent to receive packets from the host.
priv	(Optional) Specifies that encryption parameters follow.
3des	Specifies the 168-bit Triple Data Encryption Standard (3DES) level of encryption for the user.
aes *aes-bit-encryption*	Specifies the Advanced Encryption Standard (AES) level of encryption for the user. Supported options are 128, 192 and 256 bit encryption.

des56	Specifies the 56-bit Data Encryption Standard (DES) level of encryption for the user.
priv-password	Privacy password, which can be clear or encrypted text, according to what is specified.
SDROwner	(Optional) Limits access to the agents for the owner secure domain router (SDR) only.
SystemOwner	(Optional) Provides system-wide access to the agents for all SDRs.
access-list-name	(Optional) Access list to be associated with this SNMP user. The *access-list-name* argument represents a value from 1 to 99, that is, the identifier of the standard IP access list.

Command Default

By default, access is limited to agents on the owner SDR only.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To use 3DES and AES encryption standards, you must have installed the security package (k9sec). For information on installing software packages, see *Upgrading and Managing Cisco IOS XR Software* in *System Management Configuration Guide for Cisco 8000 Series Routers*.

Note Only one remote host can be assigned to the same username for SNMP version 3. If you configure the same username with different remote hosts, only the last username and remote host combination will be accepted and will be seen in the **show running** configuration. In the case of multiple SNMP managers, multiple unique usernames are required.

Table 37: snmp-server user Default Descriptions

Characteristic	Default
passwords	Text strings are assumed.
access lists	Access from all IP access lists is permitted.

SDR and System-wide Access

When the **snmp-server user** command is entered with the **SDROwner** keyword, SNMP access is granted only to the MIB object instances in the owner SDR.

When the **snmp-server user** command is entered with the **SystemOwner** keyword, SNMP access is granted to the entire system.

Task ID

Task ID	Operations
snmp	read, write

The following example shows how to enter a plain-text password for the string *abcd* for user2 in group2:

```
RP/0/RP0/CPU0:router(config)# snmp-server user user2 group2 v3 auth md5 clear abcd
```

To learn if this user has been added to the configuration, use the **show snmp user** command.

If the localized Message Digest 5 (MD5) or Secure Hash Algorithm (SHA) digest is known, specify that string instead of the plain-text password. The digest should be formatted as AA:BB:CC:DD where AA, BB, CC, and DD are hexadecimal values. The digest should also be exactly 16 octets long.

This example shows how to specify the command with a digest name of 00:11:22:33:44:55:66:77:88:99:AA:BB:CC:DD:EE:FF:

```
RP/0/RP0/CPU0:router(config)# snmp-server user user2 group2 v3 auth md5 encrypted
     00:11:22:33:44:55:66:77:88:99:AA:BB:CC:DD:EE:FF
```

snmp-server view

To create or update a Simple Network Management Protocol (SNMP) view entry, use the **snmp-server view** command in

global configuration

mode. To remove the specified server view entry, use the **no** form of this command.

snmp-server view *view-name oid-tree* {**excluded** | **included**}
no snmp-server view *view-name oid-tree* {**excluded** | **included**}

Syntax Description

view-name	Label for the view record being updated or created. The name is used to reference the record.
oid-tree	Object identifier (OID) of the ASN.1 subtree to be included or excluded from the view. To identify the subtree, specify a text string consisting of numbers, such as 1.3.6.2.4, or a word, such as *system*. Replace a single subidentifier with the asterisk (*) wildcard to specify a subtree family; for example 1.3.*.4.
excluded	Excludes the MIB family from the view.
included	Includes the MIB family in the view.

Command Default

No view entry exists.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Other SNMP commands require a view as a keyword. Use the **snmp-server view** command to create a view to be used as keywords for other commands that create records including a view.

Instead of defining a view explicitly, you can rely on the following predefined views, which are supported by the SNMP agent:

all

Predefined view indicating that a user can see all objects.

CfgProt

Predefined view indicating that a user can see all objects except the SNMPv3 configuration tables.

vacmViewTreeFamilyEntry

Predefined view indicating that a user can see the default configuration of vacmViewTreeFamilyEntry.

The predefined views supported on Cisco IOS XR software, however, do not match the predefined views specified in RFC 3415.

Task ID

Task ID	Operations
snmp	read, write

This example creates a view that includes all objects in the MIB-II subtree:

```
RP/0/RP0/CPU0:router(config)# snmp-server view mib2 1.3.6.1.2.1 included
```

This example shows how to create a view that includes all objects in the MIB-II system group and all objects in the Cisco enterprise MIB:

```
RP/0/RP0/CPU0:router(config)# snmp-server view view1 1.3.6.1.2.1.1 included
RP/0/RP0/CPU0:router(config)# snmp-server view view1 1.3.6.1.4.1.9 included
```

This example shows how to create a view that includes all objects in the MIB-II system group except for sysServices (System 7) and all objects for interface 1 in the MIB-II interfaces group:

```
RP/0/RP0/CPU0:router(config)# snmp-server view view1 1.3.6.1.2.1.1 included
RP/0/RP0/CPU0:router(config)# snmp-server view view1 1.3.6.1.2.1.1.7 excluded
RP/0/RP0/CPU0:router(config)# snmp-server view view1 1.3.6.1.2.1.2.2.1.*.1 included
```

snmp-server vrf

To configure the VPN routing and forwarding (VRF) properties of Simple Network Management Protocol (SNMP), use the **snmp-server vrf** command in

global configuration

mode. To remove the configuration, use the **no** form of this command.

snmp-server vrf *vrf-name*[**host** *address* [**clear** | **encrypted**][**traps**][**version** {**1** | **2c** | **3** *security-level*}] *community-string*[**udp-port** *port*]][**context** *context-name*]
no snmp-server vrf *vrf-name*

Syntax Description

vrf-name	Name of the VRF.
host *address*	(Optional) Specifies the name or IP address of the host (the targeted recipient).
clear	(Optional) Specifies that the *community-string* argument is clear text.
encrypted	(Optional) Specifies that the *community-string* argument is encrypted text.
traps	(Optional) Specifies that notifications should be sent as traps. This is the default.
version {**1** | **2c** | **3**}	(Optional) Specifies the version of the SNMP used to send the traps. The default is SNMPv1. When the **version** keyword is used, one of these keywords must be specified: • **1**—SNMPv1 • **2c**—SNMPv2C • **3**—SNMPv3
security-level	(Optional) Security level for SNMPv3. Options are: • **auth**—authNoPriv • **noauth**—noAuthNoPriv • **priv**—authPriv
community-string	Specifies the community string for SNMPv1 and SNMPv2, or the SNMPv3 user.
udp-port *port*	(Optional) Specifies the UDP port to which notifications should be sent.
context *context-name*	(Optional) Name of the context that must be mapped to VRF identified by value of the *vrf-name* argument.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use this command to enter SNMP VRF configuration mode and configure an SNMP notification recipient on a VRF. You can also map a VRF to an SNMP context.

SNMP notification recipient that is reachable by way of a VRF can be configured. Notification is forwarded to the recipient represented by its address using the routing table instance identified by the VRF name.

The *address* argument can be either a host name or an IP address.

Use the **clear** keyword to specify that the clear text community string you enter is displayed encrypted in the **show running** command output. To enter an encrypted string, use the **encrypted** keyword. To enter a clear text community string that is not encrypted by the system, use neither of these keywords.

An SNMP context identified by the value of the *context-name* argument can be mapped to a VRF in this mode. This context must be created using **snmp-server context** command.

Task ID

Task ID	Operations
snmp	read, write

This example shows how to configure a host IP address for a VRF name:

```
RP/0/RP0/CPU0:router(config)# snmp-server vrf vrfa
RP/0/RP0/CPU0:router(config-snmp-vrf)# host 12.21.0.1 traps version
2c public udp-port 2525
```



Software Package Management Commands

- install commit, on page 436
- show install active, on page 437
- show install available, on page 440
- show install cached, on page 441
- show install committed, on page 443
- show install fixes, on page 445
- show install history , on page 447
- show install log, on page 449
- show install request, on page 450
- show install rollback, on page 452

install commit

To make the active software persistent across system reloads, use the **install commit** command in XR EXEC mode.

install commit [**synchronous**]

synchronous	(Optional) Applies the changes synchronously.

Command Default

Commits the active software set.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To make the current active software persistent across reloads, use the **install commit** command.

If the system is restarted before the active software set is saved with the **install commit** command, the previously committed software set is used.

Task ID

Task ID	Operations
pkg-mgmt	read, write

The following example shows how to make the current active software set persistent:

```
Router# install commit

Install operation 16 'install commit' started by user 'user_b' at 19:18:58 UTC
Sat Apr 08 2006.
Install operation 16 completed successfully at 19:19:01 UTC Sat Apr 08 2006.
```

show install active

To display active packages, use the **show install active** command in XR EXEC mode.

show install active [**all** [**location** { *node-id* | **all** }] | **keys** | **node-inconsistencies** | **origin** | **summary**]

Syntax Description

all location	Displays all the active packages.
location *node-id* **location all**	Displays all packages in a specified location or all locations.
keys	Display all keys used to sign each package.
node-inconsistencies	Displays all active non-XR packages inconsistent across all nodes.
origin	Displays all packages from a specific origin.
summary	Displays a summary of active packages along with a list of optional packages.

Command Default

The command without the **all** keyword just displays the currently active XR packages. It does not display all packages on the system such as OS packages.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 26.1.1	A new keyword, **keys**, was added to the command to show all active packages, and the key ID of the keys the packages are signed with.
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

This example shows how to view all active packages:

```
Router#show install active all
Wed Mar  8 04:57:35.230 UTC

Software Hash: bf46f5add2ea3cd85c91c647a2ca043e1a829c49e5c290baec0c89c937a17429
Package                                                          Version
-------------------------------------------------------          ------------------
8000-boot-scripts                                                7.8.1.34Iv1.0.0-r0
8000-cpa-setup-x86                                               7.8.1.34Iv1.0.0-r0
8101-32h-cpa-sb-x86                                              7.8.1.34Iv1.0.0-r0
8102-64h-cpa-sb-x86                                              7.8.1.34Iv1.0.0-r0
8111-32eh-cpa-sb-x86                                             7.8.1.34Iv1.0.0-r0
8201-32fh-cpa-sb-x86                                             7.8.1.34Iv1.0.0-r0
8201-cpa-sb-x86                                                  7.8.1.34Iv1.0.0-r0
8202-32fh-m-cpa-sb-x86                                           7.8.1.34Iv1.0.0-r0
8202-cpa-sb-x86                                                  7.8.1.34Iv1.0.0-r0
8203-88h16fh-m-cpa-sb-x86                                        7.8.1.34Iv1.0.0-r0
```

```
8212-cpa-sb-x86                                                 7.8.1.34Iv1.0.0-r0
8608-fb-data-cpa-sb-x86                                         7.8.1.34Iv1.0.0-r0
8608-mpa-crevalle-cpa-sb-x86                                    7.8.1.34Iv1.0.0-r0
8608-mpa-narwhal-cpa-sb-x86                                     7.8.1.34Iv1.0.0-r0
8608-mpa-pinafore-cpa-sb-x86                                    7.8.1.34Iv1.0.0-r0
8608-mpa-tiber-cpa-sb-x86                                       7.8.1.34Iv1.0.0-r0
8804-fc-data-cpa-sb-x86                                         7.8.1.34Iv1.0.0-r0
acl                                                             2.2.52-r0.61
argon                                                           7.8.1.34Iv1.0.0-r0
 --More--
```

This example shows how to view all active packages in a specified location:

```
Router#show install active all location 0/RP0/CPU0
Wed Mar  8 04:58:25.254 UTC

Software Hash: ef055dac99b856182211deb7585c02fb812c02d6a04cd5b92b521ccf7b9c6ee5
Package                                     Architecture                    Version
------------------------------------------- ------------------  -------------------
8000-boot-scripts                           corei7_64            7.8.1.34Iv1.0.0-r0
8000-cpa-setup-x86                          corei7_64            7.8.1.34Iv1.0.0-r0
8101-32h-cpa-sb-x86                         corei7_64            7.8.1.34Iv1.0.0-r0
8102-64h-cpa-sb-x86                         corei7_64            7.8.1.34Iv1.0.0-r0
8111-32eh-cpa-sb-x86                        corei7_64            7.8.1.34Iv1.0.0-r0
8201-32fh-cpa-sb-x86                        corei7_64            7.8.1.34Iv1.0.0-r0
8201-cpa-sb-x86                             corei7_64            7.8.1.34Iv1.0.0-r0
8202-32fh-m-cpa-sb-x86                      corei7_64            7.8.1.34Iv1.0.0-r0
8202-cpa-sb-x86                             corei7_64            7.8.1.34Iv1.0.0-r0
8203-88h16fh-m-cpa-sb-x86                   corei7_64            7.8.1.34Iv1.0.0-r0
8212-cpa-sb-x86                             corei7_64            7.8.1.34Iv1.0.0-r0
8608-fb-data-cpa-sb-x86                     corei7_64            7.8.1.34Iv1.0.0-r0
8608-mpa-crevalle-cpa-sb-x86                corei7_64            7.8.1.34Iv1.0.0-r0
8608-mpa-narwhal-cpa-sb-x86                 corei7_64            7.8.1.34Iv1.0.0-r0
8608-mpa-pinafore-cpa-sb-x86                corei7_64            7.8.1.34Iv1.0.0-r0
8608-mpa-tiber-cpa-sb-x86                   corei7_64            7.8.1.34Iv1.0.0-r0
8804-fc-data-cpa-sb-x86                     corei7_64            7.8.1.34Iv1.0.0-r0
acl                                         corei7_64            2.2.52-r0.61
argon                                       wrlsdk_x86           7.8.1.34Iv1.0.0-r0
 --More--
```

This example shows how to view all active keys used to sign packages:

```
Router# show install active keys
Tue Feb 10 05:02:59.398 UTC

XR Software Hash: 78a02a7bd0157fae32a04cf764b8679d0ddd365541ae9a5799a60bb526a924ff

XR Package                                        Version                GPG Key ID
------------------------------------------------  ---------------------  ----------------
xr-8000-af-ea                                    26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-aib                                      26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-bfd                                      26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-buffhdr-ea                               26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-bundles                                  26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-card-support                             26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-cdp-ea                                   26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-cem-driver                               26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-cfm                                      26.1.1.30Iv1.0.0-1     747278115c06976b
xr-8000-common-otn                               26.1.1.30Iv1.0.0-1     747278115c06976b
--More--
```

This example shows how to view the summary of active pakages:

```
Router#show install active summary
Wed Mar  8 05:00:29.564 UTC
```

```
Active Packages:    XR: 205    All: 1465
Label:              7.8.1.34I-PROD_BUILD_7_8_1_34I_SIT_IMAGE
Software Hash:      bf46f5add2ea3cd85c91c647a2ca043e1a829c49e5c290baec0c89c937a17429

Optional Packages                                                    Version
---------------------------------------------------- -          ------------------
xr-8000-l2mcast                                                  7.8.1.34Iv1.0.0-1
xr-8000-mcast                                                    7.8.1.34Iv1.0.0-1
xr-8000-netflow                                                  7.8.1.34Iv1.0.0-1
xr-bgp                                                           7.8.1.34Iv1.0.0-1
xr-cdp                                                           7.8.1.34Iv1.0.0-1
xr-ipsla                                                         7.8.1.34Iv1.0.0-1
xr-is-is                                                         7.8.1.34Iv1.0.0-1
xr-k9sec                                                         7.8.1.34Iv1.0.0-1
xr-lldp                                                          7.8.1.34Iv1.0.0-1
xr-mcast                                                         7.8.1.34Iv1.0.0-1
xr-mpls-oam                                                      7.8.1.34Iv1.0.0-1
xr-netflow                                                       7.8.1.34Iv1.0.0-1
xr-ospf                                                          7.8.1.34Iv1.0.0-1
xr-perf-meas                                                     7.8.1.34Iv1.0.0-1
xr-perfmgmt                                                      7.8.1.34Iv1.0.0-1
xr-telnet                                                        7.8.1.34Iv1.0.0-1
xr-track                                                         7.8.1.34Iv1.0.0-1
 --More--
```

show install available

To display packages available for installation, use the **show install available** command in XR EXEC mode.

