Hewlett Packard Enterprise

HPE VSR1000_HPE-CMW710-E0324-X64 Release Notes

Software version:VSR1000_HPE-CMW710-E0324-X64

Contents

List of Tables

This document describes the features, restrictions and guidelines, open problems, and workarounds for version *VSR1000_HPE-CMW710-E0324-X64*. Before you use this version in a live network, back up the configuration and test the version to avoid software upgrade affecting your live network.

Use this document in conjunction with *HPE VSR1000_HPE-CMW710-E0324-X64 Release Notes (Software Feature Changes)* and the documents listed in "Related documents"

Version information

Version number

HPE Comware Software, Version 7.1.059, ESS 0324

Versionhistory

Table 1 Version history

Version number	Last version	Release date	Release type	Remarks
VSR1000_HPE-CMW710-E0324-X64	VSR1000_HPE-CMW710-E0322P01-X64	2016-10-20	ESS version	Fixedbugs
VSR1000_HPE-CMW710-E0322P01-X64	VSR1000_HPE-CMW710-E0322-X64	2016-04-27	ESS version	New features: • Setting a local QoS ID match criterion in policy-based routing Fixedbugs
VSR1000_HPE-CMW710-E0322-X64	VSR1000_HP-CMW710-E0321P01-X64	2016-01-29	ESS version	Modified feature: • Domain • BGP Fixedbugs
VSR1000_HPE-CMW710-E0321P01-X64	VSR1000_HP-CMW710-E0321-X64	2015-12-5	ESS version	New features: • IKEv2
VSR1000_HP-CMW710-E0321-X64	VSR1000_HP-CMW710-E0301-X64	2015-08-26	ESS version	New features: • IRF&ISSU • Reth Interface • AFT • Monitor Link • Service chain • L7 LB • Outbound Link LB • WAAS • VXLAN • ADVPN • LLDP • OpenFlow
VSR1000_HP-CMW710-E0301-X64	VSR1000_HP-CMW710-R0202-X64	2014-06-12	ESS version	New features: • OVF with start-up configuration

				deployment • EVI • Multi-control plane • MTR • Zone-based security management
VSR1000_HP-CMW710-R0202-X64	VSR1000_HP-CMW710-E0102-X64	2014-04-11	Release version	New features: • VirtIO Disk • VMware ESXi 5.5 • Intel 82599 VF
VSR1000_HP-CMW710-E0102-X64	VSR1000_HP-CMW710-E0101P01-X64	2013-12-19	ESS version	None
VSR1000_HP-CMW710-E0101P01-X64	First release	2013-09-18	ESS version	None

Software operating environments

Table 2 Software operating environments

Hypervisor(s)	Virtual Machine requirements		
VMware/KVM	VSR1001	VSR1004	VSR1008
	1 vCPU (2.0 GHz or higher)	4vCPUs (2.0 GHz or higher)	8vCPUs (2.0 GHz or higher)
	1 GB RAM or more	2 GB RAM or more	4 GB RAM or more
	Disk space: 8GB Network interfaces: 2 or more virtual Network Interface Cards (vNICs)		

Table 3 Software Licenses

J#	Description
JG810AAE	HP VSR1001 60 Day Eval SW and E-LTU
JG811AAE	HP VSR1001 Virtual Services Router E-LTU
JG812AAE	HP VSR1004 Virtual Services Router E-LTU
JG813AAE 5012-3667	HP VSR1008 Virtual Services Router E-LTU HP VSR1001 Virtual Services Router 90 Day Evaluation License
5012-3668	HP VSR1004 Virtual Services Router 90 Day Evaluation License
5012-3669	HP VSR1008 Virtual Services Router 90 Day Evaluation License

Hardware and software compatibility matrix

CAUTION:

To avoid an upgrade failure, use Table 4to verify the hardware and software compatibility before performing an upgrade.

Table 4 Hardware and software compatibility matrix

Item	Specifications
Product series	VSR1000
Hypervisor	VMware ESXi 4.1, 5.0, 5.1, 5.5 Linux KVM (Linux Kernel > 2.6.25).The recommended Linux distributions include: • CentOS 7.0 • Ubuntu 14.04 • RedHat Enterprise Linux (RHEL) 6.3 • SUSE Linux Enterprise Server 11SP2
Virtual Network Interface	E1000 VirtIO VMXNET3 Intel 82599VF
Virtual Disk	IDE (VMware ESXi, Linux KVM) SCSI - LSI Logic Parallel (VMware ESXi) VirtIO (Linux KVM)
Boot ROM version	Version 1.08 or higher
Host software	VSR1000_HPE-CMW710-E0324-X64.ipe VSR1000_HPE-CMW710-E0324-X64.iso VSR1000_HPE-CMW710-E0324-X64.ova VSR1000_HPE-CMW710-E0324-X64.qco VSR1000_HPE-CMW710-E0324-X64.descriptor ToMarketToolsV1.0.zip **NOTE:** • The *VSR1000_HPE-CMW710-E0324-X64.ova* is only available on VMware platform. • The *VSR1000_HPE-CMW710-E0324-X64.ova* is based on **Virtual machine version 8** which is compatible with ESXi 5.0 and laterhosts. • The *VSR1000_HPE-CMW710-E0324-X64.qco* is only available on KVM platform. • ToMarketToolsV1.0.zip is used to create scripts for installing linux bridge, OVS, and VF.

Item	Specifications
iMC version	iMC BIMS 7.2(E0402) iMC EAD 7.2(E0406) iMC EIA 7.2(E0406) iMC IVM 7.2(E0402H02) iMC MVM 7.2(E0402) iMC NTA7.2(E0401P04) iMC PLAT7.2(E0403L09) iMC QoSM 7.2(E0403) iMC RAM 7.2(E0402) iMC SHM 7.2(E0402) iMC UBA 7.2(E0401P04)
iNode version	iNode PC 7.2(E0411)

Upgrading restrictions and guidelines

None

Hardware feature updates

None

Software feature and command updates

For more information about the software feature and command update history, see *HP VSR1000_HPE-CMW710-E0324-X64 Release Notes (Software Feature Changes)*.

MIB updates

Table 5 MIB updates

Item	MIB file	Module	Description
VSR1000_HPE-CMW710-E0324-X64			
New	None		
Modified	None		
VSR1000_HPE-CMW710-E0322P01-X64			
New	None		
Modified	None		
VSR1000_HPE-CMW710-E0322-X64			
New	None		
Modified	None		
VSR1000_HPE-CMW710-E0321P01-X64			
New	hh3c-nat.mib	HH3C-NAT-MIB	hh3NATStatVer2Table is

			supported
Modified	None		
VSR1000_HP-CMW710-E0321-X64			
New	hh3c-tunnel.mib	HH3C-TUNNEL-MIB	
New	rfc5643-ospfv3.mib	OSPFV3-MIB	
New	rfc2981-disman-event.mib	DISMAN-EVENT-MIB	
New	lldp-ext-dot1.mib	LLDP-EXT-DOT1-MIB	
New	lldp-ext-dot1-v2.mib	LLDP-EXT-DOT1-V2-MIB	
New	lldp-ext-dot3.mib	LLDP-EXT-DOT3-MIB	
New	lldp-ext-dot3-v2.mib	LLDP-EXT-DOT3-V2-MIB	
New	lldp-ext-med.mib	LLDP-EXT-MED-MIB	
New	lldp.mib	LLDP-MIB	
New	lldp-v2.mib	LLDP-V2-MIB	
New	hh3c-vxlan.mib	HH3C-VXLAN-MIB	
New	hh3c-nvgre.mib	HH3C-NVGRE-MIB	
New	hh3c-vsi.mib	HH3C-VSI-MIB	
Modified	hh3c-trap.mib	HH3C-TRAP-MIB	hh3cTrapConfigSwitch2 has been added to hh3cTrapConfigTable.
Modified	hh3c-sys-man.mib	HH3C-SYS-MAN-MIB	The PDS of hh3cSysLocalClock and hh3cSysLocalClockString has been modified.
Modified	hh3c-dhcp4.mib	HH3C-DHCP4-MIB	hh3cDhcpServer2PoolTable has been modified.
Modified	hh3c-cbqos2.mib	HH3C-CBQOS2-MIB	The description of hh3cCBQoSQueueLengthh as been modified.
Modified	hh3c-acl.mib	HH3C-ACL-MIB	hh3cAclNumberGroupTable, hh3cPfilterApplyTable, hh3cPfilterAclGroupRunInfoTable, hh3cPfilterStatisticSumTable and added the hh3cAclNamedGroupTable, hh3cAclIPAclNamedBscTable, hh3cAclIPAclNamedAdvTable, hh3cAclNamedMACTable, hh3cAclIntervalTable hh3cAclNamedUserTable, hh3cPfilter2ApplyTable, hh3cPfilter2, hh3cPfilter2AclGroupRunInfoTable, hh3cPfilter2AclRuleRunInfoTable, hh3cPfilter2StatisticSumTable

Modified	rfc4292-ip-forward.mib	IP-FORWARD-MIB	Update from RFC2096 to RFC4292
Modified	rfc2819-rmon.mib	RMON-MIB	The default description has been modified.
	rfc4502-rmon.mib	RMON2-MIB	The default description has been modified.
	hh3c-common-system.mib	HH3C-COMMON-SYSTEM-MIB	Update to V2.7
	hh3c-config-man.mib	HH3C-CONFIG-MAN-MIB	Update to V2.6
	hh3c-flash-man.mib	HH3C-FLASH-MAN-MIB	Update to V3.3
	hh3c-snmp-ext.mib	HH3C-SNMP-EXT-MIB	Update to V1.5
Modified	rfc1213.mib	RFC1213-MIB	The description of ipForwarding and ipDefaultTTLhas been modified.
Modified	hh3c-snmp-ext.mib	HH3C-SNMP-EXT-MIB	hh3cSnmpExtTrapSource and hh3cSnmpExtInformSource are supported.
Modified	hh3c-dhcp4.mib	HH3C-DHCP4-MIB	hh3cDhcpServer2DefOptGrpTable, hh3cDhcpServer2ValidClassTable, hh3cDhcpServer2RuleHwAddrTable, hh3cDhcpServer2OptionGroupTable, and hh3cDhcpServer2OptionTable are supported
Modified	rfc2925-disman-ping.mib	DISMAN-PING-MIB	hh3cNqajitter (for ICMP-jitter test) is supported. The description of hh3cNqaCtlCodecType has been modified.
VSR1000_HP-CMW710-E0301-X64			
New	rfc4560-disman-traceroute.mib	DISMAN-TRACEROUTE-MIB	
New	rfc5519-mgmd-std.mib	MGMD-STD-MIB	
New	rfc6445-mpls-frr-facility-std.mib	MPLS-FRR-FACILITY-STD-MIB	
New	rfc6445-mpls-frr-general-std.mib	MPLS-FRR-GENERAL-STD-MIB	
New	rfc1473-ppp-ip.mib	PPP-IP-NCP-MIB	
New	rfc1471-ppp-lcp.mib	PPP-LCP-MIB	
New	rfc5603-pw-enet-std.mib	PW-ENET-STD-MIB	
New	rfc5602-pw-mpls-std.mib	PW-MPLS-STD-MIB	
New	rfc5601-pw-std.mib	PW-STD-MIB	
New	hh3c-bfd-std.mib	HH3C-BFD-STD-MIB	

New	hh3c-ipran-dcn.mib	HH3C-IPRAN-DCN-MIB	
New	hh3c-lbv2.mib	HH3C-LBV2-MIB	
New	hh3c-license.mib	HH3C-LICENSE-MIB	
New	hh3c-mplsext.mib	HH3C-MPLSEXT-MIB	
New	hh3c-mplste.mib	HH3C-MPLSTE-MIB	
New	hh3c-snmp-ext.mib	HH3C-SNMP-EXT-MIB	
New	hh3c-entity-ext.mib	HH3C-ENTITY-EXT-MIB	
New	hh3c-ssh.mib	HH3C-SSH-MIB	
New	rfc3970-te.mib	TE-MIB	
New	hh3c-nqa.mib	HH3C-NQA-MIB	
New	rfc6445-mpls-frr-facility-std.mib	MPLS-FRR-FACILITY-STD-MIB	
New	rfc3970-te.mib	TE-MIB	
New	rfc4560-disman-traceroute.mib	DISMAN-TRACEROUTE-MIB	
New	rfc2925-disman-ping.mib	DISMAN-PING-MIB	
New	hh3c-bfd-std.mib	HH3C-BFD-STD-MIB	
New	rfc5601-pw-std.mib	PW-STD-MIB	
New	hh3c-trap.mib	HH3C-TRAP-MIB	
New	hh3c-license.mib	HH3C-LICENSE-MIB	
New	hh3c-lbv2.mib	HH3C-LBV2-MIB	
Modified	hh3c-ntp.mib	HH3C-NTP-MIB	The description of hh3cNTPSystemMIB has been modified.
Modified	rfc1213.mib	RFC1213-MIB	The description of OBJECT sysName has been modified.
VSR1000_HP-CMW710-R0202-X64			
New	None		
Modified	rfc4133-entity.mib	ENTITY-MIB	The description of OBJECT entPhysicalSerialNum has been modified.
Modified	hh3c-config-man.mib	HH3C-CONFIG-MAN-MIB	The description of OBJECTs hh3cCfgOperateSrvAddrType, hh3cCfgOperateSrvAddrRev and hh3cCfgOperateSrvVPNName has been modified.
Modified	rfc2925-disman-ping.mib	DISMAN-PING-MIB	The description of TABLE pingCtlTable has been modified.
VSR1000_HP-CMW710-E0102-X64			
New	rfc4502-rmon.mib	RMON2-MIB	RMON2-MIB is supported

Modified	rfc4133-entity.mib	ENTITY-MIB	OBJECT entPhysicalSerialNum is supported.
VSR1000_HP-CMW710-E0101P01-X64			
New	First release		
Modified	First release		

Operation changes

Operation changes in VSR1000_HPE-CMW710-E0324-X64

None.

Operation changes in VSR1000_HPE-CMW710-E0322P01-X64

None.

Operation changes in VSR1000_HPE-CMW710-E0322-X64

None.

Operation changes in VSR1000_HPE-CMW710-E0321P01-X64

None.

Operation changes in VSR1000_HP-CMW710-E0321-X64

None.

Operation changes in VSR1000_HP-CMW710-E0301-X64

None.

Operation changes in VSR1000_HP-CMW710-R0202-X64

None.

Operation changes in VSR1000_HP-CMW710-E0102-X64

VSR Device ID

The file name of **Device ID** has been changed from ‘DeviceID.DID’ to ‘*SN*.DID’. The *SN* is the Serial Number of your device, for example:

```
<HP>display license device-id
SN: 5254003c4f48910b13121307
Device ID: flash:/license/5254003c4f48910b13121307.did
```

Operation changes in VSR1000_HP-CMW710-E0101P01-X64

First release.

Restrictions and cautions

VSR License

Without a valid license, VSR performancewill be verylimited. To get full performance, please purchase and install a valid license. A free 60-day evaluation license is available from the HP website (http://hpe.com/networking/mynetworking/). Please see the HP VSR1000 Virtual Services Router Installation and Getting Started Guide for full details. See "Support and other resources".

Open problems and workarounds

201507240253

- Symptom: After session synchronization for stateful failover is enabled, only new sessions are synchronized. Existing sessions cannot be synchronized.
- Condition: This symptom might occur if session synchronization for stateful failover is enabled after the virtual router has been running for a period of time.
- Workaround: Enable session synchronization for stateful failover after the virtual routerstarts.

201505090048

- Symptom: After a physical server reboots because of power failure, KVM cannot start when the KVM-based VSR loads the existing .qco file.
- Condition: This symptom might occur if the .qco file is corrupted because of power failure.
- Workaround: Install a UPS for the physical server to avoid reboots caused by power failure.

List of resolved problems

Resolved problems in VSR1000_HPE-CMW710-E0324-X64

201607040237

- Symptom: CVE-2016-4953
- Condition: Fixed vulnerability in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service by sending a spoofed packet with incorrect authentication data at a certain time.
- Symptom: CVE-2016-4954
- Condition: Fixed vulnerability in ntpd in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service by sending spoofed packets from source IP addresses in a certain scenario.
- Symptom: CVE-2016-4956
- Condition: Fixed vulnerability in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service via a spoofed broadcast packet.

201606060158

- Symptom: CVE-2016-2105
- Condition: Fixed vulnerability in “EVP Encode” in OpenSSL before 1.0.1t and 1.0.2 before 1.0.2h allows remote attackers to cause a denial of service (heap memory corruption) via a large amount of binary data.
- Symptom: CVE-2016-2106
- Condition: Fixed vulnerability in “EVP Encrypt” in OpenSSL before 1.0.1t and 1.0.2 before 1.0.2h allows remote attackers to cause a denial of service (heap memory corruption) via a large amount of binary data.
- Symptom: CVE-2016-2107
- Condition: Fixed vulnerability in OpenSSL before 1.0.1t and 1.02h allows remote attackers to obtain sensitive cleartext information via a padding-oracle attack against an AES CBC session.
- Symptom: CVE-2016-2108
- Condition: Fixed vulnerability in OpenSSL before 1.0.1o and 1.0.2 before 1.0.2c allows remote attackers to execute arbitrary code or cause a denial of service (buffer underflow and memory corruption).
- Symptom: CVE-2016-2109
- Condition: Fixed vulnerability in “asn” before 1.0.1t and 1.0.2 before 1.0.2h allows remote attackers to cause a denial of service (memory consumption) via a short invalid encoding.
- Symptom: CVE-2016-2176
- Condition: Fixed vulnerability in “X509” in OpenSSL before 1.0.1t and 1.0.2 before 1.0.2h allows remote attackers to obtain sensitive information from memory or cause a denial of service

201605160353

- Symptom: CVE-2016-1547
- Condition: Fixed vulnerability where an off-path attacker can deny service to ntpd clients by demobilizing preemptable associations using spoofed crypto-NAK packets.
- Symptom: CVE-2016-1548
- Condition: Fixed vulnerability where an attacker can change the time of an ntpd client or deny service to an ntpd client by forcing it to change from basic client/server mode to interleaved symmetric mode.
- Symptom: CVE-2016-1550

- Condition: Fixed vulnerability in ntpd function allow an attacker to conduct a timing attack to compute the value of the valid authentication digest causing forged packets to be accepted by ntpd.
- Symptom: CVE-2016-1551
- Condition: Fixed vulnerability in ntpd allows unauthenticated network attackers to spoof refclock packets to ntpd processes on systems that do not implement bogon filtering.
- Symptom: CVE-2016-2518
- Condition: Fixed vulnerability when using a crafted packet to create a peer association lookup to make an out-of-bounds reference.
- Symptom: CVE-2016-2519
- Condition: Fixed vulnerability in ntpd will abort if an attempt is made to read an oversized value.
- Symptom: CVE-2015-7704
- Condition: Fixed vulnerability in ntpd that a remote attacker could use, to send a packet to an ntpd client that would increase the client's polling interval value, and effectively disable synchronization with the server.

201605060563

- Symptom: CVE-2015-8158
- Condition: Fixed vulnerability in ntpq allows attackers to cause MITM attack or a denial of service.
- Symptom: CVE-2015-8138
- Condition: Fixed vulnerability in ntpd which attackers may be able to disable time synchronization by sending a crafted NTP packet to the NTP client.
- Symptom: CVE-2015-7979
- Condition: Fixed vulnerability in ntpd allows attackers to send special crafted broadcast packets to broadcast clients, which may cause the affected NTP clients to become out of sync over a longer period of time.
- Symptom: CVE-2015-7975
- Condition: Fixed vulnerability in ntpq allows attackers to crash an ntpq client instance.
- Symptom: CVE-2015-7974
- Condition: Fixed vulnerability in NTP 4.x before 4.2.8p6 and 4.3.x before 4.3.90 which might allow remote attackers to conduct impersonation attacks via an arbitrary trusted key.
- Symptom: CVE-2015-7973
- Condition: Fixed vulnerability when NTP is configured in broadcast mode, a man-in-the-middle attacker or a malicious client could replay packets received from the broadcast server to all (other) clients, which cause the time on affected clients to become out of sync over a longer period of time.

20160090347

- Symptom: CVE-2009-3238
- Condition: The get_random_int function in the Linux kernel before 2.6.30 produces insufficiently random numbers, which allows attackers to predict the return value, and possibly defeat protection mechanisms.

201607290008

- Symptom: CVE-2012-0036
- Condition: Fixed vulnerability in curl and libcurl 7.2x before 7.24.0 that allows remote attackers to conduct data-injection attacks via a crafted URL, as demonstrated by a CRLF injection attack on the (1) IMAP, (2) POP3, or (3) SMTP protocol.

201607290225

- Symptom: CVE-2016-4953

- Condition: Fixed vulnerability in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service by sending a spoofed packet with incorrect authentication data at a certain time.
- Symptom: CVE-2016-4954
- Condition: Fixed vulnerability in ntpd in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service by sending spoofed packets from source IP addresses in a certain scenario.
- Symptom: CVE-2016-4956
- Condition: Fixed vulnerability in NTP 4.x before 4.2.8p8 allows remote attackers to cause a denial of service via a spoofed broadcast packet.

201610280087

- Symptom: CVE-2016-2177
- Condition: Fixed vulnerability in s3_srvr.c, ssl_sess.c, and t1_lib.c functions in OpenSSL through 1.0.2h that allows remote attackers to cause a denial of service (integer overflow and application crash), or possibly have an unspecified other impact by leveraging unexpected malloc behavior.

Resolved problems in VSR1000_HPE-CMW710-E0322P01-X64

201602010067

- Symptom: The route filtering feature takes effect, but the configuration cannot be displayed by using the **display current configuration** command.
- Condition: This symptom occurs if the following conditions exist:
  - The **filter-policy** command is configured in RIP view.
  - The configuration is saved and the device is rebooted.
  - A .cfg configuration file is used to recover the configuration.