show install available [**all**]

Syntax Description

all location *node-id*	(Optional) Displays all packages available to be installed in a system.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to view the available packages:

```
Router# show install available all
```

show install cached

To display cached packages, use the **show install cached** command in XR EXEC mode.

show install cached

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to view the cached packages:

```
Router# show install cached
Wed Aug 12 10:00:25.105 UTC

Package                                                Architecture
Version
------------------------------------------------------ ----------------
--------------------------
8000-cpa-setup-nos-x86                                 corei7_64
7.0.12v1.0.0-r0
8000-cpa-setup-x86                                     corei7_64
7.0.12v1.0.0-r0
8201-cpa-sb-x86                                        corei7_64
7.0.12v1.0.0-r0
8800-lc-36fh-cpa-sb-x86                                corei7_64
7.0.12v1.0.0-r0
8800-lc-48h-cpa-sb-x86                                 corei7_64
7.0.12v1.0.0-r0
8800-rp-cpa-sb-x86                                     corei7_64
7.0.12v1.0.0-r0
8808-data-cpa-sb-x86                                   corei7_64
7.0.12v1.0.0-r0
8812-data-cpa-sb-x86                                   corei7_64
7.0.12v1.0.0-r0
8818-data-cpa-sb-x86                                   corei7_64
7.0.12v1.0.0-r0
acl                                                    corei7_64
2.2.52-r0.9
at                                                     corei7_64
3.1.20-r0.9
attr                                                   corei7_64
2.4.47-r0.9
audit                                                  corei7_64
2.7-r0.10
auditd                                                 corei7_64
```

```
2.7-r0.10
base-files                                      wrlsdk_x86
3.0.14-r89.7
base-passwd                                     corei7_64
3.5.29-r0.1.9
base-passwd-update                              corei7_64
3.5.29-r0.1.9
bash                                            corei7_64
4.3.30-r0.9
bash-completion                                 corei7_64
2.4-r0.9
bash-completion-extra                           corei7_64
2.4-r0.9
```

show install committed

To display committed packages, use the **show install committed** command in XR EXEC mode.

show install committed [**all** [**location** { *node-id* | **all** }] | **keys** | **node-inconsistencies** | **origin** | **summary**]

Syntax Description

all	Displays user-installable XR packages on all locations.
location *node-id* **location all**	Displays all packages in a specified location or all locations.
keys	Display all keys used to sign each package.
node-inconsistencies	Displays all committed non-XR packages inconsistent across all nodes.
origin	Displays all packages from a specific origin.
summary	Displays summary of committed packages (optional and bugfix packages only). .

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 26.1.1	A new keyword, **keys**, was added to the command to show all keys used to sign packages.
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

This example shows how to view all active keys used to sign packages:

```
Router# show install committed keys
Tue Feb 10 05:06:12.091 UTC

XR Software Hash: 78a02a7bd0157fae32a04cf764b8679d0ddd365541ae9a5799a60bb526a924ff

XR Package                                    Version                      GPG Key ID
--------------------------------------------- ---------------------------- ----------------
xr-8000-af-ea                                 26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-aib                                   26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-bfd                                   26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-bsp                                   26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-buffhdr-ea                            26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-bundles                               26.1.1.30Iv1.0.0-1           747278115c06976b
xr-8000-card-support                          26.1.1.30Iv1.0.0-1           747278115c06976b
```

```
xr-8000-cdp-ea                                26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cem-driver                            26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cfm                                   26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-common-otn                            26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-core                                  26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa                                   26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-devobj-gnss                       26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-devobj-misc                       26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-driver-fpgalib-kmod-oe            26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-npu                               26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-npu-sdk                           26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-cpa-sb-data                           26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-dot1x                                 26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-dsm                                   26.1.1.30Iv1.0.0-1          747278115c06976b
xr-8000-dyinggasp-pd                          26.1.1.30Iv1.0.0-1          747278115c06976b
--More--
```

The following example shows how to view the summary of committed packages:

```
Router#show install committed summary
Wed Jun 28 10:34:18.792 UTC
Committed Packages: XR: 206    All: 1537
Label:              7.10.1.31I
Software Hash:      8ba19933d88a72378955c470a1e07b0742ddded043ec332a72c5640c1ec62c55

Optional Packages                                                    Version
-------------------------------------------                  ------------------
xr-8000-l2mcast                                              7.10.1.31Iv1.0.0-1
xr-8000-mcast                                                7.10.1.31Iv1.0.0-1
xr-8000-netflow                                              7.10.1.31Iv1.0.0-1
xr-bgp                                                       7.10.1.31Iv1.0.0-1
xr-ipsla                                                     7.10.1.31Iv1.0.0-1
xr-is-is                                                     7.10.1.31Iv1.0.0-1
xr-lldp                                                      7.10.1.31Iv1.0.0-1
xr-mcast                                                     7.10.1.31Iv1.0.0-1
xr-mpls-oam                                                  7.10.1.31Iv1.0.0-1
xr-netflow                                                   7.10.1.31Iv1.0.0-1
xr-ops-script-repo                                           7.10.1.31Iv1.0.0-1
xr-ospf                                                      7.10.1.31Iv1.0.0-1
xr-perf-meas                                                 7.10.1.31Iv1.0.0-1
xr-perfmgmt                                                  7.10.1.31Iv1.0.0-1
xr-track                                                     7.10.1.31Iv1.0.0-1
```

show install fixes

To display bug fixes, use the **show install fixes** command in XR EXEC mode.

show install fixes { **active** | **available** | **committed** | **deactivate** } [**brief** | **bugids** *value*]

active	Display active bug fixes.
available	Display the list of available bug fixes.
committed	Display committed bug fixes
deactivate *bugfix_list*	Display information on how to remove a bug fix
brief	(Optional) Displays only the bug IDs and not the relevant packages.
bugids *value*	(Optional) Displays bug fixes for a specific bug ID. You can choose to view information for more than one bug id seperated by a space.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to view the list of available bug fixes:

```
Router# show install fixes available
Bug Id             Packages                       Repository
-----------    ----------------------------  ----------------
CSCxx12345     xr-8000-core-7.0.11v1.0.1-1       <repository-name>
               xr-core-7.0.11v1.0.1-1            <repository-name>
```

The following example shows how to view the information for a active bug ID:

```
Router# show install fixes active bugids CSCxx12345
```

The following example shows how to remove the DDTS CSCwc26944:

```
Router#show install fixes deactivate CSCwc26944

User-requested DDTSs deactivated by this command: CSCwc26944

All DDTSs deactivated by this command: CSCvs01738,CSCwc26944

Package changes:
```

```
  xr-8000-core-7.5.2v1.0.5 -> xr-8000-core-7.5.2v1.0.4
  xr-8000-fib-ea-7.5.2v1.0.1 -> xr-8000-fib-ea-7.5.2v1.0.0           (missing)
  xr-8000-leabaofa-7.5.2v1.0.3 -> xr-8000-leabaofa-7.5.2v1.0.2
  xr-8000-mcast-7.5.2v1.0.1 -> xr-8000-mcast-7.5.2v1.0.0             (missing)
  xr-8000-utapp-blaze-7.5.2v1.0.2 -> xr-8000-utapp-blaze-7.5.2v1.0.1
  xr-fib-7.5.2v1.0.3 -> xr-fib-7.5.2v1.0.2
  xr-mcast-7.5.2v1.0.1 -> xr-mcast-7.5.2v1.0.0                       (missing)
  xr-ncs540l-core-7.5.2v1.0.14 -> xr-ncs540l-core-7.5.2v1.0.10
  xr-ncs5700-core-7.5.2v1.0.14 -> xr-ncs5700-core-7.5.2v1.0.10
  xr-ofa-7.5.2v1.0.3 -> xr-ofa-7.5.2v1.0.1
  xr-snmp-7.5.2v1.0.1 -> xr-snmp-7.5.2v1.0.0                         (missing)

Example install commands:
  install source any-configured xr-8000-core-7.5.2v1.0.4 xr-8000-fib-ea-7.5.2v1.0.0
xr-8000-leabaofa-7.5.2v1.0.2 xr-8000-mcast-7.5.2v1.0.0 xr-8000-utapp-blaze-7.5.2v1.0.1
xr-fib-7.5.2v1.0.2 xr-mcast-7.5.2v1.0.0 xr-ncs540l-core-7.5.2v1.0.10
xr-ncs5700-core-7.5.2v1.0.10 xr-ofa-7.5.2v1.0.1 xr-snmp-7.5.2v1.0.0
  install package downgrade xr-8000-core-7.5.2v1.0.4 xr-8000-fib-ea-7.5.2v1.0.0
xr-8000-leabaofa-7.5.2v1.0.2 xr-8000-mcast-7.5.2v1.0.0 xr-8000-utapp-blaze-7.5.2v1.0.1
xr-fib-7.5.2v1.0.2 xr-mcast-7.5.2v1.0.0 xr-ncs540l-core-7.5.2v1.0.10
xr-ncs5700-core-7.5.2v1.0.10 xr-ofa-7.5.2v1.0.1 xr-snmp-7.5.2v1.0.0

IMPORTANT: The above commands cannot currently be run because there are missing packages.Put
 the following packages in an accessible repository.
  xr-8000-fib-ea-7.5.2v1.0.0
  xr-8000-mcast-7.5.2v1.0.0   (optional package)
  xr-mcast-7.5.2v1.0.0        (optional package)
  xr-snmp-7.5.2v1.0.0

IMPORTANT: If the optional packages are not available, then they can be completely removed
 before removing the DDTS using install package remove xr-8000-mcast-7.5.2v1.0.0
xr-mcast-7.5.2v1.0.0
```

show install history

To display the history of the installation process , use the **show install history** command in XR EXEC mode.

show install history { **all** [**location** *node-id*] [**reverse**] [**verbose**] | **table** [**ID** *number* | [**last** *entries*] [**reverse**]] }

all	Displays the history of all install operations.
last *entries*	Displays a last entry of the history information table
location *node-id*	Displays details of the specified location. The *node-id* argument is entered in the `rack/slot/module` notation.
table	Displays a summary table of the install operations.
ID *number*	Specifies the table ID of the history information.
reverse	Displays entries in reverse order.
verbose	Displays detailed information.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To identify whether a reload is required or only process restart is needed, use **show install history last transaction verbose** command.

When there is insufficient disk space in the root file system, the **show install request** command displays an error message. Use **show install history last package** command to view the details of the packaging operation and any errors that have occurred.

The following example shows how to view the history of the install operation:

```
Router# show install history table
  Transaction             Atomic Change          Packaging Operations
--------------------  -------------------------  ----------------------------------
Id   Status          Id  Method   Status      Id  Operation   Inputs Status
--------------------  -------------------------  ----------------------------------
 1   In progress     1   Reload   Success      1   Upgrade        1  Success
```

The following example shows how to view history of last transaction:

```
Router# show install history last transaction verbose
2019-09-11 17:01:46 UTC    Transaction 3 started
2019-09-11 17:01:46 UTC      Atomic change 3.1 started
2019-09-11 17:01:46 UTC      Packaging operation 3.1.1 startedtion cleanup in progress
2019-09-11 17:16:46 UTC    Transaction 3 complete
```

The following example shows how to view the last packaging operation for a specific location:

```
Router# show install history last package location 0/RP0/CPU0 verbose
```

The following example shows how to view the complete details of the last packaging operation, including a failure:

```
Router# show install history last package
Wed Jan  8 13:29:51.586 UTC
2020-01-08 13:25:32 UTC        Packaging operation 3.2.1 started
2020-01-08 13:25:32 UTC        Add
2020-01-08 13:25:32 UTC         xr-telnet
2020-01-08 13:28:26 UTC       Error on 0/1/CPU0: Insufficient disk space to install packages
2020-01-08 13:28:26 UTC        Packaging operation 3.2.1 failed: Disk space check failed
on nodes: 0/1/CPU0
2020-01-08 13:28:26 UTC        Packaging operation 3.2.1 aborted: Automatically recovered
after failure

Location 0/1/CPU0
  Last event: Error (Insufficient disk space to install packages)
  Disk space pre-check failure:
     Phase: Download
     Required space: 140944B
     Available space: 110623B
```

show install log

To display the details of installation requests, use the **show install log** command in XR EXEC mode.

show install log [**all**] [**detail**] [**id number**] [**last** *number*] [**reverse**]

Syntax Description

all	(Optional) Displays a log for all install operations.
detail	(Optional) Displays details including impact to processes and nodes.
id *number*	(Optional) Displays the log information for the specified transaction ID.
last *number*	(Optional) Displays the log information for the last number of transactions.
reverse	(Optional) Displays the logs in reverse order.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Enter the **show install log** command with no arguments to display a summary of all installation operations, including the changes to files and the processes impacted by each request. Specify the **id** argument to display details for a specific operation.

Task ID

Task ID	Operations
pkg-mgmt	read

The following example shows the log information for a specific transaction ID:

```
Router# show install log detail id 1
```

show install request

To display the current status of install requests, use the **show install request** command in XR EXEC mode.

show install request { **brief** [**monitor**] | **location** *node-id* [**monitor**] | { **verbose location** *node-id* [**monitor**] | **monitor location** *node-id* } }

Syntax Description

brief	Displays install request information summary.
location *node-id*	Displays install information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation.
verbose	Displays detailed information about the install request.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.
Release 7.8.1	The keyword **monitor** was deprecated.

Usage Guidelines

Cisco IOS XR processes only one installation request per device at a time. The **show install request** command displays any incomplete request that is currently running.

Note When the command is running asynchonously, use **show install request** command to monitor the progress.

If the install operation encounters a failure, the **show install request** command, displays an error message. Use the **show install history last package** command to obtain complete details of the failure.

The following example shows sample output from the **show install request** command:

```
Router# show install request

Install operation 17 'install add /tftp://172.31.255.255/dir/19mcast' started by user
'user_b' at 14:38:45 UTC Thu Mar 30 2006. The operation is 1% complete 2,017KB
downloaded The operation can still be aborted.
```

The following example shows sample output from the **show install request** command when there is insufficient disk space in the root file system. An error message is displayed to indicate the insufficient disk space .

```
Router#show install request
Wed Jan  8 13:29:38.086 UTC
```

```
User request: install replace /harddisk:/replace.iso
Operation ID: 3.2
State:        Failure since 2020-01-08 13:28:26 UTC
Disk space check failed on nodes: 0/1/CPU0. Automatically recovered after failure, ready
for next user request.

Current activity:    Await user input
Time started:        2020-01-08 13:29:25

The following actions are available:
    install package add
    install package remove
    install package upgrade
    install package downgrade
    install package replace
    install package rollback
    install commit
```

show install rollback

To display rollback information, use the **show install rollback** command in XR EXEC mode.

show install rollback { **id** *number* **changes** | **list-ids** }

Syntax Description

id *number*	Displays information for the specified transaction ID.
changes	Display changes relative to currently installed packages.
list-ids	Displays a list of all rollback IDs.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to view the valid list of rollback transaction IDs:

```
Router# show install rollback list-ids
```

The following example shows how to view rollback information for a specific transaction ID:

```
Router# show install rollback id 1
```

The following example shows how to view the changes for a specific rollback transaction ID:

```
Router# show install rollback id 1 changes
```



Terminal Services Commands

- absolute-timeout, on page 454
- access-class, on page 455
- autocommand, on page 457
- banner exec, on page 459
- banner login, on page 461
- banner motd, on page 463
- clear line, on page 465
- clear line vty, on page 466
- disconnect-character, on page 467
- escape-character, on page 468
- exec-timeout, on page 469
- length, on page 470
- line, on page 471
- send, on page 472
- session-timeout, on page 473
- show line, on page 474
- show terminal, on page 477
- show cli submode-exit, on page 479
- show users, on page 480
- terminal length, on page 481
- terminal width, on page 482
- timestamp disable, on page 483
- transport input, on page 484
- transport output, on page 485
- transport preferred, on page 486
- vty-pool, on page 487
- cli interactive syntax check, on page 489
- cli whitespace completion, on page 490

absolute-timeout

To set the absolute timeout for line disconnection, use the **absolute-timeout** command in line template configuration mode. To remove the **absolute-timeout** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

absolute-timeout *minutes*

Syntax Description

minutes	Absolute timeout interval, in minutes. Range is from 10 to 10000.