201604120324

- Symptom: The configuration of CAR, GTS, or a QoS policy with CAR or GTS on a VXLAN interface does not take effect.
- Condition:This symptom occurs if the VXLAN interface is configured with CAR or GTS or has a QoS policy with CAR or GTS applied.

201603170336

- Symptom: The device cannot reboot correctly after the file system is damaged.
- Condition: This symptom occurs if the following conditions exist:
  - A VSR image is created by using an OVF template.
  - The system is started up by using the VSR image.
  - The device reboots unexpectedly.
  - The file system is damaged.

201603180143

- Symptom 1: CVE-2015-3194
- Condition: Fixed vulnerability which can be exploited in a DoS attack, if device is presented with a specific ASN.1 signature using the RSA.
- Symptom 2: CVE-2015-3195

- Condition: Fixed vulnerability with malformed OpenSSL X509_ATTRIBUTE structure used by the PKCS#7 and CMS routines which may cause memory leak.
- Symptom 3: CVE-2015-3196
- Condition: Fixed vulnerability where a race condition can occur when specific PSK identity hints are received.
- Symptom 4: CVE-2015-1794
- Condition: Fixed vulnerability if a client receives a ServerKeyExchange for an anonymous Diffie-Hellman (DH) ciphersuite which can cause possible Denial of Service (DoS) attack.

201603250389

- Symptom 1: CVE-2016-0701
- Condition: Fixed vulnerability in the DH_check_pub_key function which makes it easier for remote attackers to discover a private DH (Diffie-Hellman) exponent by making multiple handshakes with a peer that chose an inappropriate number. This issue affects OpenSSL version 1.0.2. and addressed in 1.0.2f. OpenSSL 1.0.1 is not affected by this CVE.
- Symptom 2: CVE-2015-3197
- Condition: Fixed vulnerability when using SSLv2 which can be exploited in a man-in-the-middle attack, if device has disabled ciphers.

201603220289

- Symptom1: CVE-2016-0705
- Condition: Fixed vulnerability when OpenSSL parses malformed DSA private keys and could lead to a DoS attack or memory corruption for applications that receive DSA private keys from untrusted sources.
- Symptom2: CVE-2016-0798
- Condition: Fixed vulnerability in OpenSSL 1.0.1 before 1.0.1s and 1.0.2 before 1.0.2g allows remote attackers to cause a denial of service (memory consumption) by providing an invalid username in a connection attempt.
- Symptom 3: CVE-2016-0797
- Condition: Fixed vulnerability in OpenSSL 1.0.1 before 1.0.1s and 1.0.2 before 1.0.2g allow remote attackers to cause a denial of service (heap memory corruption or NULL pointer dereference).
- Symptom 4: CVE-2016-0799
- Condition: Fixed vulnerability in OpenSSL 1.0.1 before 1.0.1s and 1.0.2 before 1.0.2g improperly calculates string lengths, which allows remote attackers to cause a denial of service which could lead to memory allocation failure or memory leaks.
- Symptom 5: CVE-2016-0702
- Condition: Fixed vulnerability in OpenSSL 1.0.1 before 1.0.1s and 1.0.2 before 1.0.2g which makes it easier for local users to discover RSA keys leveraging cache-bank conflicts, aka a "CacheBleed" attack.

201604110147

- Symptom: CVE-2016-2842
- Condition: Fixed vulnerability in the doapr_outch function in crypto/bio/b_print.c, which allows remote attackers to cause a denial of service (out-of-bounds write or memory consumption) or possibly have unspecified other impact via a long string.

Resolved problems in VSR1000_HPE-CMW710-E0322-X64

201512180265

- Symptom: When VXLANis running, the system encapsulates the VXLAN packets with 0 as the UDP source port. But in some environment, some devices such as GGSN in mobile scenariomight drop VXLAN packets which use 0 as source UDP port.
- Condition: This symptom might occur if some third party devices drop VXLAN packets which use 0 as source UDP port.

201512030079

- Symptom: When SNMP is used to get interface counters on a VXLAN tunnel interface, the counters are always 0.
- Condition: This symptom must occur when SNMP is used to get interface counters on a VXLAN tunnel interface.

201512050025

- Symptom 1: CVE-2015-7871
- Condition: Cause ntpd to accept time from unauthenticated peers.
- Symptom 2: CVE-2015-7704
- Condition: An ntpd client forged by a DDoS attacker located anywhere on the Internet, that can exploit NTP's to disable NTP at a victim client or it may also trigger a firewall block for packets from the target machine.
- Symptom 3: CVE-2015-7705
- Condition: The DDoS attacker can send a device a high volume of ntpd queries that are spoofed to look like they come from the client. The servers then start rate-limiting the client.
- Symptom 4: CVE-2015-7855
- Condition: Ntpd mode 6 or mode 7 packet containing an unusually long data value could possibly use cause NTP to crash, resulting in a denial of service.

Resolved problems in VSR1000_HPE-CMW710-E0321P01-X64

201507180147

- Symptom 1: CVE-2014-8176
- Condition: If a DTLS peer receives application data between the ChangeCipherSpec and Finished messages. May result in a segmentation fault or potentially, memory corruption.
- Symptom2:CVE-2015-1788
- Condition 3: When processing an ECParameters structure OpenSSL enters an infinite loop. This can be used to perform denial of service against any system which processes public keys, certificate requests or certificates.
- Symptom 4: CVE-2015-1789
- Condition: X509_cmp_time does not properly check the length of the ASN1_TIME string and/or accepts an arbitrary number of fractional seconds in the time string. An attacker can use this to craft malformed certificates and CRLs of various sizes and potentially cause a segmentation fault, resulting in a DoS on applications that verify certificates or CRLs.

- Symptom 5: CVE-2015-1790
- Condition: The PKCS#7 parsing code does not handle missing inner EncryptedContent correctly. An attacker can craft malformed PKCS#7 blobs with missing content and trigger a NULL pointer dereference on parsing.
- Symptom 6: CVE-2015-1791
- Condition: If a NewSessionTicket is received by a multi-threaded client when attempting to reuse a previous ticket then a race condition can occur potentially leading to a double free of the ticket data.
- Symptom 7: CVE-2015-1792
- Condition: When verifying a signedData message the CMS code can enter an infinite loop. This can be used to perform denial of service against any system which verifies signedData messages using the CMS code.

201510200471

- Symptom: The system uptime becomes incorrect if the virtual router has been running for seven months. As a result, time-based features cannot operate correctly. For example, neighbors cannot learn the routes that the virtual router advertises through OSPF.
- Condition: This symptom might occur if the virtual router has been running for seven months and the system uptime becomes incorrect.

201511040524

- Symptom: An interface cannot communicate with other devices on the same network segment if a 31-bit subnet mask is assigned to the interface.
- Condition: This symptom might occur if a 31-bit subnet mask is assigned to the interface, and the interface determines that the IP addresses of the other devices are invalid.

201512020081

- Symptom: In an environment where HOS 2.0, OVS, and DPDK are used together, the forwarding rate of the VSR1001 virtual router (1 vCPU) drops unexpectedly if traffic is sent to the virtual router at a high rate (for example, 50 kpps).
- Condition: This symptom might occur if the VSR1001 virtual router is in an environment where HOS 2.0, OVS, and DPDK are used together, and traffic is sent to the virtual router at a high rate.

Resolved problems in VSR1000_HP-CMW710-E0321-X64

201412310432

- Symptom: CVE-2014-9295
- Condition: Stack-based buffer overflows in ntpd in NTP before 4.2.8 allow remote attackers to execute arbitrary code via a crafted packet.

201501070148

- Symptom 1: CVE-2014-3567
- Condition: When an OpenSSL SSL/TLS/DTLS server receives a session ticket the integrity of that ticket is first verified. In the event of a session ticket integrity check failing, OpenSSL will fail to free memory causing a memory leak. By sending a large number of invalid session tickets an attacker could exploit this issue in a Denial of Service attack.
- Symptom 2: SSL 3.0 Fallback protection

- Condition: OpenSSL has added support for TLS_FALLBACK_SCSV to allow applications to block the ability for a MITM attacker to force a protocol downgrade. Some client applications (such as browsers) will reconnect using a downgraded protocol to work around interoperability bugs in older servers. This could be exploited by an active man-in-the-middle to downgrade connections to SSL 3.0 even if both sides of the connection support higher protocols. SSL 3.0 contains a number of weaknesses including POODLE (CVE-2014-3566).
- Symptom 3: CVE-2014-3568
- Condition: When OpenSSL is configured with "no-ssl3" as a build option, servers could accept and complete a SSL 3.0 handshake, and clients could be configured to send them.

201410290401

- Symptom: CVE-2014-3508
- Condition: A flaw in OBJ_obj2txt may cause pretty printing functions such as X509_name_oneline, X509_name_print_ex et al. to leak some information from the stack. Applications may be affected if they echo pretty printing output to the attacker.

201504200458

- Symptom 1:CVE-2014-3571
- Condition:A carefully crafted DTLS message can cause a segmentation fault in OpenSSL due to a NULL pointer dereference. This could lead to a Denial Of Service attack.
- Symptom 2:CVE-2015-0206
- Condition:A memory leak can occur in the dtls1_buffer_record function under certain conditions. In particular this could occur if an attacker sent repeated DTLS records with the same sequence number but for the next epoch. The memory leak could be exploited by an attacker in a Denial of Service attack through memory exhaustion.
- Symptom 3:CVE-2015-0205
- Condition:An OpenSSL server will accept a DH certificate for client authentication without the certificate verify message. This effectively allows a client to authenticate without the use of a private key. This only affects servers which trust a client certificate authority which issues certificates containing DH keys.
- Symptom 4:CVE-2014-3570
- Condition:Bignum squaring (BN_sqr) may produce incorrect results on some platforms, including x86_64. This bug occurs at random with a very low probability, and is not known to be exploitable in any way.
- Symptom 5:CVE-2015-0204
- Condition:An OpenSSL client will accept the use of an RSA temporary key in a non-export RSA key exchange ciphersuite. A server could present a weak temporary key and downgrade the security of the session.
- Symptom 6:CVE-2014-3572
- Condition:An OpenSSL client will accept a handshake using an ephemeral ECDH ciphersuite using an ECDSA certificate if the server key exchange message is omitted. This effectively removes forward secrecy from the ciphersuite.
- Symptom 7:CVE-2014-8275
- Condition:By modifying the contents of the signature algorithm or the encoding of the signature, it is possible to change the certificate's fingerprint. Only custom applications that rely on the uniqueness of the fingerprint may be affected.
- Symptom 8:CVE-2014-3569
- Condition:The ssl23_get_client_hello function in s23_srvr.c in OpenSSL 0.9.8zc, 1.0.0o, and 1.0.1j does not properly handle attempts to use unsupported protocols, which allows remote attackers to cause a denial of service (NULL pointer dereference and daemon crash) via an

unexpected handshake, as demonstrated by an SSLv3 handshake to a no-ssl3 application with certain error handling.

201504210231

- Symptom: CVE-2015-1799
- Condition: Authentication doesn't protect symmetric associations against DoS attacks.