Command Default

minutes: 1440

Command Modes

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **absolute-timeout** command to terminate the connection after the specified time has elapsed, regardless of whether the connection is being used at the time of termination. You can specify an absolute-timeout value for each port. The user is notified 20 seconds before the session is terminated.

The following example shows how to set the session timeout value to 2880 minutes (2 days) for the default line template:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# absolute-timeout 2880
```

access-class

To restrict incoming and outgoing connections using an IPv4 or IPv6 access list, use the **access-class** command in line template configuration mode. To remove the restriction, use the **no** form of this command.

access-class *list-name* {**in** | **out**}

Syntax Description

list-name	IPv4 or IPv6 access list name.
in	Filters incoming connections.
out	Filters outgoing connections.

Command Default

No access class is set.

Command Modes

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **access-class** command to restrict incoming or outgoing connections to addresses defined in an access list. Use the **ipv4 access-list** or **ipv6 access-list** command to define an access list by name.

Note To restrict access of incoming or outgoing connections over IPv4 and IPv6, the IPv4 access list and IPv6 access list must share the same name.

The following example shows how to specify an access class assigned to outgoing connections for the default line template:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# access-class vtyacl out
```

The following sample output from the **show ipv4 access-lists** command displays the IPv4 access list named vtyacl:

```
RP/0/RP0/CPU0:router# show ipv4 access-lists vtyacl

ipv4 access-list vtyacl
 10 permit ip host 10.32.162.48 any
 20 permit ip host 10.20.49.170 any
 30 permit ip host 10.60.3.5 any
```

The following sample output from the **show ipv6 access-lists** command displays the IPv6 access list name vtyacl:

```
RP/0/RP0/CPU0:router# show ipv6 access-lists vtyacl

ipv6 access-list vtyacl
 10 permit ipv6 host 2001:db8:2222:: any
 20 permit ipv6 host 2001:db8:0:4::2 any
```

autocommand

To automatically run one or more commands after a user logs in to a vty terminal session, use the **autocommand** command in line default or line template configuration mode. To remove the **autocommand** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

autocommand *command*

Syntax Description

command	Command or command alias to be executed on user login to a vty session.

Command Default

No default behavior or values

Command Modes

Line template configuration

Line default configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **autocommand** command to automatically run a command or command alias when a user logs in to a vty session. To run multiple commands, use a command alias for the *command* argument. When the user logs in, the commands included in the alias are run sequentially.

Note The **autocommand** command is supported on vty connections only; it is not supported on console or aux line connections. Use this command to automatically run a command after user login.

The following example shows how to use the **autocommand** command to automatically run the **show ip interface brief** command when a user logs in to a default vty session:

```
RP/0/RP0/CPU0:router# configure terminal
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# autocommand show ip interface brief
RP/0/RP0/CPU0:router(config-line)# end

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes

RP/0/RP0/CPU0:router# exit

  <Your 'TELNET' connection has terminated>
```

The following example shows how to disable the feature using the **no** form of the **autocommand** command. In this example, the autocommand for the **show ip interface brief** command is disabled. When the user logs out, and logs back in, the **autocommand** command does not run.

```
RP/0/RP0/CPU0:router# configure terminal
RP/0/RP0/CPU0:router(config)# line default
```

```
RP/0/RP0/CPU0:router(config-line)# no autocommand ?

  LINE  Appropriate EXEC command

RP/0/RP0/CPU0:router(config-line)# no autocommand show ip interface brief
RP/0/RP0/CPU0:router(config-line)# end

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes

RP/0/RP0/CPU0:router# exit

<Your 'TELNET' connection has terminated>

User Access Verification

Username: lab
Password:
RP/0/RP0/CPU0:router#
```

banner exec

To create a message that is displayed when an EXEC process is created (an EXEC banner), use the **banner exec** command in XR Config mode. To delete the EXEC banner, use the **no** form of this command.

banner exec *delimiter message delimiter*

Syntax Description

delimiter	Delimiting character is (c).
message	Message text. Text may include tokens in the form $(*token*) in the message text. Tokens are replaced with the corresponding configuration variable. Tokens are described in *banner exec tokens*.

Command Default

No EXEC banner is displayed.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **banner exec** command to specify a message that is displayed when an EXEC process is created (a line is activated or an incoming connection is made to a vty). Follow this command with one or more blank spaces and the delimiting character (c). After entering one or more lines of text, terminate the message with the delimiting character (c).

When a user connects to a router, the message-of-the-day (MOTD) banner appears first, followed by the login banner and prompts. After the user logs in to the router, the EXEC banner or incoming banner is displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner is displayed. For all other connections, the router displays the EXEC banner.

Use tokens in the form $(*token*) in the message text to customize the banner. Tokens display current configuration variables, such as the router hostname and IP address.

The tokens are described in this table.

Table 38: banner exec Tokens

Token	Information Displayed in the Banner
$(hostname)	Displays the hostname for the router.
$(domain)	Displays the domain name for the router.
$(line)	Displays the vty or tty (asynchronous) line number.

The following example shows how to set an EXEC banner that uses tokens:

```
RP/0/RP0/CPU0:router(config)# banner exec c
```

```
  Enter TEXT message.  End with the character 'c'.

THIS IS AN EXEC BANNER
c
```

banner login

To create a customized banner that is displayed before the username and password login prompts, use the **banner login** command in XR Config mode. To disable the login banner, use **no** form of this command.

banner login *delimiter message delimiter*

Syntax Description

delimiter	Delimiting character is (c).
message	Message text. You can include tokens in the form $(*token*) in the message text. Tokens are replaced with the corresponding configuration variable. Tokens are described in *banner login tokens*.

Command Default

No login banner is displayed.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Follow the **banner login** command with one or more blank spaces and the delimiting character (c). Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character (c).

When a user connects to the router, the message-of-the-day (MOTD) banner (if configured) appears first, followed by the login banner and prompts. After the user successfully logs in to the router, the EXEC banner or incoming banner is displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner is displayed. For all other connections, the router displays the EXEC banner.

To customize the banner, use tokens in the form $(*token*) in the message text. Tokens display current configuration variables, such as the router hostname and IP address.

Tokens are described in the this table.

Table 39: banner login Tokens

Token	Information Displayed in the Banner
$(hostname)	Displays the hostname for the router.
$(domain)	Displays the domain name for the router.
$(line)	Displays the vty or tty (asynchronous) line number.

The following example shows how to set a login banner:

```
RP/0/RP0/CPU0:router(config)# banner login c

Enter TEXT message. End with the character 'c'.
THIS IS A LOGIN BANNER
```

c

banner motd

To create a message-of-the-day (MOTD) banner, use the **banner motd** command in XR Config mode. To delete the MOTD banner, use the **no** form of this command.

banner motd *delimiter message delimiter*

Syntax Description

delimiter	Delimiting character is (c).
message	Message text. You can include tokens in the form $(*token*) in the message text. Tokens are replaced with the corresponding configuration variable.

Command Default

No MOTD banner is displayed.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Follow this command with one or more blank spaces and the delimiting character (c). Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character (c).

This MOTD banner is displayed to all terminals connected and is useful for sending messages that affect all users (such as impending system shutdowns). Use the **no banner motd** command to disable the MOTD banner on a line.

When a user connects to the router, the MOTD banner (if configured) appears first, followed by the login banner and prompts. After the user successfully logs in to the router, the EXEC banner or incoming banner is displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner is displayed. For all other connections, the router displays the EXEC banner.

To customize the banner, use tokens in the form $(token) in the message text. Tokens display current configuration variables, such as the router hostname and IP address.

Tokens are described in this table.

Table 40: banner motd Tokens

Token	Information Displayed in the Banner
$(hostname)	Displays the hostname for the router.
$(domain)	Displays the domain name for the router.
$(line)	Displays the vty or tty (asynchronous) line number.

The following example shows how to configure an MOTD banner with a token:

```
RP/0/RP0/CPU0:router(config)# banner motd c
```

```
  Enter TEXT message.  End with the character 'c'.
Notice: all routers in $(domain) will be upgraded beginning April 20
c
```

clear line

To clear an auxiliary or console line to an idle state, use the **clear line** command in XR EXEC mode.

clear line {**aux** | **console**} **location** *node-id*

Syntax Description

aux	Clears the auxiliary line.
console	Clears the console line.
location *node-id*	Specifies the location of a route processor (RP) where the auxiliary or console line to be cleared resides. The *node-id* argument is entered in the *rack / slot / module* notation.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows how to clear the console line, putting it in an idle state:

```
RP/0/RP0/CPU0:router# clear line console location 0/RP1/CPU0
```

clear line vty

To clear a virtual terminal line (vty) to an idle state, use the **clear line vty** command in XR EXEC mode.

clear line vty *line-number*

Syntax Description

line-number	Line number in the range from 0 to 99.

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show users** command to determine the origin of the connection and which lines to clear. When a line is cleared to an idle state, the user on the other end of the connection receives notice that the connection was closed by a foreign host.

The following example shows how to reset vty 3 to the idle state:

```
RP/0/RP0/CPU0:router# clear line vty 3
```

disconnect-character

To define a character to disconnect a session, use the **disconnect-character** command in line template configuration mode. To remove the **disconnect-character** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

disconnect-character [*numbercharacter*]

Syntax Description

number	(Optional) ASCII decimal equivalent of the disconnect character. Range is from 0 through 255.
character	(Optional) Disconnect character.

Command Default

No disconnect character is defined.

Command Modes

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The break character is represented by 0; null cannot be represented.

The following example shows how to set the disconnect character for the default line template to the Esc character, which is the ASCII decimal equivalent 27:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# disconnect-character 27
```

escape-character

To define a character to escape a session, use the **escape-character** command in line template configuration mode. To remove the **escape-character** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

escape-character {**break***numbercharacter* | **default** | **none**}

Syntax Description

break	Sets the escape character to the Break key.
number	ASCII decimal equivalent of the escape character. Range is from 0 through 255.
character	Escape character.
default	Specifies the default escape character (^^X).
none	Disables the escape function.

Command Default

The default escape character is ^^X.

Command Modes

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **escape-character** command to define an escape character sequence that is different from the default. Use the escape character to exit from an existing connection and return to the EXEC prompt.

The default escape key sequence is Ctrl-Shift-6, X (^^X).The **escape-character** command with the **default** keyword sets the escape character to the Break key (the default setting for the Break key is Ctrl-C).

The following example shows how to set the escape character for the default line template to Ctrl-P, which is the ASCII decimal character 16:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# escape-character 16
```

exec-timeout

To set the interval that the EXEC command interpreter waits until user input is detected, use the **exec-timeout** command in the appropriate line configuration mode. To remove the **exec-timeout** command from the running configuration and restore the system to its default condition, use the **no** form of this command.

exec-timeout *minutes seconds*

Syntax Description

minutes	Minutes for the wait interval. Range is from 0 to 35791.
seconds	Seconds for the wait interval. Range is from 0 to 2147483.

Command Default

minutes: 10

seconds: 0

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

If no input is detected during the interval, the EXEC facility resumes the current connection. If no connections exist, the EXEC facility returns the terminal to the idle state and disconnects the incoming session. To disable the EXEC timeout function so that the EXEC session never timeouts, enter the following command:

```
exec-timeout 00
```

The following example shows how to set the timeout interval for the console line template to 60 minutes, 0 seconds:

```
RP/0/RP0/CPU0:router(config)# line console
RP/0/RP0/CPU0:router(config-line)# exec-timeout 60 0
```

length

To set the number of lines that display at one time on the screen, use the **length** command in line template configuration mode. To remove the **length** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

length *lines*

Syntax Description

lines	Number of lines that displays on a screen. Range is from 0 through 512. 0 specifies no pausing. The default is 24.

Command Default

lines : 24

Command Modes

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **length** command to modify the default length setting for the specified line template. The length setting determines when the screen pauses during the display of multiple-screen output. Specifying a value of 0 for the lines argument prevents the router from pausing between screens of output.

The following example shows how to set the length of the default line template to 33 lines:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# length 33
```

line

To specify the console, the default, or a user-defined line template and enter line template configuration mode, use the **line** command in

mode.

line {**console** | **default** | **template** *template-name*}

Syntax Description

console	Specifies the line template for the console line.
default	Specifies the default line template.
template *template-name*	Specifies a user-defined line template to be applied to a vty pool.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **line** command to specify a line template type and enter into line template configuration mode. Line templates are a collection of attributes used to configure and manage physical terminal line connections (the console and auxiliary ports) and vty connections. The following templates are available in Cisco IOS XR software:

- Default line template—The default line template that applies to a physical and virtual terminal lines.
- Console line template—The line template that applies to the console line.
- User-defined line templates—User-defined line templates that can be applied to a range of virtual terminal lines.

The following example shows how to enter line template configuration mode to allow configuration changes to be made to the default line template:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)#
```

send

To send messages to one or all terminal lines, use the **send** command in

mode.

send {***line-number** | **aux 0** | **console 0** | **vty** *number*}

Syntax Description

*****	Sends a message to all tty lines.
line-number	Line number to which the message is sent. A number from 0 to 101.
aux 0	Sends a message to the auxiliary line.
console 0	Sends a message to the console line.
vty *number*	Sends a message to a virtual terminal line (vty). Range is 0 to 99.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The system prompts for the message, which can be up to 500 characters long. Enter **Ctrl-Z** to end the message. Enter **Ctrl-C** to abort this command.

The following example shows how to send a message to all lines:

```
RP/0/RP0/CPU0:router# send *

Enter message, end with CTRL/Z; abort with CTRL/C:
The system will be shut down in 10 minutes for repairs.^Z
Send message? [confirm]
RP/0/RP0/CPU0:router#

***
***
*** Message from tty to all terminals:
***
The system will be shut down in 10 minutes for repairs.
```

session-timeout

To set the timeout interval for all outgoing connections from the current terminal, use the **session-timeout** command in the appropriate line configuration mode. To remove the **session-timeout** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

session-timeout *minutes* [**output**]

Syntax Description

minutes	Timeout interval, in minutes. Range is 0 to 35791. The default is 10.
output	(Optional) Specifies that when traffic is sent to an asynchronous line from the router (within the specified interval), the connection is retained.

Command Default

minutes : 10

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **session-timeout** command to set the interval that Cisco IOS XR software waits for traffic before closing the connection to a remote device and returning the terminal to an idle state. If the **output** keyword is not specified, the session timeout interval is based solely on detected input from the user. If the keyword is specified, the interval is based on input and output traffic.

The following example shows how to set the session timeout value for the default line template to 120 minutes (2 hours):

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# session-timeout 120
```

show line

To display the parameters of terminal lines, use the **show line** command in

mode.

show line [**aux location** *node-id* | **console location** *node-id* | **vty** *number*]

Syntax Description

aux	(Optional) Displays the terminal line parameters for the auxiliary line.
location *node-id*	(Optional) Specifies the location for the route processor (RP) on which the auxiliary or console port resides. The *node-id* argument is entered in the *rack/slot* notation.
console	(Optional) Displays the terminal line parameters for the console line.
vty *number*	(Optional) Specifies a virtual terminal line (vty) number. Range is from 0 through 99.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following example shows sample output from the **show line** command. The asterisk (*) indicates the current terminal session.