201504140088

- Symptom 1: CVE-2015-0209
- Condition: A malformed EC private key file consumed via the d2i_ECPrivateKey function could cause a use after free condition. This could lead to a DoS attack or memory corruption for applications that receive EC private keys from untrusted sources.
- Symptom 2: CVE-2015-0286
- Condition: DoS vulnerability in certificate verification operation. Any application which performs certificate verification is vulnerable including OpenSSL clients and servers which enable client authentication.
- Symptom 3: CVE-2015-0287
- Condition: Reusing a structure in ASN.1 parsing may allow an attacker to cause memory corruption via an invalid write. Applications that parse structures containing CHOICE or ANY DEFINED BY components may be affected.
- Symptom4:CVE-2015-0288
- Condition: The function X509_to_X509_REQ will crash with a NULL pointer dereference if the certificate key is invalid.
- Symptom 5: CVE-2015-0289
- Condition: The PKCS#7 parsing code does not handle missing outer ContentInfo correctly. An attacker can craft malformed ASN.1-encoded PKCS#7 blobs with missing content and trigger a NULL pointer dereference on parsing.

201506190058

- Symptom 8: CVE-2015-3143
- Condition: cURL and libcurl 7.10.6 through 7.41.0 does not properly re-use NTLM connections, which allows remote attackers to connect as other users via an unauthenticated request.
- Symptom 9: CVE-2015-3148
- Condition: cURL and libcurl 7.10.6 through 7.41.0 does not properly re-use authenticated Negotiate connections, which allows remote attackers to connect as other users via a request.

201411240090

- Symptom: The promiscuous mode of an Ethernet interface is set if the link mode of the interface is changed from route to bridge and then changed back to route.
- Condition: This symptom might occur if the following operations have been performed on the Ethernet interface:
  - **a.** Change the link mode from route to bridge.
  - **b.** Change the link mode back to route.

201505250351

- Symptom 1: NQA TCP operation failure causes service interruption on the virtual router.
- Condition: This symptom might occur if a large number of NQA TCP operations are performed over a long period of time, for example, 24 days. TCP port resources are exhausted, and TCP operations fail.
- Symptom 2: A VSR virtual router (8 vCPUs, 4 GB memory) created by using a VM template cannot start.

- Condition: This symptom might occur if the server hosts a large number of VMs, including compute node VMs.

201505090069(CVE-2015-5434)

- Symptom: When an interface without MPLS enabled receives MPLS-labeled packets, the interface incorrectly forwards the MPLS-labeled packets to the next LSR by LFIB entry.
- Condition: This symptom occurs when the interface does not have MPLS enabled and the interface receives MPLS-labeled packet that match the FIB entries.

201505110083

- Symptom: The MAC address modification for a Layer 3 aggregate interface does not take effect.
- Condition: This symptom might occur if the MAC address of the Layer 3 aggregate interface is modified.

201505110083

- Symptom: Services conveyed by the VSR virtual router are interrupted for about 50 minutes after the virtual router has been forwarding traffic for a long period of time. IPsec statistics show that IPsec authentication failure causes packet loss.
- Condition: This symptom might occur if the virtual router sets up an IPsec tunnel to a Hillstone gateway, and the virtual router and the gateway use different encryption algorithms for PFS negotiation.

201507130320

- Symptom: DHCP clients cannot obtain IP addresses when a VXLAN IP gateway acts as a DHCP server.
- Condition: This symptom might occur if the DHCP clients require the DHCP server to send unicast replies.

201404240335

- Symptom: sFlow cannot be used in this release.
- Condition: This symptom might occur if you attempt to use sFlow to monitor the network flow.

201406150005

- Symptom: After an EVI tunnel interface is deleted, the promiscuous mode of the associated Layer 2 interface is not set.
- Condition: This symptom might occur if the following conditions exist:
  - A Layer 3 interface is changedtothe Layer2 interface,and the promiscuous modeis set.
  - After the EVI link is set up successfully, delete the EVI tunnel interface.

201405280245

- Symptom: A dual-homed EVI network does not work.
- Condition: This symptom might occur if the dual-homed EVI network causes loops and flooding occurs.

201404160272

- Symptom: VSR1000 automatically saves the current configuration during a scheduled reboot.
- Condition: This symptom might occur if the current configuration changed after last saving and a scheduled reboot is enabled.

201406160178

- Symptom: Some system exceptions might occur when kernel thread deadloop detection is enabled.

- Condition: This symptom might occur if kernel thread deadloop detection is enabled.

201405260541

- Symptom: When QinQ termination is configured on a subinterface, no packet statistic is collected forthe subinterface.
- Condition: This symptom might occur if QinQ termination is configured on the subinterface.

Resolved problems in VSR1000_HP-CMW710-E0301-X64

201401160131

- Symptom: If there is not any configuration file on the VSR virtual router, the VSR virtual router cannot obtain the configuration settings from the BIMS server.
- Condition: This symptom might occur if there is not any configuration file on the VSR virtual router.

Resolved problems in VSR1000_HP-CMW710-R0202-X64

201312310514

- Symptom: The VSR virtual router will tear down the TCP connections in established state when receiving wrong TCP packets.
- Condition: This symptom might occur if those TCP connections in established state receive TCP SYN packetsthat carry a sequence number falling into the connection receiving window and then RST packetsare sent.

201311280047

- Symptom: BothLB and NAT functions are enabled on the VSR virtual router. A data stream that matches the LB and NAT services at the same time might cause VSR crash.
- Condition: This symptom might occur if both LB and NAT functionsare enabled.

Resolved problems in VSR1000_HP-CMW710-E0102-X64

4000284306

- Symptom: Stable MPLS LDP sessions cannot be established betweenan MSR router and a VSR1000 virtual router.
- Condition: This symptom might occur if tunnel interfaces are used to create MPLS LDP sessions.

201308140533

- Symptom: For E0101P01 release, some services of IPv6 Session are not available, including ASPF, Connection-Limit, Load Balancing, and APR.
- Condition: This symptom might occur if these services are used in an IPv6 environment.

Resolved problems in VSR1000_HP-CMW710-E0101P01-X64

First release.

Support and other resources

Accessing Hewlett Packard Enterprise Support

- For live assistance, go to the Contact Hewlett Packard Enterprise Worldwide website:

  www.hpe.com/assistance
- To access documentation and support services, go to the Hewlett Packard Enterprise Support Center website:

  www.hpe.com/support/hpesc

Information to collect:

- Technical support registration number (if applicable).
- Product name, model or version, and serial number.
- Operating system name and version.
- Firmware version.
- Error messages.
- Product-specific reports and logs.
- Add-on products or components.
- Third-party products or components.

Documents

To find related documents, see the Hewlett Packard Enterprise Support Center website at http://www.hpe.com/support/hpesc.

- Enter your product name or number and click **Go**. If necessary, select your product from the resulting list.
- For a complete list of acronyms and their definitions, see HPE FlexNetwork technology acronyms.

Related documents

The following documents provide related information:

- *HP VSR1000 Virtual Services Router Installation and Getting Started Guide*
- *HP VSR1000 Virtual Services Router Configuration Guides*
- *HP VSR1000 Virtual Services Router Command References*

Documentation feedback

Hewlett Packard Enterprise is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation

Feedback (docsfeedback@hpe.com). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.

Appendix A Feature list

Hardware features

None

Software features

Table 6 Software features of the VSR1000 series

Category	Features
Interface	Layer 2 Ethernet Interface (It is switched from Layer 3 Ethernet interface by executing CLI, only used for layer 2 access)
	Layer 3 Ethernet Interface
	Layer 3 Ethernet Subinterface
	Ethernet_II, Ethernet_SNAP, 802.3x
	802.1Q VLAN, VLAN/QinQ Terminating
	PPPoE Client (IPv4), PPPoE Server (IPv4/IPv6)
	LLDP
	ANCP (Access Node Control Protocol) Server
IP Routing	Static Routing (IPv4/IPv6)
	Dynamic Routing: RIPv1/v2, RIPng, OSPFv2, OSPFv3, BGP, BGP4+, IS-IS, IS-ISv6
	Multicast Routing: IGMPv1/v2/v3, MLDv1/v2, PIM, IPv6 PIM, MSDP
	Routing Policy
	Multi-Topology Routing (MTR)
	DCN (Data Communication Network)
IP Services	ARP, Gratuitous ARP, Proxy ARP (Common Proxy ARP, Local Proxy ARP) ARP Table: Dynamic ARP Entry, Static ARP Entry (Short Static Entry), OpenFlow ARP Entry, Rule ARP Entry(for IPoE/VXLAN ISIS/TRILL) ARP PnP (Plug and Play) ARP Suppression ARP Direct Route Advertisement
	IPv4/IPv6 Dual Stack
	IPv6 Transition: DS-Lite, Tunneling, 6PE, AFT(NAT64)
	Forwarding/Fast Forwarding (Unicast/Multicast)
	TCP, UDP, IP Option, IP Unnumbered
	PBR
	Ping, Trace
	DHCP Server, DHCP Relay, DHCP Client

	DNS Client, DNS Proxy, DDNS
	FTP Server, FTP Client, TFTP Client
	Telnet Server, Telnet Client
	HTTP/HTTPS Server
	UDP Helper
	NTPv3/NTPv4/SNTPv3/SNTPv4
	WAAS (Wide Area Application Services)
MPLS	LDP, LSM, Static LSP, Static CR-LSP
	L3VPN, L2VPN, VPLS L2VPN access to L3VPN or IP backbone
	6VPE
	MPLS TE, RSVP TE
QoS	Traffic Classification: based on port, MAC address, IP address, IP priority, DSCP priority, TCP/UDP port number, and protocol type
	Traffic Policing: CAR, LR
	Traffic Shaping: GTS
	Congestion Management: FIFO, WFQ, CBQ
	Congestion Avoidance: Tail-Drop, WRED
	H-QoS
	MPLS QoS
Security	PKI, IKEv1
	IPSec Encryption (DES, 3DES, AES-128, AES-192, AES-256, ESP-NULL) IPSec Authentication (HMAC-MD5-96, HMAC-SHA1-96, AH-HMAC-MD5-96, AH-HMAC-SHA1-96)
	ADVPN
	SSL
	VPDN: L2TPv2
	GRE
	NAT/NAPT
	AAA (Local Authentication, RADIUS, HWTACACS, LDAP), RBAC, Portal
	ACL, Packet Filter, ASPF
	Security Zone, Object Group, Object Policy
	FIPS Mode
	uRPF(unicast Reverse Path Forwarding)
	Attack Detection and Protection
	Session Management, Connection Limit, Password Management, PBAR

DC	Ethernet Virtual Interconnect (EVI)
	VXLAN
	OpenFlow 1.3
	ENDP
	Service chain
High Availability	VRRP/VRRPv3
	BFD
	Track
	Interface Backup
	Interface Reth and Redundancy Group
	IRF(Layer 2)
	Monitor Link
	Outbound Link Load Balancing
	Layer 4 and Layer 7 Server Load Balancing
Management & Maintenance	CLI: Console Port, SSHv1.5/2.0, Telnet
	Network Management: SNMPv1/v2c/v3, MIB, TR069(CWMP), NETCONF over HTTP/HTTPS/SSH
	Network Monitoring: RMON, Syslog, NQA, sFlow, NetStream, EAA
	Others: Automatic Configuration, File System, TCL, Python
Deployment	OVF with Start-up Configuration Deployment

Appendix B Registering software

IMPORTANT:

To get full performance, please purchase and install a valid license. To make the new license take effect, you should restart the HPE VSR1000 virtual router after the license activation file has been installed.