```
RP/0/RP0/CPU0:router# show line

    Tty             Speed     Modem  Uses   Noise Overruns           Acc I/O
    aux0_0_0         9600       -     -       -      0/0              -/-
*   con0_0_0         9600       -     -       -      0/0              -/-
    vty0             0/0        -     -       -      0/0              -/-
    vty1             0/0        -     -       -      0/0              -/-
    vty2             0/0        -     -       -      0/0              -/-
    vty3             0/0        -     -       -      0/0              -/-
    vty4             0/0        -     -       -      0/0              -/-
    vty100           0/0        -     -       -      0/0              -/-
    vty101           0/0        -     -       -      0/0              -/-
    vty102           0/0        -     -       -      0/0              -/-
    vty103           0/0        -     -       -      0/0              -/-
    vty104           0/0        -     -       -      0/0              -/-
    vty105           0/0        -     -       -      0/0              -/-
```

Table 41: show line Field Descriptions

Field	Description
Tty	Available ttys and vtys.
Speed	Baud rate that the inbound serial connection is using, in bps.
Modem	Not implemented.
Uses	Not implemented.
Noise	Not implemented.
Overruns	Hardware Universal Asynchronous Receiver/Transmitter (UART) overruns or software buffer overflows, both defined as the number of overruns or overflows that have occurred on the specified line since the system was restarted. Hardware overruns are buffer overruns; the UART chip has received bits from the software faster than it can process them. A software overflow occurs when the software has received bits from the hardware faster than it can process them.
Acc I/O	Not implemented.

The following example shows sample output from the **show line** command with the console line specified:

```
RP/0/RP0/CPU0:router# show line console location 0/rp0/cpu0

    Tty             Speed     Overruns             Acc I/O
    con0/RP0/CPU0    9600          0/0                 -/-

  Line con0_RP0_CPU0, Location "0/RP0/CPU0", Type "Console"
  Length: 24 lines, Width: 80 columns
  Baud rate (TX/RX) is 9600, 1 parity, 2 stopbits, 8 databits
  Template: console
  Capabilities: Timestamp Disabled
  Allowed transports are none.
```

Table 42: show line location Field Descriptions

Field	Description
Tty	Unique identifier of the tty; it contains the type of tty and, for physical ttys, it indicates the physical location of the tty.
Speed	Baud rate that the inbound serial connection is using in bps.
Overruns	Hardware UART overruns or software buffer overflows, both defined as the number of overruns or overflows that have occurred on the specified line since the system was restarted. Hardware overruns are buffer overruns; the UART chip has received bits from the software faster than it can process them. A software overflow occurs when the software has received bits from the hardware faster than it can process them.
Acc I/O	Not implemented.

Field	Description
Line	Unique identifier of the TTY. This field displays the type of TTY and the physical location of physical TTYs.
Location	Location of the line.
Type	Line type.
Length	Length of the terminal or screen display, in rows.
Width	Width of the terminal or screen display, in columns.
Baud rate (TX/RX)	Transmit rate/receive rate of the line, in bps.
parity	Parity bits value used for physical terminal connections.
stopbits	Stop bits value used for physical terminal connections.
databits	Data bits value used for physical terminal connections.
Template	Line template being sourced by the particular connection.
Config	Configuration applied to the tty. This field indicates the allowed incoming transports that can be used to access the router from this tty.
Allowed transports are	Incoming transport protocols that can be used by this tty to access the router.

show terminal

To obtain information about the terminal configuration attribute settings for the current terminal line, use the **show terminal** command in

mode.

show terminal

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

This example shows sample output from the **show terminal** command:

```
RP/0/RP0/CPU0:router# show terminal

Line vty0, Location "10.56.249.67", Type "VTY"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 0, 0 parity, 0 stopbits, 0 databits
Template: default
Capabilities: Timestamp Disabled
Allowed transports are telnet ssh.
```

Table 43: show terminal Field Descriptions

Field	Description
Line	Line that is currently being used.
Location	Location of the terminal accessing the router.
Type	Type of line.
Length	Length of the terminal or screen display, in rows.
Width	Width of the terminal or screen display, in columns.
Baud rate (TX/RX)	Transmit or receive rate of the line, in bps.
parity	Parity bits value used for physical terminal connections.
stopbits	Stop bits value used for physical terminal connections.

Field	Description
databits	Data bits value used for physical terminal connections.
Template	Line template being sourced by the particular connection.
Config	Configuration applied to the tty. This field indicates the allowed incoming transports that can be used to access the router from this tty.
Allowed transports are	Incoming transport protocols that can be used by this tty to access the router.

show cli submode-exit

To display the status of terminal submode-exit configuration, use **show cli submode-exit status** command in XR EXEC mode.

show cli submode-exit status

Syntax Description

This command has no keywords or arguments.

Command Default

No default behavior or values.

Command Modes

XR EXEC mode.

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The following block shows the output from the **show cli submode-exit status** command:

```
RP/0/RP0/CPU0:router #show cli submode-exit status
Global  submode exit feature is enabled.
Session submode exit feature is enabled
```

show users

To display information about the active lines on the router, use the **show users** command in System Admin EXEC or mode.

show users

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **show users** command to display the line number, connection name, idle time, hosts, and terminal location. An asterisk (*) indicates the current terminal session.

Note To display all user groups and task IDs associated with the currently logged-in user, use the **show user** command in EXEC mode. See the *Authentication, Authorization, and Accounting Commands on Cisco IOS XR Software* module in *System Security Command Reference for Cisco 8000 Series Routers*.

The following example shows sample output identifying an active vty terminal session:

```
RP/0/RP0/CPU0:router# show users

     Line            User                Service  Conns   Idle        Location
     con0_RP0_CPU0   cisco               hardware     0  18:33:48
     vty0            cisco               telnet       0  00:30:36    10.33.54.132
  *  vty1            cisco               telnet       0  00:00:00    10.33.54.132
```

Table 44: show users Command Output Field Descriptions

Field	Description
Line	All current connections. An asterisk (*) indicates the active connection.
User	Username of the user logged into the line.
Service	Physical or remote login service used.
Conns	Number of outgoing connections.
Idle	Interval (in hours:minutes:seconds) since last keystroke.
Location	IP address of remote login host. For local (physical) terminal connections, this field is blank.

terminal length

To set the number of lines that display at one time on the screen for the current terminal session, use the **terminal length** command in

mode.

terminal length *lines*

Syntax Description

lines	Number of lines that display on a screen. Range is from 0 through 512.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **terminal length** command to set the terminal length value for only the current terminal session and not save it to the running configuration. Exiting from the terminal session returns the terminal length value to the value configured with the **length** command.

Specifying a value of 0 for the *lines* argument prevents the router from pausing between screens of output.



Note The **terminal** commands are active for the current terminal session only. To apply a setting to all sessions, use the **line** commands.

This example shows how to set the length for the current terminal session to 120 lines:

```
RP/0/RP0/CPU0:router# terminal length 120
```

terminal width

To set the width of the display terminal for the current terminal session, use the **terminal width** command in mode.

terminal width *characters*

Syntax Description

characters	Number of characters to display on a screen. Range is from 0 to 512.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **terminal width** command to set the terminal width value for only the current terminal session and not save it to the running configuration. Exiting from the terminal session returns the terminal width value to the value configured with the **width** (display) command.

The following example shows how to set the terminal width for the current terminal session to 120 characters:

```
RP/0/RP0/CPU0:router# terminal width 120
```

timestamp disable

To disable time-stamp recording at the top of each command output, use the **timestamp disable** command in the appropriate line configuration mode. To reenable time-stamp recording if disabled, use the **no** form of this command.

timestamp disable

Syntax Description

This command has no keywords or arguments.

Command Default

Time-stamp recording at the top of each command output is enabled.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

By default, the time stamp is displayed at the top of any command output. The time stamp records the time at which the command was issued. You can use the **snmp-server view** command to disable this setting so that the time stamp does not appear at the top of the command output. This setting applies to all command outputs on any terminal line to which the current line template applies.

This example shows how to disable time-stamp recording for the console line template:

```
RP/0/RP0/CPU0:router(config)# line console
RP/0/RP0/CPU0:router(config-line)#  timestamp disable
```

transport input

To define the transport protocols that can be used to access the router, use the **transport input** command in the appropriate line configuration mode. To change or remove the protocol, use the **no** form of this command.

transport input {**all** | **none** | **ssh** | **telnet**}

Syntax Description

all	Specifies the Secure Shell (SSH) and Telnet protocols.
none	Specifies that the router rejects incoming SSH and Telnet transport protocol connections.
ssh	Specifies the SSH transport protocol.
telnet	Specifies the Telnet transport protocol.

Command Default

All protocols are allowed on the line.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To be accepted, incoming network connections to an asynchronous port (terminal line) must use a transport protocol specified with the **transport input** command. This command can be useful in limiting the acceptable transport protocols to include or exclude those used by different types of users, or to restrict a line to secure connections (SSH connections).

This example shows how to set the transport input setting for the default line template to SSH connections:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# transport input ssh
```

transport output

To specify the transport protocols that can be used for outgoing connections from a line, use the **transport output** command in the appropriate line configuration mode. To change or remove the protocol, use the **no** form of this command.

transport output {**all** | **none** | **ssh** | **telnet**}

Syntax Description

all	Specifies the Secure Shell (SSH) and Telnet transport protocols.
none	Specifies that the router rejects outgoing SSH and Telnet transport protocol connections.
ssh	Specifies the SSH transport protocol.
telnet	Specifies the Telnet transport protocol.

Command Default

All protocols are allowed on the line.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Any settings made with the **transport output** command override settings made with the **transport preferred** command.

This example shows how to set the default line template to prevent any outgoing transport protocol connections:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# transport output none
```

transport preferred

To specify the default outgoing transport protocol to be used for initiating network connections, use the **transport preferred** command in the appropriate line configuration mode. To change or remove the protocol, use the **no** form of this command.

transport preferred {**none** | **ssh** | **telnet**}

Syntax Description

none	Disables the feature.
ssh	Specifies the Secure Shell (SSH) transport protocol.
telnet	Specifies the Telnet transport protocol.

Command Default

No transport protocol is set as the default outgoing protocol.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **transport preferred** command to provide a default transport protocol to use when initiating outgoing connections. The preferred outgoing transport protocol specified with the **transport preferred** command enables you to initiate an outgoing connection without explicitly specifying the transport protocol.

Cisco IOS XR software assumes that any unrecognized command is a hostname and the software attempts a connection. When the protocol is set to **none**, the system ignores unrecognized commands entered at the EXEC prompt, and does not attempt a connection.

The default setting, the same as using the **transport preferred** command with the **none** keyword, prevents errant connection attempts.

The following example shows how to set the preferred transport setting for the default line template to SSH:

```
RP/0/RP0/CPU0:router(config)# line default
RP/0/RP0/CPU0:router(config-line)# transport preferred ssh
```

vty-pool

To create or modify a virtual terminal line (vty) pool, use the **vty-pool** command in XR Config mode. To delete a vty pool, use the **no** form of this command.

vty-pool {**default** | **eem***pool-name*} *first-vty last-vty* [**line-template** {**default***template-name*}]

Syntax Description

default	Specifies the default vty pool.
eem	Specifies the embedded event manager vty pool.
pool-name	User-defined vty pool.
first-vty	First vty line in the pool. • For the default vty pool, you must specify 0 for the first vty line. • For a user-defined vty pool, the range is 5 to 99. • For the embedded event manager vty pool, you must specify 100 for the first vty line.
last-vty	Last vty line in the pool. • The default vty pool must contain at least five vtys. Range is 4 to 99. • For a user-defined vty pool, the range is 5 to 99. • The embedded event manager vty pool must contain at least six vtys. Range is 105 to 199.
line-template	(Optional) Specifies the terminal template to be used in the configuration of virtual terminals in the vty pool.
default	Specifies that the vty pool should reference the default template.
template-name	User-defined template to be applied to the vtys in the vty pool.

Command Default

default *vty-pool* : 5 vtys (vty 0 through 4) referencing the default line template.

eem *vty pool* : 6 vtys (vty 100 through 105) referencing the default line template.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

When creating or modifying vty pools, follow these usage guidelines:

- Before creating or modifying the vty pools, enable the Telnet server using the **telnet server** command in global configuration mode.
- The vty range for the default vty pool must start at vty 0 and must contain a minimum of five vtys.

- The vty range from 0 through 99 can reference the default vty pool.
- The vty range from 5 through 99 can reference a user-defined vty pool.
- The vty range from 100 is reserved for the embedded event manager vty pool.
- The vty range for embedded event manager vty pools must start at vty 100 and must contain a minimum of six vtys.
- A vty can be a member of only one vty pool. A vty pool configuration fails if the vty pool includes a vty that is already in another pool.

If you attempt to remove an active vty from the active vty pool when configuring a vty pool, the configuration for that vty pool fails.

This example shows how to configure a user-defined vty pool (test1) that contains vtys 10 through14 and references the user-defined line template test2:

```
RP/0/RP0/CPU0:router(config)# vty-pool test1 10 14 line-template test2
```

cli interactive syntax check

To enable interactive syntax checking, use the **cli interactive syntax check** command in the appropriate line configuration mode. To disable interactive syntax checking, use the **no** form of this command.

cli interactive syntax check

Syntax Description

This command has no keywords or arguments.

Command Default

Interactive syntax checking is disabled.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	No modification.
Release 24.4.1	This command was deprecated.

Usage Guidelines

Use the **cli interactive syntax check** command to check command syntax as you type. You are not allowed to enter incorrect syntax.

Task ID

Task ID	Operations
tty-access	read, write

The following example shows how to enable interactive syntax checking:

```
RP/0/RP0/CPU0:router(config)# line console
RP/0/RP0/CPU0:router(config-line)# cli interactive syntax check
```

Related Commands

Command	Description
cli whitespace completion, on page 490	Enables completion of a command when you type the space key.

cli whitespace completion

To enable completion of a command when you type the space key, use the **cli whitespace completion** command in the appropriate line configuration mode. To disable whitespace completion, use the **no** form of this command.

cli whitespace completion

Syntax Description

This command has no keywords or arguments.

Command Default

Whitespace completion is disabled.

Command Modes

Line console configuration

Line default configuration

Line template configuration

Command History

Release	Modification
Release 7.0.12	No modification.

Usage Guidelines

Use the **cli whitespace completion** command to complete the next word of the command syntax if you type the space key before completing the word. If more than one option is valid, all options are displayed for you to choose one.

Task ID

Task ID	Operations
tty-access	read, write

The following example shows how to enable whitespace completion:

```
RP/0/RP0/CPU0:router(config)# line console
RP/0/RP0/CPU0:router(config-line)# cli whitespace completion
```

Related Commands

Command	Description
cli interactive syntax check, on page 489	Enables interactive syntax checking.



Utility Commands

- utility cut, on page 492
- utility date, on page 496
- utility date set, on page 498
- utility fgrep, on page 499
- utility find, on page 501
- utility less, on page 503
- utility mv, on page 505
- utility sort, on page 506
- utility tail, on page 509
- utility uniq, on page 511
- utility wc, on page 513
- utility bc, on page 515
- utility df, on page 519
- utility du, on page 523
- utility egrep, on page 526
- utility head, on page 529
- utility which, on page 531
- utility xargs, on page 533

utility cut

To extract selected characters or fields from standard input or from a file, use the **utility cut** command in XR EXEC mode.

utility cut { {**list** *character-list* | **fields** *field-list* [**nodelim**] [**delimiter** *delimiter-character*]*WORD*} [**file** *input-file*] | **usage**}

Syntax Description

list *character-list*	(**-c**) Cuts out the characters that are located on each line as specified with the *character-list* argument. The *character-list* argument specifies the character positions or range of the characters to be cut. • Use a comma (,) to indicate more than one character. For example, **utility list 1,2,5** outputs the first, second, and fifth characters. • Use a dash (-) to indicate a range. For example, **utility list 1-64** outputs the first 64 characters of each line, **utility list 5-** outputs the fifth character to the end of the line. **Note** Lines are separated by a delimiter. The default delimiter is tab.
fields *field-list*	(**-f**) Cuts out the fields (lines) as indicated with the *field-list* argument. The *field-list* argument specifies the field numbers or ranges. For example, **utility field 2,9** outputs the second and ninth fields, **utility field 1-3** outputs the first three fields, **utility field -6** outputs the first six fields. **Note** The fields indicated by the *field-list* argument are assumed to be separated in the file by a delimiter character. The default delimiter is tab. Use the **delimiter** *delimiter* option to specify a delimiter character. Lines without field delimiters are processed unless the **nodelim** keyword is specified.
nodelim	(Optional) (**-s**) Ignores lines with no delimiter. Use this optional keyword when the **fields** *field-list* keyword and argument is specified.
delimiter *delimiter-character*	(Optional) (**-d**) Specifies an alternative delimiter to indicate the end of each field. Replace the *delimiter-character* argument with the character used as the delimiter.
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
file *input-file*	(Optional) Storage device and directory path of the text file used instead of the standard input (keyboard input). The syntax of the *input-file* argument is: *device* **:**[*/ directory-path*]*/ filename* The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.

usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If no file is specified, **keyboard** input (standard input) is used.

The default for delimiter is **tab**.

Command Modes

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **utility cut** command cuts out columns, fields, or characters displayed from standard input or from a file.

Use the **fields** *field-list* keyword and argument if the fields vary in length from line to line. (The lines must be separated by a delimiter character.) By default, the field delimiter character is the Tab key. Use the **delimiter** *delimiter-character* keyword and argument to specify a different delimiter.

Use the **list** *character-list* keyword and argument only if the fields are of a fixed length. Replace the *character-list* argument with the character positions to be extracted.

For the *character-list* argument, use a comma (,) to indicate more than one character, or use a dash (-) to indicate a range. For example, **utility list 1,2,5** outputs the first, second, and fifth characters, **utility list 1-64** outputs the first 64 characters of each line, **utility list 5-** outputs the fifth character to the end of the line.

You can also use the cut utility as a filter. If no files are specified, the keyboard input (standard input) is used.

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **fields** keyword can also be entered using the UNIX-equivalent (**-f**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility cut** command is entered with the **list** *character-list* keyword and argument to display the first 10 characters in each line. The output is from the results of the **show version** command, which is entered with the pipe (|) character:

```
RP/0/RP0/CPU0:router# show version | utility cut list 1-10

  Cisco IOS
  Copyright

  ROM: Syste

  router upt
  System ima

  cisco CRS-
  7457 proce

  16 Gigabit
  2 Ethernet
  20 Packet
  20 SONET/S
```

```
  2043k byte
  38079M byt
  1000592k b
  1000640k b

  Configurat
  Package ac
   --More--
```

In the following example, the **utility cut** command is used to extract fields from a file:

```
RP/0/RP0/CPU0:router# utility cut fields 1,5 delimiter : file disk0:/usr/passwd

  root:Super-User
  daemon:
  bin:
  sys:
  adm:Admin
  lp:Line Printer Admin
  uucp:uucp Admin
  nuucp:uucp Admin
  listen:Network Admin
  nobody:Nobody
```

In the following example, the **utility cut** command is used with the **delimiter** keyword to specify an alternative field delimiter:

```
RP/0/RP0/CPU0:router# utility cut fields 1,4,5 delimiter : file disk0:/usr/passwd

  root:1:Super-User
  daemon:1:
  bin:2:
  sys:3:
  adm:4:Admin
  lp:8:Line Printer Admin
  uucp:5:uucp Admin
  nuucp:9:uucp Admin
  listen:4:Network Admin
```

In the following example, a range of fields is specified:

```
RP/0/RP0/CPU0:router# utility cut fields 1-4 delimiter : file disk0:/usr/passwd

  root:x:0:1
  daemon:x:1:1
  bin:x:2:2
  sys:x:3:3
  adm:x:4:4
  lp:x:71:8
  uucp:x:5:5
  nuucp:x:9:9
  listen:x:37:4
```

In the following example, the **list** *character-list* keyword and argument are used to specify the character positions to be extracted:

```
RP/0/RP0/CPU0:router# utility cut list 1-30 file disk0:/usr/passwd
```

```
  root:x:0:1:Super-User:/:/sbin/
  daemon:x:1:1::/:
  bin:x:2:2::/usr/bin:
  sys:x:3:3::/:
  adm:x:4:4:Admin:/var/adm:
  lp:x:71:8:Line Printer Admin:/
  uucp:x:5:5:uucp Admin:/usr/lib
  nuucp:x:9:9:uucp Admin:/var/sp
  listen:x:37:4:Network Admin:/u
  nobody:x:60001:60001:Nobody:/:
  noaccess:x:60002:60002:No Acce
  nobody4:x:65534:65534:SunOS 4.

  ================================================================================
```

In the following example, the UNIX equivalent options are used directly. First, the **utility cut** command is entered with the **usage** keyword to display the possible options. Next, the **utility cut** command is entered with the options to extract the desired data.

```
RP/0/RP0/CPU0:router# utility cut usage

  cut -c list [file], cut -f list [-d delim] [-s] [file]

RP/0/RP0/CPU0:router# utility cut -f 1,4 -d : disk0:/usr/passwd

  root:1
  daemon:1
  bin:2
  sys:3
  adm:4
  lp:8
```

utility date

To display the date and time, use the **utility date** command in XR EXEC mode.

utility date {**format** *word* | **universal** | **usage***WORD*}

Syntax Description

format *word*	(Optional) (+) Specifies the format for the date display. Use the online help system to display the available format syntax for the *word* argument.
universal	(Optional) (**-u**) Displays the date in Coordinated Universal Time (UTC) instead of local time. UTC is the standard term for Greenwich Mean Time (GMT).
usage	(Optional) Displays the UNIX options supported by this command.
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.

Command Default

The date is displayed in local time.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **utility date** command displays the internal time and date for the router.

Date Format

Use the **format** *word* option to specify the format and content of the displayed date and time. The format is composed of ASCII characters and field descriptors prefaced with %, in a manner similar to a C-language printf() format specifier. In the output, each field descriptor is replaced by its corresponding value; all other characters are copied to the output without change. The format is specified using the following characters:

%C

Century in 'CC' form. For example: 20

%y

Year in 'YY' form. For example: 06

%m

Month in 'MM' form. For example: 08

%d

Date in 'DD' form. For example: 28

%H

Hour in 'hh (24 hr.)' form. For example: 18

%M

Minutes in 'mm' form. For example: 55

%S

seconds in 'ss' form. For example: 24

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **universal** keyword can also be entered using the UNIX-equivalent (**-u**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

This example shows how to display the router date and time using the **utility date** command:

```
RP/0/RP0/CPU0:router# utility date

  Fri Aug 04 11:53:38 UTC 2006
```

This example shows how to display the router date and time using a variety of options with the **format** keyword:

```
RP/0/RP0/CPU0:router# utility date format "%y%m%d"

  060828

RP/0/RP0/CPU0:router# utility date format "%y-%m-%d"

  06-08-28

RP/0/RP0/CPU0:router# utility date format "%C%y-%m-%d"

  2006-08-28

RP/0/RP0/CPU0:router# utility date format "%C%y-%m-%d:%H:%M:%S"

  2006-08-28:02:09:58

RP/0/RP0/CPU0:router# utility date format "DATE: %y-%m-%d %nTIME: %H:%M:%S"

  DATE: 06-09-17
  TIME: 12:42:24
```

utility date set

To set the router time, use the **utility date set** command in System Admin EXEC mode.

utility date set *hh*:*mm*:*ss*

Syntax Description

hh	Specifies the hour in 2-digit numerical format. Range is 00 to23.
mm	Specifies the minutes in 2-digit numerical format. Range is 0 to 59.
SS	Specifies the seconds in 2-digit numerical format. Range is 0 to 59.

Command Default

None

Command Modes

System Admin EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

A colon (:) is required between the entry for hour, minutes, and seconds.

Note Generally, if the system is synchronized by a valid outside timing mechanism, such as a Network Time Protocol (NTP) clock source, or if you have a networking device with calendar capability, you need not set the software clock. Use the **date** command or the **clock set** command if no other time sources are available.

Note To manually copy the hardware clock (calendar) settings into the software clock, use the **clock read-calendar** command in EXEC mode.

By default, the system makes a "slow adjustment" if the new time is in the range of the following:

- –2.5 minutes + old time
- 5 minutes + old time

In a slow adjustment, the clock speed increases by less than 100 percent or decreases by less than 50 percent over a period of time from 1 second to 5 minutes until the clock catches up with the new time. This slow adjustment does not cause major discontinuities in the time flow. Use the **-S0** option to disable the slow adjustment.

The following example shows how to set the time using the **utility date set** command:

```
RP/0/RP0/CPU0:router(admin)# utility date set 13:07:00

  Fri Sep 15 13:07:00 UTC 2006
```

utility fgrep

To search a file for a fixed character string, use the **utility fgrep** command in XR EXEC mode.

utility fgrep {**expr** *expression* | **script** *expression-file*} [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

utility fgrep *expression* [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

utility fgrep usage

Syntax Description

expr *expression*	(**-e**) A regular expression, whose type is determined by the -e and -f options. This form is used when only one expression is specified on the command line. Any names specified after this option are treated as input files.
script *expression-file*	(**-f**) A file containing a set of regular expressions, each separated by a new line. The type of the expressions is determined by the -e and -f options. This form is used when more than one expression is specified. You can specify more than one -f option. The syntax of the *expression-file* argument is: *device* **:**[*/ directory-path*]*/ filename*
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 20.
count	(Optional) (**-c**) Displays a count of selected lines.
linenum	(Optional) (**-n**) Before each output line, displays the line's line number.
matchfile	(Optional) (**-l**) (“el”) Displays only the names of files containing the selected lines.
matchline	(Optional) (**-x**) Includes only input lines selected against an entire fixed string or regular expression.
nocase	(Optional) (**-i**) Ignores uppercase and lowercase distinctions during comparisons.
nofile	(Optional) (**-h**) Displays results without a filename prefix attached to the matched lines. This option applies only when more than one file is searched.
reverse	(Optional) (**-v**) Selects only those lines that don't match the specified patterns.
file *search-file*	(Optional) The file used for the search. Replace the *search-file* argument with the device and directory path of the file. The syntax for the *search-file* argument is: *device* **:**[*/ directory-path*]*/ filename*
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

The keyboard input (standard input) is used if no files are specified.

If more than one input file is specified, then the filename is displayed before each line.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **utility fgrep** command searches files for a fixed character string (as opposed to grep and egrep, which search for a a pattern that matches an expression).

The results are displayed to the standard output (terminal screen).



Note The fgrep utility options are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **count** keyword can also be entered using the UNIX-equivalent (**-c**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

The following example, the **utility fgrep** command is used with the **nocase** and **linenum** keywords:

```
RP/0/RP0/CPU0:router# show version | utility fgrep expr uptime nocase linenum

  7:router uptime is 5 days, 20 hours, 10 minutes
```

utility find

To locate files within one or more directories, use the **utility find** command in XR EXEC mode.

utility find {**path** *directory-path* {*LINE* | **name** *filename-pattern* | **user** *user-id*} | **usage**}

Syntax Description

path *directory-path*	Specifies the storage device and directory for the file search. The search is performed for the specified directory and all subdirectories in that directory tree. If a directory path is not specified, then the search is performed in the current directory (a path of . [dot] is assumed).
LINE	(Optional) UNIX command-line expressions provided as a string.
name *filename-pattern*	(Optional) Searches for the name of the file. The *filename-pattern* argument is a regular expression string.
user *user-id*	(Optional) Searches for files belonging to a specific user. The *user-id* argument is the username of the file owner.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If a directory path is not specified, then the search is performed in the current directory.

If a **name** *filename-pattern* is not specified, then the search return all files in the specified directory.

If a user is not specified, then the search is performed for all users.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **utility find** command to locate files within one or more directories. You can perform the search for a specific directory (and its subdirectories). If a directory is not specified, then the search is performed for the current directory.

To search for a regular expression string, use the **name** *filename-pattern* keyword and argument. Replace the *filename-pattern* argument with the regular expression string. If this option is not used, then all files within the specified directory are displayed.

To search for files belonging to a specific user, use the *user-id* argument. If this option is not used, then files belonging to all users are displayed.

In the following example, the **utility find** command is used to locate files matching a pattern. In this example, all files ending in ".txt" are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr name *.txt
```

```
disk0:/usr/test2.txt
```

In the following example, the UNIX equivalent option is used to locate files matching a pattern. In this example, all files ending in ".txt" are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0: -name *.txt

disk0:/-base-3.8.0/etc/vim/doc/editing.txt
disk0:/-base-3.8.0/etc/vim/doc/help.txt
disk0:/-base-3.8.0/etc/vim/doc/intro.txt
disk0:/-base-3.8.0/etc/vim/doc/uganda.txt
disk0:/usr/test2.txt
```

In the following example, the files belonging to a specific user are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr user 0

disk0:/usr
disk0:/usr/passwd
disk0:/usr/test2.txt
```

In the following example, the UNIX equivalent option is used to display files belonging to a specific user:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr -user 0

disk0:/usr
disk0:/usr/passwd
disk0:/usr/test2.txt
```

utility less

To display a file page-by-page, use the **utility less** command in XR EXEC mode.

utility less {[**exitEOF**] [*WORD*] | **nocase** | **position** *line-number* | **startat** *string*} [**file** *source-file*]

Syntax Description

exitEOF	(Optional) (**-E**) Automatically exits the utility the first time an end-of-file is encountered.
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
nocase	(Optional) (**-i**) Ignores uppercase and lowercase distinctions during comparisons.
position *line-number*	(Optional) (**-j**) Uses the line at *line-number* on the screen to position matched lines during a patter search.
startat *string*	(Optional) (**-p**) Starts at the first occurrence of the pattern specified by the *string* argument in the file.
file *source-file*	(Optional) Specifies the storage device and directory path for the text file to be displayed. The default is standard input. The syntax for the *source-file* argument is: *device* **:**[/ *directory-path*]/ *filename*

Command Default

If no text file is specified, standard input is assumed.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **utility less**command to display files page by page. You can specify regular expressions for pattern matching using the **startat** keyword. You can scroll up as well as down. When you enter the less mode, commands are similar to the “vi” editor.

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

The following example, the **utility less** command is used to display the file “config_store”. Only part of the file is shown here.