The HP website (http://hpe.com/networking/mynetworking/) generates a license activation file based on the Device ID file (DID) and a suitable purchased license. To use the software function of the HP VSR1000 virtual router, you must install the license activation file on the device. The same website can provide a free 60-day evaluation license.

The software registration procedure requires the following steps:

- Purchasing a license or obtaining a free 60-day evaluation license
- Obtaining the device information file
- Registering the license on the HP website
- Installing the activation file

Full details on how to license your VSR1000 product can be found in the HPE VSR1000 Virtual Services Router Installation and Getting Started Guide. See "Related documents".

Appendix C Upgrading software

This section describes how to upgrade system software while the router is operating normally or when the router cannot correctly start up.

System software file types

System software images are in .bin format (for example, main.bin) and run at startup. You can set a system software image as a **main**, **backup**, or **secure** image.

At startup, the router always attempts to boot first with the main system software image. If the attempt fails, for example, because the image file is corrupted, the router tries to boot with the backup system software image. If the attempt still fails, the router tries to boot with the secure system software image. If all attempts fail, the router displays a failure message.

The following software types are available:

- **Comware image**—Includes the following image subcategories:
  - **Boot image**—A .bin file that contains the Linux operating system kernel. It provides process management, memory management, file system management, and the emergency shell.
  - **System image**—A .bin file that contains the minimum feature modules required for device operation and some basic features, including device management, interface management, configuration management, and routing. To have advanced features, you must purchase feature packages.
- **Patch packages**—Irregularly released packages for fixing bugs without rebooting the device. A patch package does not add new features or functions.

Boot image, and system image are required for the system to work. These images would be released as a whole in one .ipe package file. If an .ipe file is used, the system automatically decompresses the file, loads the .bin boot and system images and sets them as startup software images. Typically, the Boot ROM and startup software images for the device are released in an .ipe file.

Upgrade methods

You can upgrade system software by using one of the following methods:

Upgrade method	Remarks
Upgrading from the CLI	• You must reboot the router to complete the upgrade. • This method can interrupt ongoing network services.
Upgrading from the ISO image	• Use this method when the router cannot correctly start up. • You must reboot the router to complete the upgrade.

Preparing for the upgrade

Before you upgrade system software, complete the following tasks:

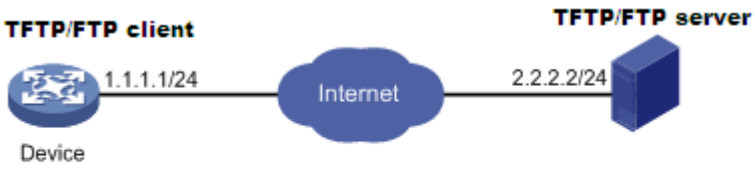
- Set up the upgrade environment as shown in Figure 1
- Configure routes to make sure that the router and the file server can reach each other.

- Run a TFTP or FTP server on the file server.
- Log in to the CLI of the router through the console port.
- Copy the upgrade file to the file server and correctly set the working directory on the TFTP or FTP server.

IMPORTANT:

Make sure that the upgrade has minimal impact on the network services. During the upgrade, the router cannot provide any services.

Figure 1 Set up the upgrade environment



Upgrading from the CLI

You can use the TFTP or FTP commands on the router to access the TFTP or FTP server to back up or download files.

UsingTFTP to upgrade software

This section describes how to upgrade system software by using TFTP.

Backing up the running system software image and configuration file

1. Perform the save command in any view to save the current configuration.

```
<Sysname>save
The current configuration will be written to the device. Are you sure? [Y/N]:y
Please input the file name(*.cfg)[flash:/startup.cfg]
(To leave the existing filename unchanged, press the enter key):
flash:/startup.cfg exists, overwrite? [Y/N]:y
 Validating file. Please wait....
 Configuration is saved to device successfully.
<Sysname>
```

2. Perform the dir command in user view to identify the system software image and configuration file names and verify that the disk has sufficient space for the new system software image.

```
<Sysname>dir
Directory of flash:
   0     drw-         -   Jun 28 2013 14:41:16   logfile
   1     drw-         -   Jun 28 2013 14:42:56   domain1
   2     -rw-     16256   Jun 28 2013 14:43:40   p2p_default.mtd
   3     -rw-      1694   Jun 28 2013 14:47:12   startup.cfg
   4     -rw-      3432   Jun 28 2013 14:47:10   system.xml
5     -rw-     9500672 Sep 13 2013 03:49:46   VSR1000-CMW710-BOOT-E0101P01-X64.bin
```

```
6     -rw-   65973248 Sep 13 2013 03:49:46   VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
7     -rw-     75481088 Sep 13 2013 03:48:14   VSR1000_HP-CMW710-E0101P01-X64.ipe
7311344 KB total (7104336 KB free)
<Sysname>
```

3. Perform the tftp put command in user view to upload the VSR1000_HP-CMW710-E0101P01-X64.ipe file to the TFTP server.

```
<Sysname>tftp 2.2.2.2 put VSR1000_HP-CMW710-E0101P01-X64.ipe
  File will be transferred in binary mode
  Sending file to remote TFTP server. Please wait... \
  TFTP: 31131648bytes sent in 70 second(s).
  File uploaded successfully.
<Sysname>
```

4. Perform the tftp put command in user view to upload the startup.cfg file to the TFTP server.

```
<Sysname>tftp 2.2.2.2 put startup.cfg
  File will be transferred in binary mode
  Sending file to remote TFTP server. Please wait... \
  TFTP:     1694 bytes sent in 0 second(s).
  File uploaded successfully.
<Sysname>
```

Upgrading the system software

1. Perform the tftp get command in user view to download the system software image file VSR1000_HP-CMW710-E0101P01-X64.ipe to the disk on the router.

```
<Sysname>tftp 2.2.2.2 get VSR1000_HP-CMW710-E0101P01-X64.ipe
  File will be transferred in binary mode
  Downloading file from remote TFTP server, please wait...|
  TFTP: 31131648bytes received in 70 second(s)
  File downloaded successfully.
<Sysname>
```

2. Perform the boot-loader command in user view to load the file VSR1000_HP-CMW710-E0101P01-X64.ipe and specify the file as the main image file at the next reboot.

```
<Sysname>boot-loader file VSR1000_HP-CMW710-E0101P01-X64.ipe main
  Images in IPE:
  VSR1000-CMW710-BOOT-E0101P01-X64.bin
  VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
This command will set the main startup software images. Continue? [Y/N]:y
Add images to the device.
flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin already exists on the device.
flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin already exists on the device.
The images that have passed all examinations will be used as the main startup software
images at the next reboot on the device.
<Sysname>
```

3. Perform the display boot-loader command in user view to verify that the file has been loaded.

```
<Sysname>display boot-loader
Software images on the device:
Current software images:
  flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
  flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
```

```
Main startup software images:
  flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
  flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
Backup startup software images:
<Sysname>
```

4. Perform the reboot command in user view to reboot the router.

```
<Sysname>reboot
Start to check configuration with next startup configuration file, please
wait.........DONE!
This command will reboot the device. Continue? [Y/N]:y
Now rebooting, please wait...
```

5. After the reboot is complete, perform the display version command to verify that the system software image is correct.

```
<Sysname>display version
HP Comware Software, Version 7.1.047, ESS 0101P01
Copyright (c) 2010-2013 Hewlett-Packard DevelopmentCompany,L.P.
HP VSR1000 uptime is 0 weeks, 0 days, 0 hours, 0 minutes
Last reboot reason : User reboot
Boot image: flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
Boot image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46
System image: flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
System image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46

CPU ID: 0x01000101, vCPUs: Total 1, Available 1
1.00G bytes RAM Memory
Basic    BootWare Version:  1.00
Extended BootWare Version:  1.00
[SLOT  1]VNIC-E1000         (Driver)1.0
[SLOT  2]VNIC-E1000         (Driver)1.0
<Sysname>
```

Using FTP to upgrade software

This section describes how to upgrade system software by using FTP.

Backing up the running system software image and configuration file

1. Perform the save command in any view to save the current configuration.

```
<Sysname>save
The current configuration will be written to the device. Are you sure? [Y/N]:y
Please input the file name(*.cfg)[flash:/startup.cfg]
(To leave the existing filename unchanged, press the enter key):
flash:/startup.cfg exists, overwrite? [Y/N]:y
 Validating file. Please wait....
 Configuration is saved to device successfully.
<Sysname>
```

2. Perform the dir command in user view to identify the system software image and configuration file names and verify that the disk has sufficient space for the new system software image.

```
<Sysname>dir
Directory of flash:
   0     drw-         -  Jun 28 2013 14:41:16   logfile
   1     drw-         -  Jun 28 2013 14:42:56   domain1
   2     -rw-     16256  Jun 28 2013 14:43:40   p2p_default.mtd
   3     -rw-      1694  Jun 28 2013 14:47:12   startup.cfg
   4     -rw-      3432  Jun 28 2013 14:47:10   system.xml
5     -rw-     9500672 Sep 13 2013 03:49:46   VSR1000-CMW710-BOOT-E0101P01-X64.bin
6     -rw-    65973248 Sep 13 2013 03:49:46   VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
7     -rw-     75481088 Sep 13 2013 03:48:14   VSR1000_HP-CMW710-E0101P01-X64.ipe
7311344 KB total (7104336 KB free)
<Sysname>
```

3. Perform the ftp command in user view to access the FTP server.

```
<Sysname>ftp 2.2.2.2
Press CTRL+C to abort.
Connected to 2.2.2.2 (2.2.2.2).
220 WFTPD 2.0 service (by Texas Imperial Software) ready for new user
User (2.2.2.2:(none)): user001
331 Give me your password, please
Password:
230 Logged in successfully
Remote system type is MSDOS
ftp>
```

4. Perform the put command in FTP client view to upload the VSR1000_HP-CMW710-E0101P01-X64.ipe file to the FTP server.

```
ftp>put VSR1000_HP-CMW710-E0101P01-X64.ipe
227 Entering passive mode (2,2,2,2,7,210)
125 Using existing data connection
226 Closing data connection; File transfer successful.
FTP: 31131648 byte(s) sent in 21.363 second(s), 1116.00Kbyte(s)/sec.
ftp>
```

5. Perform the put command in FTP client view to upload the startup.cfg file to the FTP server.

```
[ftp]put startup.cfg
227 Entering passive mode (2,2,2,2,7,177)
125 Using existing data connection
226 Closing data connection; File transfer successful.
FTP: 1677 byte(s) sent in 0.142 second(s), 11.00Kbyte(s)/sec.
ftp>
```

Upgrading the system software

1. Perform the get command in FTP client view to download the system software image file VSR1000_HP-CMW710-E0101P01-X64.ipe to the router.

```
ftp> get VSR1000_HP-CMW710-E0101P01-X64.ipe
227 Entering passive mode (2,2,2,2,7,225)
125 Using existing data connection
226 Closing data connection; File transfer successful.
FTP: 31131648 byte(s) received in 30.907 second(s), 772.00K byte(s)/sec.
ftp>
```

2. Perform the quit command in FTP client view to return to user view.

```
ftp>quit
221 Service closing control connection
<Sysname>
```

3. Perform the boot-loader command in user view to load the file VSR1000_HP-CMW710-E0101P01-X64.ipe and specify the file as the main image file at the next reboot.

```
<Sysname>boot-loader file flash:/VSR1000_HP-CMW710-E0101P01-X64.ipe main
Images in IPE:
  VSR1000-CMW710-BOOT-E0101P01-X64.bin
  VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
This command will set the main startup software images. Continue? [Y/N]:y
Add images to the device.
flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin already exists on the device.
flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin already exists on the device.
The images that have passed all examinations will be used as the main startup software
images at the next reboot on the device.
<Sysname>
```

4. Perform the display boot-loader command in user view to verify that the file has been loaded.

```
<Sysname>display boot-loader
Software images on the device:
Current software images:
  flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
  flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
Main startup software images:
  flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
  flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
Backup startup software images:
  None
<Sysname>
```

5. Perform the reboot command in user view to reboot the router.

```
<Sysname>reboot
Start to check configuration with next startup configuration file, please
wait.........DONE!
This command will reboot the device. Continue? [Y/N]:y
Now rebooting, please wait...
```

6. After the reboot is complete, perform the display version command to verify that the system software image is correct.

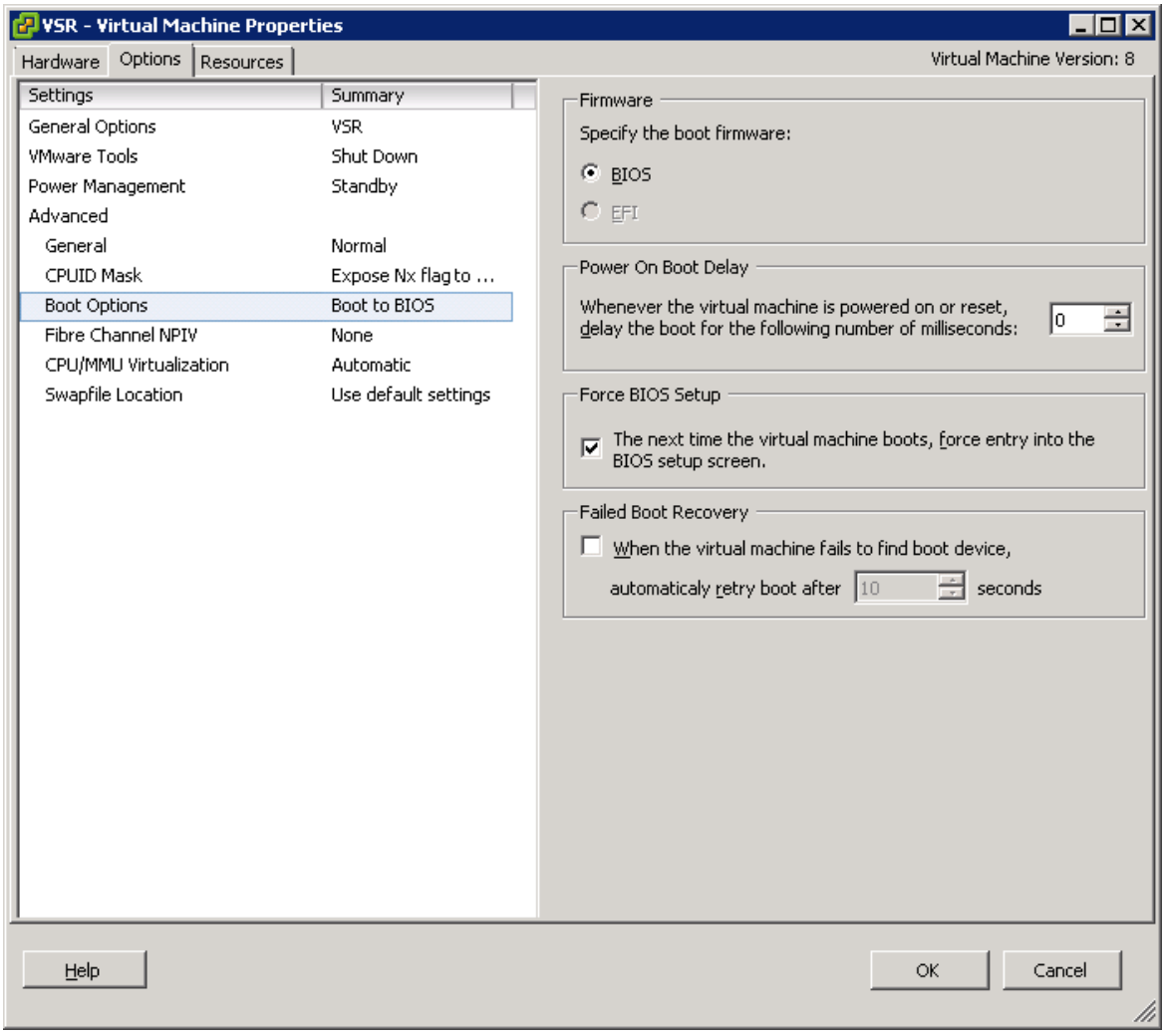
```
<Sysname>display version
HP Comware Software, Version 7.1.047, ESS 0101P01
Copyright (c) 2010-2013 Hewlett-Packard DevelopmentCompany,L.P.
HP VSR1000 uptime is 0 weeks, 0 days, 0 hours, 0 minutes
Last reboot reason : User reboot
Boot image: flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
Boot image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46
System image: flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
System image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46
```

```
CPU ID: 0x01000101, vCPUs: Total 1, Available 1
1.00G bytes RAM Memory
Basic    BootWare Version:  1.00
Extended BootWare Version:  1.00
[SLOT  1]VNIC-E1000         (Driver)1.0
[SLOT  2]VNIC-E1000         (Driver)1.0
<Sysname>
```

Upgrading from the ISO image

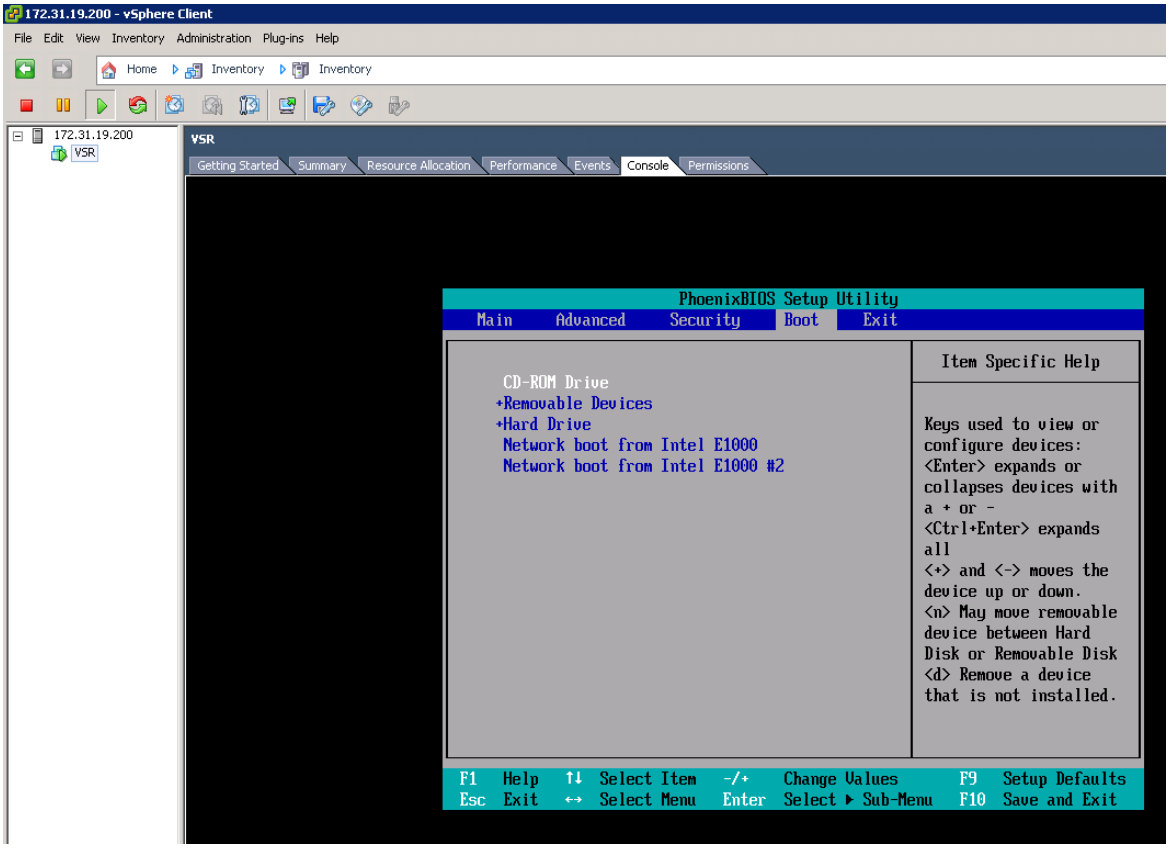
1. Select the newly created virtual machine from the navigation tree, and select Edit from the right-click menu. Click the Options tab. Select the box in the Force BIOS Setup area, and click OK.

Figure 2 Selecting Force BIOS Setup



2. Select the newly created virtual machine from the navigation tree, and click  to start the virtual machine. The page inFigure 3 appears. In the **Console** tab, select the **Boot** tab, configure the virtual machine to preferentially boot from the CD-ROM drive, save the configuration, and exit.