```
RP/0/RP0/CPU0:router# utility less file disk0:/usr/config_store

  Last configuration change at Tue Feb 20 18:34:02 2007 by xxx
```

```
!
hostname H1
line console
 exec-timeout 600 0
 session-timeout 600
!
line default
 exec-timeout 600 0
 session-timeout 600
!
.
.
.
```

utility mv

To rename or move a file from one directory to another, use the **utility mv** command in XR EXEC mode.

utility mv {[*WORD* | **force** | **interactive**] **source** *source-file* **target** *target-file* | **usage**}

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
force	(Optional) (**-f**) Forces an overwrite if the target file already exists. There is no confirmation prompt.
interactive	(Optional) (**-i**) Specifies to prompt for confirmation before renaming a file.
source *source-file*	Specifies the storage device, directory, and filename for the file to be moved.
target *target-file*	Specifies the new storage device, directory, and filename for the file.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

No default behavior or values

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines



Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility mv** command is used to move the file "aaa" from disk0a: to disk1a:

```
RP/0/RP0/CPU0:router# utility mv source disk0a:/aaa target disk1a:/aaa
```

utility sort

To sort, merge, or sequence-check the lines in one or more files, or from the standard input, use the **utility sort** command in XR EXEC mode.

utility sort {[[*WORD*] | [[**dict**] [**fieldSep** *character*] [**ignoreblank**] [**key** *key-definition*] [**lowercase**] [**merge**] [**numeric**] [**outfile** *filename*] [**printable**] [**reverse**] [**unique**]]] [**file** *filename*] | **usage**}

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
dict	(Optional) (**-d**) Sorts in dictionary order. Uses only alphanumeric and blank characters in the sort operation.
fieldSep *character*	(Optional) (**-t**) Specifies a character as the field separator.
ignoreblank	(Optional) (**-b**) Ignores leading blank characters in field comparisons.
key *key-definition*	(Optional) (**-k**) Defines a key to be the sort key. The *key-definition* argument field is defined using the following syntax: *field_start* [*type_string*] [,*field_end*] [*type_string*] • *field_start* and *field_end*—Specifies the beginning and end of the key field. • *type_string*—Specifies attributes specific to the key. The *field_start* and *field_end* arguments are each specified by a pair of digits of the form m.n, where the m refers to the field starting after the mth field separator in a line. For field_start, the .n refers to the nth character of the specified field, and is taken as zero if not specified. For field_end, the .n refers to the nth character after the last character of the specified field, and is taken as zero if not specified. The *type_string* argument may be formed from the characters bdfinr, which apply their defined attributes to the determination of the key. **Note** When ordering options appear independent of key field specifications, the requested field ordering rules are applied globally to all sort keys. When attached to a specific key, the specified ordering options override all global ordering options for that key.
lowercase	(Optional) (**-f**) Folds uppercase letters into lowercase (ignores case and treats upper case characters the same as lowercase characters).
merge	(Optional) (**-m**) Merges sorted files. Assumes that the files are already sorted and so does not sort the files.
numeric	(Optional) (**-n**) Interprets the field as numeric and sorts in numeric order. Includes the sign and optional thousands separator. This keyword also ignores leading blank characters in field comparisons (implies the **ignoreblank** keyword).

outfile *filename*	(Optional) (**-o**) Writes the results to a file. The *filename* argument is the destination disk, directory, and filename. The *filename* argument can be the same as the source file.
printable	(Optional) (**-i**) Ignores all nonprintable characters.
reverse	(Optional) (**-r**) Reverses the sort order. The sort is ascending by default.
unique	(Optional) (**-u**) Suppresses all but one line in each set of lines having equal keys.
file *filename*	(Optional) Specifies a file to be sorted.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If no file is specified, then the standard input (keyboard) is used.

If an **outfile** *filename* keyword and argument is not specified, then the standard output (display) is used.

The file is sorted in ascending order.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility sort** command is used to sort the contents of the file "words.txt":

```
RP/0/RP0/CPU0:router# utility sort file disk0:/usr/words.txt

   The
   few
   inquires
  A
  Code.
  Date
  Done
  This
  best-selling
  bestseller
  book
  come
  concerning
  fiction,
  have
  its
  list
```

```
muscled
of
onto
our
the
way
way
work
```

utility tail

To copy the end portion of a file or the standard input, use the **utility tail** command in XR EXEC mode.

utility tail {[[*WORD*] | [**bytes**] [**continuous**] [**count** *number*]] [**file** *input-file*] | **usage**}

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
bytes	(Optional) (**-c**) Copies the end of the file measured in bytes. The default is lines.
continuous	(Optional) (**-f**) Continues to copy data from the end of the file after the last line is reached. The operation pauses for 1 second, and then resumes in a continuous loop. The input file must be a regular file, not a terminal or a FIFO special file (a named pipe).
count *number*	(Optional) (**-n**) Copies the number of lines (default) or bytes specified with the *number* argument. The range is 0 to 4294967295. By default, the last 10 lines are copied. The *number* argument is a decimal integer that defines the location in the file to begin copying: • Include the plus (+) character to copy from the beginning of the file. • Include the minus (-) character to copy from the end of the file. • Do not include a character to copy from the end of the file. **Note** Select the **bytes** keyword to copy the information measured in a count of bytes.
file *input-file*	(Optional) Directory path and filename for the input file. If no file is specified, then the standard input is used. The syntax for the *input-file* argument is: *device* **:**[*/ directory-path*]*/ filename* The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If the **utility tail** command is entered without keywords or arguments, the last 10 lines of the standard input are copied.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **utility tail** command to copy data from the end of a file. By default, the last 10 lines are copied. Use the **bytes** keyword to copy the data measured in bytes. Use the **count** *number* option to define the number of lines or bytes to copy. Use the **file** *filename* option to specify an input file.



Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility tail** command is used with the bytes keyword to display the last 10 bytes in the output:

```
RP/0/RP0/CPU0:router# show version | utility tail count 10 bytes

  .95.3-p8

RP/0/RP0/CPU0:router#
```

utility uniq

To display or remove repeated lines in a file, use the **utility uniq** command in XR EXEC mode.

utility uniq [[[*WORD*] | [**afterChars** *number*] [**afterField** *number*] [**count**] [**nonrepeating** | **repeating**]] [**infile** *input-file* **outfile** *output-file*] | **usage**]

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
afterChars *number*	(Optional) (**-s**) Ignores the first characters on each line of the input file. Use the *number* argument to specify the number of characters. The range is 0 to 4294967295.
afterField *number*	(Optional) (**-f**) Ignores the first fields on each line of the input file. Use the *number* argument to specify the number of fields. The range is 0 to 4294967295.
count	(Optional) (**-c**) Displays the number of times the line appeared in the input file at the beginning of each output line.
nonrepeating	(Optional) (**-u**) Displays only the nonrepeating lines from the input file (repeating lines are not displayed).
repeating	(Optional) (**-d**) Displays only the repeating lines from the input file (nonrepeating lines are not displayed).
infile *input-file*	(Optional) Specifies an input file for processing. The *input-file* argument specifies the device, directory, and filename of the input file. If no input file is specified, then the standard input (keyboard) is used. The syntax of the *input-file* argument is: *device* **:**[*/ directory-path*]*/ filename*. The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.
outfile *output-file*	(Optional) Specifies an output file. The *output-file* argument specifies the device, directory, and filename of the output file. If no file is specified, then the standard output (display) is used. The syntax of the *output-file* argument is: *device* **:**[*/ directory-path*]*/ filename*. The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If no input file is specified, then the standard input is used.

If no output file is specified, then the standard output is used.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **utility uniq** command to display only lines that are repeated in a file, or to display only lines that appear once. This utility compares only adjacent lines, so the file or standard input must be sorted.

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility uniq** command is used to display the repeating lines in the output of the **show environment** command:

```
RP/0/RP0/CPU0:router# show environment | utility uniq repeating

host    5V              4500,5500       4250,5750       4000,6000
fabricq 1.25V           1125,1375       1063,1438       1000,1500
fabricq 1.25V           1125,1375       1063,1438       1000,1500
ingress 1.25V           1125,1375       1063,1438       1000,1500
spa5    1.5V            1500,0          1575,1425       0,0
host    5V              4500,5500       4250,5750       4000,6000
fabricq 1.25V           1125,1375       1063,1438       1000,1500
fabricq 1.25V           1125,1375       1063,1438       1000,1500
ingress 1.25V           1125,1375       1063,1438       1000,1500
spa5    1.5V            1500,0          1575,1425       0,0
```

utility wc

To count words, lines, or bytes in a file, use the **utility wc** command in XR EXEC mode.

utility wc [[[*WORD*] | [**bytes**] [**lines**] [**words**]] [**file** *input-file*] | **usage**]

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
bytes	(Optional) (**-c**) Displays the number of bytes in each input file.
lines	(Optional) (**-l**) (-œel-?) Displays the number of lines in each input file.
words	(Optional) (**-w**) Displays the number of words in each input file.
file *input-file*	(Optional) Specifies the input file. The *input-file* argument specifies the device, directory, and filename of the input file. If no input file is specified, then the standard input (keyboard) is used. The syntax of the *input-file* argument is: *device* **:**[*/ directory-path*]*/ filename*. The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

Output is displayed in the order bytes, words, and lines, even if the options are entered in a different order.

Command Modes

XR EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Output is displayed in the following order:

- When keywords are entered, the output appears in the order bytes, words, and lines.
- When no keyword is entered, the output appears in the order lines, words, and bytes.
- When any UNIX equivalent options are entered, the output appears in the order specified by the options. For example, if the command **utility wc -w -l -c** is entered, the output appears in the order words, lines, and bytes.



Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility wc**command is issued to display the number of lines, words, and bytes in the output of the **show version** command:

```
RP/0/RP0/CPU0:router# show version | utility wc

        221     1160    10820
```

The output displays the following:

- 221 lines
- 1160 words
- 10820 bytes

In the following example, the **utility wc** command is entered with the **words** keyword to display the number of words in the output of the **show version** command:

```
RP/0/RP0/CPU0:router# show version | utility wc words

       1160
```

utility bc

To implement an arbitrary precision calculator, use the **utility bc** command in XR EXEC mode.

utility bc [**file** *input-file*]

Syntax Description

file *input-file* (Optional) Specifies the text file containing commands and function definitions to be interpreted by the bc utility.

After all files have been read, the bc utility reads input from the standard input (keyboard). If no files are specified, then only the standard input (keyboard) is used.

The syntax of the *input-file* argument is as follows: *device* **:**[*/ directory-path*]*/ filename*

Possible values of the *device:* argument are:

disk0:

Uses a file from disk0: file system.

disk0a:

Uses a file from disk0a: file system partition.

disk1:

Uses a file from disk1: file system.

disk1a:

Uses a file from disk1a: file system partition.

ftp:

Uses a file from an FTP network server. The syntax is **ftp:**[[[//*username*[:*password*]@]*location*]/*directory*]/*filename*

harddisk:

Uses a file from the hard disk drive file system (if present).

harddiska:

Uses a file from the hard disk partition (if present).

nvram:

Uses a file from the nvram: file system.

ipv4

Uses a file from an IPv4 access list or prefix list.

ipv6

Uses a file from an IPv6 access list or prefix list.

rcp:

Uses a file from a remote copy protocol (rcp) network server. The syntax is **rcp:**[[[//*username*@]*location*]/*directory*]/*filename*

tftp:

Uses a file from a TFTP network server. The syntax is **tftp:**[[//*location*]/*directory*]/*filename*

Use the online help (**?**) function to display the available devices and network protocols.

Command Default

If an input file is not specified, the standard input (keyboard) is used.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **utility bc** command to use the interactive, programmable calculator that supports a complete set of control structures, including functions. The utility first processes any specified files, and then reads input from the keyboard (standard input).

Input files (and standard input) are ASCII text files containing sequences of bc statements to be executed.



Note The bc utility uses the bc programming language, an arbitrary precision calculator language with syntax similar to the C programming language. The bc utility does not support character or string manipulation.

The bc utility supports:

- 26 functions
- 26 simple variables
- 26 array variables (up to 2048 elements per array).

The bc utility supports the following common programming language constructs:

- "if", "while", and "for" statements
- User-defined functions with parameters
- Local variables

Information About Supported Network Protocols

In the syntax for the **ftp:**, **rcp:**, and **tftp:** network protocols, the location is either an IP address or a hostname. The filename is specified relative to the directory used for file transfers.

When no device is specified, the current directory is used. To view the current directory, enter the **pwd** command.

Table 45: Network Protocols Supported by Cisco IOS XR Software

Prefix	Name	Description
tftp:	Trivial File Transfer Protocol	*TFTP* is a simplified version of FTP that allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password).
ftp:	File Transfer Protocol	*FTP* is an application protocol, part of the TCP/IP protocol stack, and is used for transferring files between network nodes. FTP requires a username and password.

Prefix	Name	Description
rcp:	remote copy protocol	*Rcp* is a protocol that allows users to copy files to and from a file system residing on a remote host or server on the network. Rcp uses TCP to ensure the reliable delivery of data. Rcp downloads require a username.

Task ID

Task ID	Operations
universal	execute

In the following example, the **utility bc** command is used to execute the bc statements contained in the ASCII text file exp.txt:

```
RP/0/RP0/CPU0:router# utility bc file disk0:/usr/exp.txt

50
15
25
3
17
```

utility df

To display the amount of disk space available for a directory or file, use the **utility df** command in XR EXEC mode.

utility df [[*WORD*] [**kbytes**] [**mountinfo**] [**vsfStats**] [**file** *input-file*] | **usage**]

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
kbytes	(Optional) (**-k**) Displays the sizes in 1-K blocks (1024-byte units) instead of the default of 512 byte blocks.
mountinfo	(Optional) (**-n**) Displays the file-system mountpoints and types only.
vsfStats	(Optional) (**-g**) Displays all statvfs() information.
file *input-file*	(Optional) Specifies the storage device and directory path of the device, directory, or file. When a directory or file is specified, the df utility displays the amount of space on the file system that contains the directory or file. If no files are specified, then only the standard input (keyboard) is used. The syntax of the *input-file* argument is as follows: *device* **:**[*/ directory-path*]*/ filename* The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

Information is displayed for all file systems.

The results are displayed in 512-byte blocks.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the (disk free) **utility df** command to display the amount of disk space available for a device, directory, or file. Enter the command without keywords or arguments to display information for all mounted file systems.

Use the **vsfStats** keyword to invoke the statvfs() function, which provides additional details for all mounted file systems.

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **kbytes** keyword can also be entered using the UNIX-equivalent (**-k**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID

Task ID	Operations
universal	execute

In the following example, the (disk free) **utility df** command is entered without keywords or arguments to display information for all file systems:

```
RP/0/RP0/CPU0:router(admin)# utility df

  /dev/hd0t6              77987744     61592  77926152       1%  /harddisk:/
  /nvram:                     4086        60      4026       2%
  /dev/disk1t6             2001280    382720   1618560      20%  /disk1:/
  /dev/disk0t6             2001184    533568   1467616      27%  /disk0:/
  /dev/fs0p1                121856        68    121787       1%  /bootflash:
```

See Table 47: utility df Column Descriptions (left to right), on page 521 for column descriptions.

In the following example, the **kbytes** keyword is used to display information in 1-K blocks (1024-byte units) instead of the default of 512-byte blocks:

```
RP/0/RP0/CPU0:router(admin)# utility df kbytes

  /dev/hd0t6              38993872     30796  38963076       1%  /harddisk:/
  /nvram:                     2043        30      2013       2%
  /dev/disk1t6             1000640    191360    809280      20%  /disk1:/
  /dev/disk0t6             1000592    266784    733808      27%  /disk0:/
  /dev/fs0p1                 60928        34     60893       1%  /bootflash:
```

See Table 47: utility df Column Descriptions (left to right), on page 521 for column descriptions.

In the following example, the **mountinfo** keyword is used to display file-system mountpoints and types only:

```
RP/0/RP0/CPU0:router(admin)# utility df mountinfo

  Filesystem           Mounted on        Type
  /dev/hd0t6           /harddisk:/       dos (fat32)
  /nvram:
  /dev/disk1t6         /disk1:/          dos (fat16)
  /dev/disk0t6         /disk0:/          dos (fat16)
  /dev/fs0p1           /bootflash:       flash
```

In the following example, the **vfsStats** keyword is used to invoke the statvfs() function, which provides additional details for all mounted file systems:

```
RP/0/RP0/CPU0:router(admin)# utility df vfsStats

  /dev/hd0t6   /harddisk:/
      Blocks:  9748468 total   9740769 avail   [4096-byte blocks]
      Files :        0 total         0 avail
      Type  : dos (fat32)
      Flags : 00000120 [32bit, noatime]

  /nvram:
```

```
    Blocks:     2043 total      2013 avail   [1024-byte blocks]
    Files :        0 total         0 avail
    Type  :
    Flags : 00000000 []

/dev/disk1t6   /disk1:/
    Blocks:    62540 total     50580 avail   [16384-byte blocks]
    Files :        0 total         0 avail
    Type  : dos (fat16)
    Flags : 00000120 [32bit, noatime]

/dev/disk0t6   /disk0:/
    Blocks:    62537 total     45863 avail   [16384-byte blocks]
    Files :        0 total         0 avail
    Type  : dos (fat16)
    Flags : 00000120 [32bit, noatime]

/dev/fs0p1   /bootflash:
    Blocks: 62390272 total  62355240 avail   [1-byte blocks]
    Files :        2 total         0 avail
    Type  : flash
    Flags : 00000000 []
```

This table describes the significant fields shown in the display.

Table 46: utility df Field Descriptions

Field	Description
Files	Number of files in the file system.
Blocks	Amount of space available on the file system in 1-K blocks.
Total	Amount of disk space used by the directory or file.
Avail	Amount of space available for use by the directory or file on the file system.
Type	Type of file system.
Flags	Displays the file system properties.

In the following example, the **file** *source* keyword and argument are used to specify a directory:

```
RP/0/RP0/CPU0:router(admin)# utility df file disk0:/usr

  /dev/disk0t6             2001184    533568   1467616      27%  /disk0:/
```

This table describes the significant fields shown in the display.

Table 47: utility df Column Descriptions (left to right)

Field	Description
Filesystem	File system for the displayed information.
1k-blocks	Amount of space available on the file system in 1-K blocks.

Field	Description
Used	Amount of disk space used by the directory or file.
Available	Amount of space available for use by the directory or file on the file system.
Use%	Percentage of space used on the file system.
Mounted on	Storage device where the file system is mounted.

Related Commands

Command	Description
utility du, on page 523	Displays the amount of disk space used by one or more directories or files.

utility du

To display the amount of disk space used in a device, directory, or file, use the **utility du** command in XR EXEC mode.

utility du [[**all** | **specified**] [**kbytes** | **bytes** | **local**] [*WORD*] [**file** *source*] | **usage**]

Syntax Description

all	(Optional) (**-a**) Displays the disk space used for each file in the directory. By default, information is displayed only for the directory. Use the **all** keyword to display the total disk space used by all files in the directory, including the directory itself.
specified	(Optional) (**-s)** Displays the total disk space used for each specified file, rather than the totals for any subdirectories.
kbytes	(Optional) (**-k**) Displays the disk space used in 1-K blocks (1024-byte units) instead of the default of 512-byte blocks.
bytes	(Optional) (**-p**) Displays the disk space used in bytes (the default is 512-byte blocks). Also generates error messages for exiting files that cannot be displayed.
local	(Optional) (**-x**) Displays information for the local device only.
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
file *source*	(Optional) Displays the disk space used for a device, directory, or file. The syntax for the *source* argument is *device* **:** / *directory-path*[/ *filename*] The *device* argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (**?**) function to display the available storage devices and network protocols. Enter the **utility du** command without specifying a device, directory, or file to display information for the current directory, and subdirectories. (The command behaves as if the filename dot (.) is entered.)
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

Information for the current directory is displayed.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Enter the **utility du** command without specifying a file to display information for the current directory. The command behaves as if the filename dot (.) is entered.

Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **kbytes** keyword can also be entered using the UNIX-equivalent (**-k**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID

Task ID	Operations
universal	execute

In the following example, the **utility du** command is used to display the disk space used for the subdirectories in the MPLS package directory:

```
RP/0/RP0/CPU0:router# utility du file disk1:/asr9k-mpls-3.9.0.14I

Tue Jul 28 03:20:34.059 DST
         1607 /disk1:/asr9k-mpls-3.9.0.14I/schema
           83 /disk1:/asr9k-mpls-3.9.0.14I/lib/cerrno
          944 /disk1:/asr9k-mpls-3.9.0.14I/lib/mib
         3434 /disk1:/asr9k-mpls-3.9.0.14I/lib
        15974 /disk1:/asr9k-mpls-3.9.0.14I/bin
           14 /disk1:/asr9k-mpls-3.9.0.14I/mib
           14 /disk1:/asr9k-mpls-3.9.0.14I/placement
           14 /disk1:/asr9k-mpls-3.9.0.14I/startup
         1099 /disk1:/asr9k-mpls-3.9.0.14I/parser
           18 /disk1:/asr9k-mpls-3.9.0.14I/configs
           11 /disk1:/asr9k-mpls-3.9.0.14I/rules
           14 /disk1:/asr9k-mpls-3.9.0.14I/partitions
          156 /disk1:/asr9k-mpls-3.9.0.14I/etc/compat
          164 /disk1:/asr9k-mpls-3.9.0.14I/etc
           59 /disk1:/asr9k-mpls-3.9.0.14I/instdb_v
          147 /disk1:/asr9k-mpls-3.9.0.14I/lc/bin
            9 /disk1:/asr9k-mpls-3.9.0.14I/lc/startup
          164 /disk1:/asr9k-mpls-3.9.0.14I/lc
          103 /disk1:/asr9k-mpls-3.9.0.14I/instdb
        22697 /disk1:/asr9k-mpls-3.9.0.14I
```

In the following example, the **utility du** command is used to display the disk space used for a subdirectories:

```
RP/0/RP0/CPU0:router# utility du file disk0:/

-mpls-3.8.0/configs37 /disk0:/-mpls-3.8.0/configs
```

In the following example, the **utility du** command is used to display the disk space used for the current working directory:

```
RP/0/RP0/CPU0:router# cd disk0:/

-mpls-3.8.0
RP/0/RP0/CPU0:router# utility du

            160 ./schema
            104 ./lib/cerrno
```

```
  625 ./lib/mib
 2545 ./lib
 9658 ./bin
   39 ./startup
  840 ./parser
   37 ./configs
   35 ./mib
   35 ./rules
   34 ./partitions
  135 ./etc/compat
  167 ./etc
   66 ./instdb_v
  181 ./lc/bin
   33 ./lc/startup
  246 ./lc
  112 ./instdb
14006 .
```

Related Commands

Command	Description
utility df, on page 519	Displays the amount of disk space available for a directory or file.

utility egrep

To search a file or the results of standard input using full regular expressions, use the **utility egrep** command in XR EXEC mode.

utility egrep {**expr** *expression* | **script** *expression-file*} [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

utility egrep *expression* [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

utility egrep usage

Syntax Description

expr *expression*	(**-e**) A regular expression. This form is used when only one expression is specified on the command line. Any names specified after this option are treated as input files.
script *expression-file*	(**-f**) A file containing a set of regular expressions, each separated by a new line. The type of the expressions is determined by the -e and -f options. This form is used when more than one expression is specified. You can specify more than one -f option. The syntax of the *expression-file* argument is: [*device* **:**]/ *filename*
WORD	(Optional) UNIX command-line option string. The maximum number of characters is 20.
count	(Optional) (**-c**) Displays a count of selected lines.
linenum	(Optional) (**-n**) Before each output line, displays the line's line number.
matchfile	(Optional) (**-l**) (“el”) Displays only the names of files containing the selected lines.
matchline	(Optional) (**-x**) Includes only input lines selected against an entire fixed string or regular expression.
nocase	(Optional) (**-i**) Ignores uppercase and lowercase distinctions during comparisons.
nofile	(Optional) (**-h**) Displays results without a filename prefix attached to the matched lines. This option applies only when more than one file is searched.
reverse	(Optional) (**-v**) Selects only those lines that don't match the specified patterns.
file *search-file*	(Optional) The file used for the search. Replace the *search-file* argument with the device and directory path of the file. The syntax for the *search-file* argument is: [*device* **:**]/ *filename*.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If no files are specified, the keyboard input (standard input) is used.

If more than one input file is specified, then the filename is displayed before each line.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The grep utility searches files for character patterns using regular expressions and returns all lines that contain that pattern. The **utility egrep** command uses full regular expressions (expressions using the full set of alphanumeric and special characters) to match the patterns.

The results are displayed to the standard output (terminal screen).

Note The egrep utility options are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **count** keyword can also be entered using the UNIX-equivalent (**-c**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID

Task ID	Operations
universal	execute

In the following example, the **utility egrep** command is used to locate the regular expression "uptime". The **linenum** keyword is also entered to display the line number before each line of output.

```
RP/0/RP0/CPU0:router# show version | utility egrep expr uptime linenum

7:router uptime is 5 days, 19 hours, 27 minutes
```

In the following example, the **utility egrep** command is used to locate a regular expression in a file. In this example, all lines with "adm" are displayed. The * character is used as a wildcard.

```
RP/0/RP0/CPU0:router# utility egrep expr Adm* nofile file disk0:/usr/passwd

adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
ssadmin:x:901218:60001:Sun StorEdge(tm) Configuration Service Agent Admin:/:/bin/false
```

In the following example, the **nocase** keyword is used to ignore the character case:

```
RP/0/RP0/CPU0:router# utility egrep expr Adm* nocase file disk0:/usr/passwd

adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
ssadmin:x:901218:60001:Sun StorEdge(tm) Configuration Service Agent Admin:/:/bin/false
```

In the following example, the **linenum** keyword is used to append the line number to the beginning of each output line:

```
RP/0/RP0/CPU0:router# utility egrep expr Adm* linenum file disk0:/usr/passwd

  5:adm:x:4:4:Admin:/var/adm:
  6:lp:x:71:8:Line Printer Admin:/usr/spool/lp:
  7:uucp:x:5:5:uucp Admin:/usr/lib/uucp:
  8:nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
  9:listen:x:37:4:Network Admin:/usr/net/nls:
 15:ssadmin:x:901218:60001:Sun StorEdge(tm) Configuration Service Agent Admin:/:/bin/false
```

Related Commands

Command	Description
utility fgrep, on page 499	Searches a file for a fixed character string.

utility head

To copy bytes or lines at the beginning of a file or from the standard input, use the **utility head** command in XR EXEC mode.

utilityhead[*WORD* | [**bytes**] [**count** *number*][**file** *source*] | **usage**]

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
bytes	(Optional) (**-c**) Copies the data in bytes from the beginning of each specified file. The default setting is to copy lines of data.
count *number*	(Optional) (**-n**) Specifies the number of lines (default) or bytes to be copied. The *number* argument is an unsigned decimal integer. By default, the **utility head** command copies the first ten units (lines or bytes) of the file. Use the **count** *number* option to change the default.
file *source*	(Optional) Specifies the storage device, directory, and filename for the files. If a file is not specified, the standard input is used.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If the **utility head** command is entered without keywords or arguments, the first ten lines of the file or standard output are copied.

If no file is specified, then the standard input is used.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The **utility head** command copies the beginning bytes (default) or lines of one or more files to the standard output (usually the user interface display). Use the **bytes** or **lines** keywords to copy the data based on lines or bytes. Use the **count** *number* option to specify the number of bytes or lines to copy. By default, the **utility head** command copies the first 10 lines of each file.

If more than one file is selected, an identifying header is added before the output for each file. If no file is specified, then the standard input (keyboard) is used.



Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility head** command is used to display the first 15 lines from the output of the **show version** command:

```
RP/0/RSP0/CPU0:router# show version | utility head count 15

Tue Jul 28 06:15:44.736 DST

Cisco IOS XR Software, Version 3.9.0.14I[DT_IMAGE]
Copyright (c) 2009 by Cisco Systems, Inc.

ROM: System Bootstrap, Version 1.1(20090521:183759) [ASR9K ROMMON],

PE44_ASR-9010 uptime is 1 week, 6 days, 14 hours, 54 minutes
System image file is "bootflash:disk0/asr9k-os-mbi-3.9.0.14I/mbiasr9k-rp.vm"

cisco ASR9K Series (MPC8641D) processor with 4194304K bytes of memory.
MPC8641D processor at 1333MHz, Revision 2.2

2 Management Ethernet
12 TenGigE
40 GigabitEthernet
```

In the following example, the **utility head** command is entered with the **bytes** keyword. Only the first 15 bytes of output are displayed.

```
RP/0/RP0/CPU0:router# show version | utility head count 15 bytes

Cisco IOS XR S

RP/0/RP0/CPU0:router#
```

utility which

To locate a program file, use the **utility which** command in XR EXEC mode.

utility which {[[*WORD*] | [**all**] [**fullname**] [**long** [*link*]]] **program** *program-name* | **usage**}

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
all	(Optional) (**-a**) Displays all occurrences of the program specified by the **program** *pathname* keyword and argument.
fullname	(Optional) (**-f**) Displays the full pathname of the program file.
long [*link*]	(Optional) (**-l**) ("el") Displays the long format for each program found, and also displays link information if the file is a symlink.
program *program-name*	Specifies the name of the program file.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

None

Command Modes

Command History

Release	Modification
Release 7.0.12	No modification.

Usage Guidelines

Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID

Task ID	Operations
universal	execute

In the following example, the**utility which** command is entered without keywords or arguments to display the location of the perl program:

```
RP/0/RP0/CPU0:router# utility which program perl

  /pkg/bin/perl
```

In the following example, the **utility which** command is entered with the **fullname** keyword to display the full directory path of the perl program:

```
RP/0/RP0/CPU0:router# utility which fullname program perl
```

```
  /disk0:/-base-3.8.0.1I/sbin/perl
```

In the following example, the **utility which** command is entered with the **long** keyword to display additional details about the perl program file:

```
RP/0/RP0/CPU0:router# utility which long program perl

  -rwxrwxrwx  1     0         0        19245 Jul 28 14:31 /pkg/bin/perl
```

Related Commands

Command	Description
utility find, on page 501	Locates a file.

utility xargs

To run a program from one or more argument lists, use the **utility xargs** command in

mode.

utility xargs [[*WORD* | **trace**] [**program** [*program-name*] [*initial-arguments*]] | **usage**]

Syntax Description

WORD	(Optional) UNIX command-line option string. The maximum number of characters is 80.
trace	(Optional) (**-t**) Prints each program on standard error before executing.
program	(Optional) Specifies the name of the program and initial arguments. If a program name is not specified, then the echo utility is used.
program-name	(Optional) Specifies the name of the program. If a program name is not specified, then the echo utility is used.
initial-arguments	(Optional) Specifies the initial arguments.
usage	(Optional) Displays the UNIX options supported by this command.

Command Default

If no program is specified, then the echo utility is used (the input lines are displayed).

Command Modes

Command History

Release	Modification
Release 7.0.12	No modification.

Usage Guidelines



Note Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID

Task ID	Operations
universal	execute

In the following example, the **utility xargs** command is used to display the egress lines:

```
RP/0/RP0/CPU0:router# more disk0:/usr/files | utility xargs program grep EGRESS

  disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-101
  disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-102
  disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-103
```