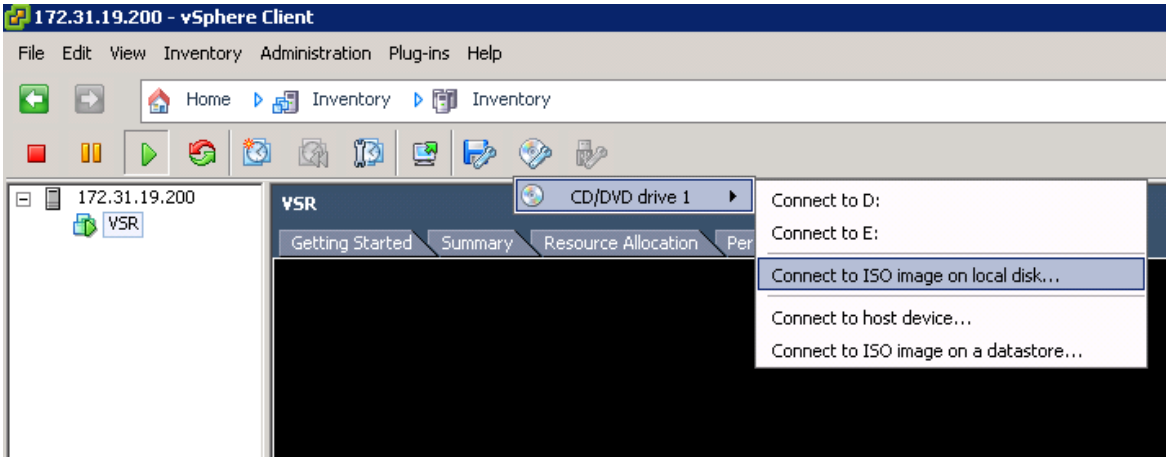
Figure 3 Configuring the virtual machine to preferentially boot from the CD-ROM drive



3. Connecting the CD drive of the virtual machine to the ISO image of the VSR and restarting the virtual machine

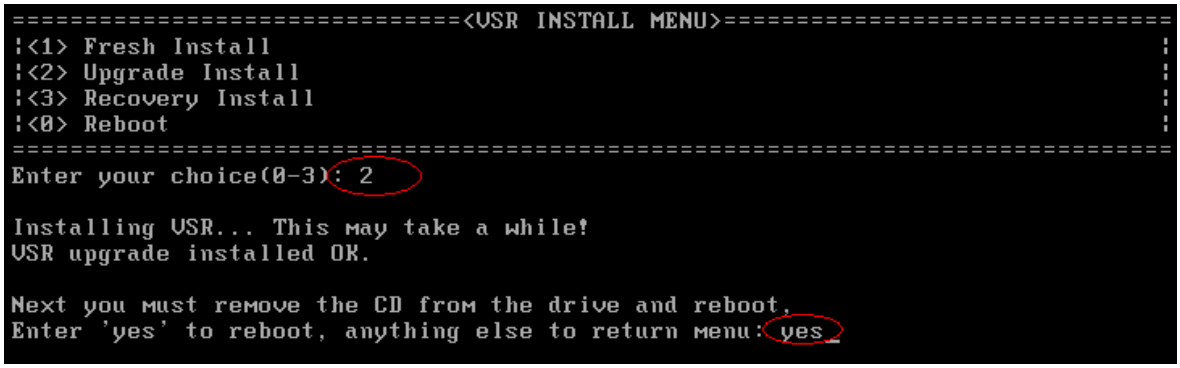
Click  to connect the CD drive of the virtual machine to the ISO image of the VSR and restart the virtual machine.

Figure 4 Connecting the CD drive of the virtual machine to the ISO image of the VSR



4. Booting the virtual machine from the ISO image and entering the installation interface
5. The installation interface is as shown inFigure 5. Enter 2 to select **Upgrade Install** to upgrade the VSR to the version in the ISO image.

Figure 5 Installation interface



6. After the installation is finished, disconnect the CD drive as shown in Figure 6andFigure 7. Then, enter **yes** to reboot the system and complete installing VSR.

Figure 6 Disconnecting the CD drive 1

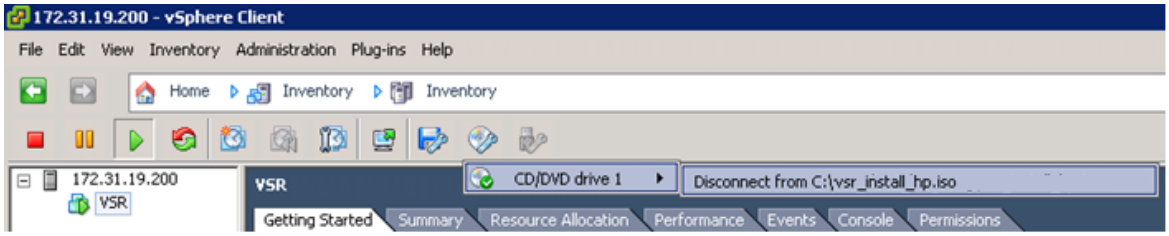
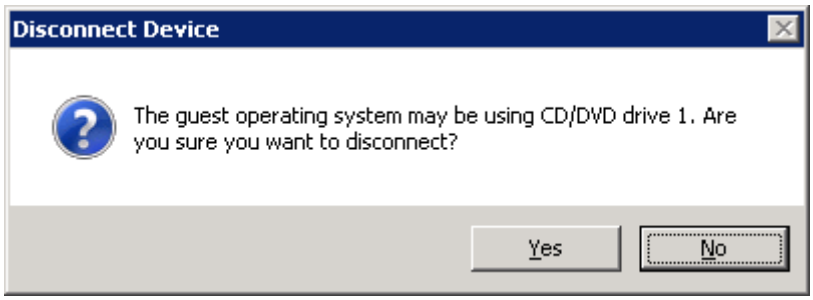


Figure 7 Disconnecting the CD drive 2



7. After the reboot is complete, perform the **display version** command to verify that the system software image is correct.

```
<Sysname>display version
HP Comware Software, Version 7.1.047, ESS 0101P01
Copyright (c) 2010-2013 Hewlett-Packard DevelopmentCompany,L.P.
HP VSR1000 uptime is 0 weeks, 0 days, 0 hours, 0 minutes
Last reboot reason : User reboot
Boot image: flash:/VSR1000-CMW710-BOOT-E0101P01-X64.bin
Boot image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46
System image: flash:/VSR1000-CMW710-SYSTEM-E0101P01-X64.bin
System image version: 7.1.047, ESS 0101P01
  Compiled Sep 13 2013 03:49:46

CPU ID: 0x01000101, vCPUs: Total 1, Available 1
```

```
1.00G bytes RAM Memory
Basic    BootWare Version:  1.00
Extended BootWare Version:  1.00
[SLOT  1]VNIC-E1000         (Driver)1.0
[SLOT  2]VNIC-E1000         (Driver)1.0
<Sysname>
```

Hewlett Packard Enterprise

HPE VSR1000_HPE-CMW710-E0324-X64Release Notes

Software Feature Changes

Software version:VSR1000_HPE-CMW710-E0324-X64

Contents

Release HPE VSR1000_HPE-CMW710-E0324-X64

None.

Release HPE VSR1000_HPE-CMW710-E0322P01-X64

Thisrelease has the following changes:

- New feature:Setting a local QoS ID match criterion in policy-based routing

New feature:Setting a local QoS IDmatch criterion in policy-based routing

Overview

A policy includes match criteria and actions to be taken on the matching packets. A policy can have one or multiple nodes as follows:

- Each node is identified by a node number. A smaller node number has a higher priority.
- Anodecontains**if-match** and **apply** clauses.An **if-match** clause specifies a match criterion, and an **apply** clause specifies an action.
- A node has a match mode of**permit** or **deny**.

A policy compares packets with nodes in priority order. If a packet matches the criteria on a node, it is processed by the action on the node. Otherwise, it goes to the next node for a match. If the packet does not match the criteria on any node, it is forwarded according to the routing table.

Policy-based routing added support for the **if-match qos-local-id** clause, which sets a local QoS ID match criterion. For more information about QoS, see *ACL and QoS Configuration Guide*.

NOTE:

A node that has noif-matchclauses matches any packet.

Setting a local QoS IDmatch criterionfor a node

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter policy node view.	**policy-based-route***policy-name*[**deny** | **permit**]**node***node-number*	N/A
3. Set a local QoS ID match criterion.	**if-match qos-local-id** *local-id-value*	By default, no local QoS ID match criterion is set. Support for this command depends on the device model.

NOTE:

When using the**if-match qos-local-id** clause, make sure one flow corresponds to only one local QoS ID. Packets belong to the same flow if they have the same 5-tuple information (source IP address, destination IP address, source port number, destination port number, and protocol.

Command reference

if-match qos-local-id

Use **if-match qos-local-id** to set a local QoS ID match criterion.

Use **undoif-match qos-local-id** to restore the default.

Syntax

if-match qos-local-id *local-id-value*

undo if-match qos-local-id

Default

No local QoS ID match criterion is set.

Views

Policy node view

Predefined user roles

network-admin

mdc-admin

Parameters

local-id-value: Specifies a local QoS ID in the range of 1 to 4095.

Examples

Match packets with local QoS ID 200.

```
<Sysname> system-view
[Sysname] policy-based-route aa permit node 11
[Sysname-pbr-aa-11] if-match qos-local-id200
```

Release HPE VSR1000_HPE-CMW710-E0322-X64

Thisrelease has the following changes:

Modified feature: Domain

Modified feature: BGP

Modified feature: Domain

Feature change description

Expand the max number of domain from 64 to 1024.

Command changes

None.

Modified feature: BGP

Feature change description

To ignorethe first AS number of EBGP route updates received from a peer or peer group (IPv4):

Step	Command	Remarks
4. Enter system view.	**system-view**	N/A
5. Enter BGP instance view or BGP-VPN instance view.	• Enter BGP instance view: **bgp** *as-number*[**instance** *instance-name*] [**multi-session-thread**] • Enter BGP-VPN instance view: **a.** **bgp** *as-number*[**instance** *instance-name*] [**multi-session-thread**] **b.** **ip vpn-instance** *vpn-instance-name*	N/A
6. Configure BGP to ignore the first AS number of EBGP route updates received from a peer or peer group.	**peer** { *group-name* | *ipv4-address* [*mask-length*]} **ignore-first-as**	By default, BGP checks the first AS number of EBGP-learned route updates.

To ignorethe first AS number of EBGP route updates received from a peer or peer group (IPv6):

Step	Command	Remarks
7. Enter system view.	**system-view**	N/A

Step	Command	Remarks
8. Enter BGP instance view or BGP-VPN instance view.	• Enter BGP instance view: **bgp** *as-number*[**instance** *instance-name*] [**multi-session-thread**] • Enter BGP-VPN instance view: **a.** **bgp** *as-number*[**instance** *instance-name*] [**multi-session-thread**] **b.** **ip vpn-instance** *vpn-instance-name*	N/A
9. Configure BGP to ignore the first AS number of EBGP route updates received from a peer or peer group.	**peer** { *group-name* | *ipv6-address* [*prefix-length*]} **ignore-first-as**	By default, BGP checks the first AS number of EBGP-learned route updates.

Command changes

peerignore-first-as

Use **peer ignore-first-as** to configure BGP to ignore the first AS number of EBGP route updates received from a peer or peer group.

Use **undo peer ignore-first-as** to remove the configuration.

Syntax

peer { *group-name* | *ipv4-address*[*mask-length*] | *ipv6-address* [*prefix-length*]} **ignore-first-as**

undo peer { *group-name* | *ipv4-address*[*mask-length*] | *ipv6-address* [*prefix-length*]} **ignore-first-as**

Default

BGP checks the first AS number ofan EBGP-learned route update. If the first AS number is neither that of the BGP peer nor a private AS number, the BGP routerdisconnects the BGP session to the peer.