```
disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-104
disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-105
disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-106
disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-107
disk0:/usr/atm.cfg:service-policy output EGRESS-GigabitEthernet0-0-0-0-108
disk0:/usr/atm.cfg:service-policy output EGRESS-common
disk0:/usr/atm.cfg:service-policy output EGRESS-common
disk0:/usr/atm.cfg:service-policy output EGRESS-common
disk0:/usr/atm.cfg:service-policy output EGRESS-common
disk0:/usr/atm.cfg:service-policy output EGRESS-common
.....
```



Zero Touch Provisioning Commands

- ztp breakout, on page 536
- ztp bootz, on page 537
- ztp bootz-server, on page 538
- ztp clean, on page 539
- ztp initiate, on page 540
- ztp terminate, on page 542

ztp breakout

Manual Zero Touch Provisioning (ZTP) invocation using the command-line interface (CLI) commands allows ZTP to run on more interfaces.

To invoke platform interface discovery before invoking DHCP, use the **ztp breakout** command in XR EXEC mode.

ztp breakout [debug] [verbose]{[nosignal-stay-in-breakout-mode] | **[nosignal-stay-in-state-noshut]** | **[hostname]** | **[apply** *configuration* **]}**

Syntax Description

debug	Run with additional logging to the console(cisco-support)
verbose	Run with logging to the console(cisco-support).
apply *configuration*	XR configuration commands to apply(cisco-support)
hostname	XR hostname to set(cisco-support)
nosignal-stay-in-breakout-mode	On no signal, prefer interfaces to remain in breakout mode(cisco-support)
nosignal-stay-in-state-noshut	On no signal, prefer interfaces to be noshut(cisco-support)

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **ztp breakout** command to perform a 10x10 breakout detection on all 100 Gigabit ports. On the 100G ports which are not able to bring up, the **ztp breakout** command will no shut all the 100G interfaces which support breakout configuration. If the interface is able to bring up, the **ztp breakout** script will keep the interface no change. Otherwise, the **ztp breakout** script will apply breakout configuration on the down interfaces.

The **nosignal-stay-in-breakout-mode** argument will force the port in breakout mode when all breakout interfaces from the same physical port have no signal locked, and place the ten 10Gigabitinterfaces in shutdown mode. The **nosignal-stay-in-state-noshut** argument will leave the port in breakout mode and place the ten 10Gigabit interfaces in no shutdown mode.

No progress logs are shown by default, although there will be XR syslogs for important events. If you wish to see more logs, add **verbose** to the **ztp terminate** command. If these logs are not enough, add **debug** before **verbose**.

Logs can be found in **disk0:/ztp/ztp.log**.

ztp bootz

To configure VRF for the Bootz operations, use the **ztp bootz** command in the XR Config mode.

ztp bootz vrf *vrf-name*
ztp bootz interface *interface-name*

Syntax Description

vrf *vrf-name*	Specifies the custom VRF on which the Bootz operations run.
interface *interface-name*	Specifies the interface that runs the Bootz operations.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 24.4.1	This command was introduced.

Usage Guidelines

Use this command if the interface used for the Bootz operations is configured under a new VRF. To run Bootz under the new VRF, you should configure the domain name servers and static routes, if any, under the same VRF to ensure reachability.

Examples

This example shows how to configure the **ztp bootz** command by providing the VRF name and interface name.

```
RP/0/RP0/CPU0:ios# config
RP/0/RP0/CPU0:ios(config)# ztp bootz
RP/0/RP0/CPU0:ios(config-bootz)# vrf mgmt
RP/0/RP0/CPU0:ios(config-bootz)# interface MgmtEth 0/RP0/CPU0/0
```

ztp bootz-server

To store the bootstrap server information that the router receives from the Bootz server during the initial boot process, use the **ztp bootz-server** command in XR Config mode.

When the standby control cards or line cards are inserted dynamically on the router, the ZTP-Bootz workflow uses this server information to communicate with the Bootz server and obtain the ownership vouchers for the standby control cards or line cards based on the serial number of the cards.

ztp bootz-server ip *ip-address* **port** *port* { **trust-anchor** *trust-anchor* }

Syntax Description

ip *ip-address*	Specifies the IPv4 or IPv6 address or hostname of the Bootz server.
port *port*	Specifies the port number of the Bootz server.
trust-anchor *trust-anchor*	Specifies the trust anchor certificate path for the Bootz server.

Command Default

No default behavior or values

Command Modes

XR Config mode

Command History

Release	Modification
Release 24.3.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Examples

This example shows how to configure the **ztp bootz-server** command by providing the trust anchor certificate path for the Bootz server.

```
RP/0/RP0/CPU0:ios(config)#ztp bootz-server ip 1.1.1.1 port 5000 trust-anchor
/misc/disk1/ta.cert
```

This example shows the stored server information.

```
RP/0/RP0/CPU0:ios#show running-config ztp
Thu Aug 29 12:35:07.013 IST
ztp
 bootz-server ip 1.1.1.1 port 5000 trust-anchor /misc/disk1/ta.cert action none
!
```

ztp clean

Manual Zero Touch Provisioning (ZTP) invocation using the command-line interface (CLI) commands allows ZTP to run on more interfaces.

To remove all Zero Touch Provisioning (ZTP) logs and settings saved on disk, use the **ztp clean** command in XR EXEC mode.

ztp clean [debug] [verbose]

Syntax Description

debug	Run with additional logging to the console(cisco-support).
verbose	Run with logging to the console(cisco-support)

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

If you wish to run ZTP as if from a clean boot, use the **ztp clean** command to remove all ZTP logs and settings. Use **commit replace** to reload, and then ZTP will run again as if from first boot.

No progress logs are shown by default, although there will be XR syslogs for important events. If you wish to see more logs, add **verbose** to the **ztp clean** command. If these logs are not enough, add **debug** before **verbose**.

Logs can be found in **disk0:/ztp/ztp.log**.

This example shows how to remove all ZTP files saved on disk:

```
RP/0/RP0/CPU0:router#ztp clean verbose
Mon Oct 10 17:03:43.581 UTC
Remove all ZTP temporary files and logs? [confirm] [y/n] :y
All ZTP files have been removed.
If you now wish ZTP to run again from boot, do 'conf t/commit replace' followed by reload.
```

ztp initiate

To invoke a new ZTP DHCP session, use the **ztp initiate** command in XR EXEC mode.

ztp initiate {[apply *configuration* **]** | **[dataport]** | **[dhcp4]** | **[dhcp6]** | **[dhcp4-client-identifier]** | **[dhcp6-client-identifier]** | **[dscp** *value* **]** | **[dscp6** *value* **]** | **[hostname]** | **[interface]** | **[management]** | **[noprompt]} [debug] [verbose]**

Syntax Description

debug	Run with additional logging to the console(cisco-support)
verbose	Run with logging to the console(cisco-support)
apply *configuration*	XR configuration commands to apply(cisco-support)
dataport	Send DHCP requests on all ADMIN UP physical LC interfaces.
dhcp4	Send only DHCP IPv4 requests(cisco-support)
dhcp6	Send only DHCP IPv6 requests(cisco-support)
dhcp4-client-identifier	Override default dhcp-client-identifier(cisco-support)
dhcp6-client-identifier	Override default dhcp6-client-id(cisco-support)
dscp *value*	DSCP/Prec Value(cisco-support)
dscp6 *value*	DSCP6/Prec Value(cisco-support)
hostname	XR hostname to set(cisco-support)
interface	Send DHCP requests only on the given interface(cisco-support)
management	Send DHCP requests on the platforms management interface(cisco-support)
noprompt	Run without prompting(cisco-support)

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Use the **ztp initiate** command to forceably inititate the ZTP, ignoring username configuration. **ztp initiate** allows the execution of a script even when the system has already been configured. This command is useful for testing ZTP without forcing a reload. This command is particularly useful to test scripts or if some manual operations are required before provisioning the box. **ztp initiate** can specify any data interfaces and management interface on the system to be used for the whole ZTP process. If you don't specify an interface, ztp will be invoked on management interface only.

No progress logs are shown by default, although there will be XR syslogs for important events. If you wish to see more logs, add **verbose** to the **ztp initiate** command. For more details, add **debug** before **verbose**.

Logs can be found in **disk0:/ztp/ztp.log**.

Example

This example shows how to bring up the interface manually:

```
RP/0/RP0/CPU0:router#ztp initiate debug verbose interface TenGigE 0/0/0/0
Invoke ZTP? (this may change your configuration) [confirm] [y/n] :
```

This example shows how to get rid of the prompting:

```
RP/0/RP0/CPU0:router#ztp initiate noprompt
Mon Jun 27 20:40:10.353 UTC
ZTP will now run in the background.
Please use "show logging" or look at /disk0:/ztp/ztp.log to check progress.
```

This example shows how to invoke the breakout discovery and ZTP, ZTP is invoked on the interfaces which are up:

```
RP/0/RP0/CPU0:router#ztp breakout debug verbose
RP/0/RP0/CPU0:router#ztp initiate dataport debug verbose
Invoke ZTP? (this may change your configuration) [confirm] [y/n] :
```

ztp terminate

To terminate all existing Zero Touch Provisioning (ZTP) processes, use the **ztp terminate** command in XR EXEC mode.

ztp terminate [debug] [verbose] [noprompt]

Syntax Description

debug	Run with additional logging to the console(cisco-support).
verbose	Run with logging to the console(cisco-support)
noprompt	Run without prompting(cisco-support)

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

If you want to terminate an already running ZTP process, use the **ztp terminate** command. Be careful when using the **ztp terminate** command. Improper usage of this command may leave your system in a partially configured state.

No progress logs are shown by default, although there will be XR syslogs for important events. If you wish to see more logs, add **verbose** to the **ztp terminate** command. If these logs are not enough, add **debug** before **verbose**.

Logs can be found in **disk0:/ztp/ztp.log**.

Note If the interface IP configuration in Linux in the XR namespace is not cleaned up properly, the same IP may be present in the global Virtual Routing and Forwarding (VRF) and XR namespace at the same time.

Example

This example shows how to terminate the ZTP sessions in progress:

```
RP/0/RP0/CPU0:router#ztp terminate verbose
Mon Oct 10 16:52:38.507 UTC
Terminate ZTP? (this may leave your system in a partially configured state) [confirm] [y/n]
 :y
ZTP terminated
```



INDEX

A

abort command 79
action command 234
active command 22
alert-group disable command 23
alias command 80
apply-template command 83
autocommand command 457

B

banner exec command 459
banner login command 461
banner motd command 463

C

call-home command 24
call-home request command 25
call-home send alert-group inventory command 28
call-home send command 27
call-home test command 29
cdp command 58
cdp holdtime command 59
cdp log adjacency changes command 60
cdp timer command 61
clear cdp counters command 62
clear cdp table command 63
clear comment command 84
clear configuration commits command 85
clear configuration inconsistency command 87
clear configuration inconsistency replica command 89
clear configuration sessions command 90
clear context command 250
clear line command 465
clear line vty command 466
clear snmp counters command 292
cli interactive syntax check command 489
cli whitespace completion command 490
commit command 91
configuration commit auto-save command 95
configure command 97
contact-email-addr command 30
contract-id command 32
customer-id command 33

D

delay command 235
description (interface) command 99
destination address command 34
destination message-size-limit command 35
destination preferred-msg-format command 36
destination transport-method command 37
disconnect-character command 467
do command 100
dumpcore command 251

E

end command 101
end-group command 103
end-template command 104
escape-character command 468
exception disk-usage-limit command 254
exception filepath command 255
exec-timeout command 469
exit command 106

F

follow command 259

H

hostname command 108

I

index persistence command 293
install commit command 436
interface (track) command 236

L

length command 470
line command 471
line-protocol track command 237

load command 109
load commit changes command 111
load configuration failed command 112
load configuration removed command 113
load rollback changes command 114

M

mail-server 38
man command 115
more command 118

N

notification linkupdown command 294

O

object command 238

P

phone-number command 39
process command 266
process core command 268
process mandatory command 270
profile command 40
pwd (config) command 122

R

rate-limit command 41
rollback configuration command 123
root command 126
route ipv4 command 239

S

save configuration changes command 129
save configuration command 128
save configuration commit changes command 130
save configuration failed command 131
save configuration merge command 132
save configuration removed command 133
save rollback changes command 134
send command 472
sender command 42
service cli commit-optimized enable 135
session-timeout command 473
set default-afi command 136
set default-safi command 137
set default-vrf command 138
show aliases command 143
show bcdl command 2
show bcdl consumers command 4
show bcdl queues command 6
show bcdl tables command 7
show bcdl trace command 9
show call-home alert-group command 45
show call-home command 43
show call-home mail-server status command 46
show call-home profile command 47
show call-home statistics command 48
show call-home trace command 50
show cdp command 65
show cdp entry command 67
show cdp interface command 69
show cdp neighbors command 71
show cdp traffic command 74
show command 140
show configuration (config) command 156
show configuration changes command 148
show configuration commit changes command 150
show configuration commit list command 154
show configuration failed incompatible command 158
show configuration failed remove command 159
show configuration failed rollback command 161
show configuration failed startup command 162
show configuration history command 163
show configuration inconsistency replica command 166
show configuration persistent command 167
show configuration removed command 169
show configuration rollback changes command 170
show configuration running command 172
show configuration running-config command 173
show configuration sessions command 175
show context command 272
show default-afi-safi-vrf command 177
show epm trace boot command 14
show history command 178
show install active command 437
show install log command 449
show install request command 450
show line command 474
show memory command 274
show memory compare command 277
show memory heap command 280
show processes command 284
show reboot command 16
show running-configuration filter vrf command 145
show snmp command 296
show snmp context command 299
show snmp context-mapping command 300
show snmp engineid command 301
show snmp group command 302
show snmp host command 304
show snmp informs details 306
show snmp interface command 308
show snmp interface notification command 310
show snmp mib command 313
show snmp mib ifmib cache 316

show snmp mib ifmib general 318
show snmp mib ifmib statsd 320
show snmp request incoming-queue detail 325
show snmp request overload stats 327
show snmp statistics oid group 330
show snmp statistics pdu 332
show snmp statistics poll oid all 334
show snmp statistics poll oid nms 336
show snmp statistics slow oid 337, 339
show snmp traps details 341
show snmp users command 343
show snmp view command 345
show terminal command 477
show track command 240
show users command 480
show xml schema command 210
site-id command 52
snmp request drop summary 322
snmp request type summary 329
snmp snmp request type detail 328
snmp-server chassis-id command 347
snmp-server community command 348
snmp-server community-map command 350
snmp-server contact command 352
snmp-server context command 353
snmp-server engineid command 358
snmp-server engineid remote command 359
snmp-server group command 361
snmp-server host command 364
snmp-server ifindex persist command 368
snmp-server ifmib stats cache command 370
snmp-server inform command 371
snmp-server interface command 372
snmp-server interface subset command 374
snmp-server ipv4 dscp command 376
snmp-server ipv4 precedence command 377
snmp-server location command 379
snmp-server notification-log-mib command 390
snmp-server packetsize command 392
snmp-server queue-length command 393
snmp-server target list command 395
snmp-server trap authentication vrf disable command 346
snmp-server trap selective-vrf-download role-change command 422
snmp-server trap throttle-time command 396
snmp-server trap-timeout command 397
snmp-server traps bgp command 404
snmp-server traps command 398
snmp-server traps mpls l3vpn command 407
snmp-server traps ospf errors command 408
snmp-server traps ospf lsa command 410
snmp-server traps ospf retransmit command 411
snmp-server traps ospf state-change command 412
snmp-server traps ospfv3 errors command 414
snmp-server traps ospfv3 state-change command 415
snmp-server traps pim interface-state-change command 417
snmp-server traps pim invalid-message-received command 418
snmp-server traps pim neighbor-change command 419
snmp-server traps pim rp-mapping-change command 420
snmp-server traps rsvp command 421
snmp-server traps snmp command 423
snmp-server traps syslog command 426
snmp-server user command 427
snmp-server view command 430
snmp-server vrf command 432
ssh server command 231
streaming command 211
street-address command 53
subscribe-to-alert-group inventory command 54
subscribe-to-alert-group syslog command 55

T

template command 180
terminal length command 481
terminal width command 482
timestamp disable command 483
track command 241
transport input command 484
transport output command 485
transport preferred command 486
type line-protocol state command 242
type list boolean command 243
type route reachability command 245
type rtr command 246

U

utility bc command 515
utility df command 519
utility du command 523
utility egrep command 526
utility head command 529
utility which command 531
utility xargs command 533

V

vrf (track) command 247

X

xml agent command 213
xml agent tty command 215