Views

BGP instance view

BGP-VPN instance view

Predefined user roles

network-admin

mdc-admin

Parameters

group-name: Specifies a peer group by its name, a case-sensitive string of 1 to 47 characters. The peer group must have been created.

ipv4-address: Specifies a peer by its IPv4 address. The peer must have been created.

mask-length: Specifies a mask length in therange of 0 to 32. You can use the *ipv4-address* and *mask-length* arguments together to specify a subnet. If you specify a subnet in this command, BGP ignores the first AS number of EBGP route updates received from all dynamic peers in the subnet.

ipv6-address: Specifies a peer by its IPv6 address. The peer must have been created.

prefix-length: Specifies a prefix length in therange of 0 to 128. You can use the *ipv6-address* and *prefix-length* arguments together to specify a subnet. If you specify a subnet in this command, BGP ignores the first AS number of EBGP route updates received from all dynamic peers in the subnet.

Usage guidelines

This command takes effect onlyon the EBGP routes received after you execute this command. After you execute the **undo peer ignore-first-as** command, BGP advertises a ROUTE-REFRESH message to request the routing information from the EBGP peer or peer group.

Examples

In BGP instance view, configure BGP to ignore the first AS number of EBGP route updates received from peer group **test**.

```
<Sysname> system-view
[Sysname] bgp 100
[Sysname-bgp-default] peer test ignore-first-as
```

Related commands

ignore-first-as

Release HPE VSR1000_HPE-CMW710-E0321P01-X64

This release has the following changes:

New feature:IKEv2

Modified feature: SSHv2

New feature:IKEv2

Overview

Internet Key Exchange version 2 (IKEv2) is an enhanced version of IKEv1. The same as IKEv1, IKEv2 has a set of self-protection mechanisms and can be used on insecure networks for reliable identity authentication, key distribution, and IPsec SA negotiation. IKEv2 provides stronger protection against attacks and higher key exchange ability and needs fewer message exchanges than IKEv1.

IKEv2 negotiation process

Compared with IKEv1, IKEv2 simplifies the negotiation process and is much more efficient.

IKEv2 defines three types of exchanges: initial exchanges, CREATE_CHILD_SA exchange, and INFORMATIONAL exchange.

As shown in Figure 1, IKEv2 uses two exchanges during the initial exchange process: IKE_SA_INIT and IKE_AUTH, each with two messages.

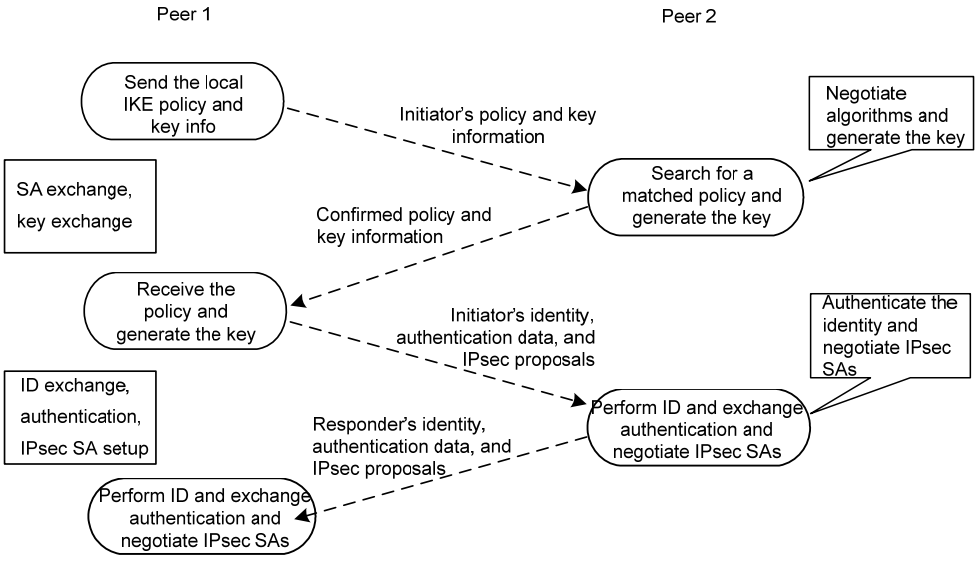
- **IKE_SA_INIT exchange**—Negotiates IKE SA parameters and exchanges keys.
- **IKE_AUTH exchange**—Authenticatesthe identity of the peer and establishes IPsec SAs.

After the four-message initial exchanges, IKEv2 sets up one IKE SA and one pair of IPsec SAs. For IKEv1 to set up one IKE SA and one pair of IPsec SAs, it must go through two phases that use a minimum of six messages.

To set up one more pair of IPsec SAs within the IKE SA, IKEv2goes on to perform an additional two-message exchange—the CREATE_CHILD_SA exchange. One CREATE_CHILD_SA exchange creates one pair of IPsec SAs. IKEv2 also uses the CREATE_CHILD_SA exchange to rekey IKE SAs and Child SAs.

IKEv2 uses the INFORMATIONAL exchange to convey control messages about errors and notifications.

Figure 1 IKEv2 Initial exchange process



New features in IKEv2

DH guessing

In the IKE_SA_INIT exchange, the initiator guesses the DH group that theresponder is most likely touse and sends it in an IKE_SA_INIT request message. If the initiator's guess is correct, the responder responds with an IKE_SA_INIT response message and the IKE_SA_INIT exchange is finished. If the guess is wrong, the responder responds with an INVALID_KE_PAYLOAD message that contains the DH group that it wants to use. The initiator then usesthe DH group selected by the responder to reinitiate the IKE_SA_INIT exchange. The DH guessing mechanism allows for more flexible DH group configuration and enables the initiator to adapt to different responders.

Cookie challenging

Messages for the IKE_SA_INIT exchange are in plain text. An IKEv1 responder cannot confirm the validity of the initiators and must maintain half-open IKE SAs, which makes the responder susceptible to DoS attacks. An attacker can send a large number of IKE_SA_INIT requests with forged source IP addresses to the responder, exhausting the responder's system resources.

IKEv2 introduces the cookie challenging mechanism to prevent such DoS attacks. When an IKEv2 responder maintains a threshold number of half-open IKE SAs, it starts the cookie challenging mechanism.The responder generates a cookie and includes it in the response sent to the initiator. If the initiator initiates a new IKE_SA_INIT request that carries the correct cookie, the responder considers the initiator valid and proceeds with the negotiation. If the carried cookie is incorrect, the responder terminates the negotiation.

The cookie challenging mechanism automatically stops working when the number of half-open IKE SAs drops below the threshold.

IKEv2 SA rekeying

For security purposes, both IKE SAs and IPsec SAs have a lifetime and must be rekeyed when the lifetime expires. An IKEv1 SA lifetime is negotiated. An IKEv2 SA lifetime, in contrast, is configured. If two peers are configured with different lifetimes, the peer with the shorter lifetime always initiates the SA rekeying. This mechanism reduces the possibility that two peers will simultaneously initiate a rekeying. Simultaneous rekeying results in redundant SAsand SA status inconsistency on the two peers.

IKEv2 message retransmission

Unlike IKEv1 messages, IKEv2 messages appear in request/response pairs. IKEv2 uses the Message ID field in the message header to identify the request/response pair. If an initiator sends a request but receives no response with the same Message ID value within a specific period of time, the initiator retransmits the request.

It is always the IKEv2 initiator that initiates the retransmission, and the retransmitted message must use the same Message ID value.

Protocols and standards

- RFC 2408, Internet Security Association and Key Management Protocol(ISAKMP)
- RFC 4306, Internet Key Exchange (IKEv2) Protocol
- RFC 4718, IKEv2 Clarifications and Implementation Guidelines
- RFC 2412, The OAKLEY Key Determination Protocol
- RFC 5996, Internet Key ExchangeProtocol Version 2 (IKEv2)

IKEv2 configuration task list

Determine the following parameters prior to IKEv2 configuration:

- The strengthof the algorithms for IKEv2 negotiation, including the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups. Different algorithms provide different levels of protection. A stronger algorithmmeansbetter resistance to decryption of protected data but requires moreresources. Typically, the longer the key, the stronger the algorithm.
- The local and remote identity authentication methods.
  - To use the pre-shared key authentication method, you must determine the pre-shared key.
  - To use the RSA digital signature authentication method, you must determine the PKI domain for the local end to use. For information about PKI, see "Configuring PKI."

To configure IKEv2, perform the following tasks:

Tasks at a glance	Remarks
(Required.) Configuring an IKEv2 profile	N/A
(Required.) Configuring an IKEv2 policy	N/A
(Optional.) Configuring an IKEv2 proposal	If you specify an IKEv2 proposal in an IKEv2 policy, you must configure the IKEv2 proposal.
Configuring an IKEv2 keychain	Required when either end or both ends use the pre-shared key authentication method.
Configure global IKEv2 parameters • (Optional.) Enabling the cookie challenging feature • (Optional.) Configuring the IKEv2 DPD feature • (Optional.) Configuring the IKEv2 NAT keepalive feature • (Optional.) Configuring IKEv2 address pools	The cookie challenging feature takes effect only on IKEv2 responders.

Configuring an IKEv2 profile

An IKEv2 profile is intended to provide a set of parameters for IKEv2 negotiation. To configure an IKEv2 profile, perform the following tasks:

1. Specify the local and remote identity authentication methods.

   The local and remote identity authentication methods must both be specified and they can be different. You can specify only one local identity authentication method and multiple remote identity authentication methods.

2. Configure the IKEv2 keychain or PKI domain for the IKEv2 profile to use:
   - To use digital signature authentication, configure a PKI domain.
   - To use pre-shared key authentication, configure an IKEv2 keychain.

3. Configure the local ID, the ID that the device uses to identify itself to the peer during IKEv2 negotiation:
   - For digital signature authentication, the device can use an ID of any type. If the local ID is an IP address that is different from the IP address in the local certificate, the device uses the FQDN as the local ID. The FQDN is the device name configured by using the **sysname** command.
   - For pre-shared key authentication, the device can use an ID of any type other than the DN.

4. Configure peer IDs.

   The device compares the received peer ID with the peer IDs of its local IKEv2 profiles. If a match is found, it uses the IKEv2 profile with the matching peer ID for IKEv2 negotiation. IKEv2 profiles will be compared in descending order of their priorities.

5. Specify a local interface or IP address for the IKEv2 profile so the profile can be applied only to the specified interface or IP address. For this task, specify the local address configured in IPsec policy or IPsec policy template view (using the **local-address** command). If no local address is configured, specify the IP address of the interface that uses the IPsec policy.

6. Specify a priority number for the IKEv2 profile. To determine the priority of an IKEv2 profile:
   a. First, the device examines the existence of the **match local** command. An IKEv2 profile with the **match local** command configured has a higher priority.
   b. If a tie exists, the device compares the priority numbers. An IKEv2 profile with a smaller priority number has a higher priority.
   c. If a tie still exists, the device prefers an IKEv2 profile configured earlier.

7. Specify a VPN instance for the IKEv2 profile. The IKEv2 profile is used for IKEv2 negotiation only on the interfaces that belong to the VPN instance.

8. Configure the IKEv2 SA lifetime.

   The local and remote ends can use different IKEv2 SA lifetimes. They do not negotiate the lifetime. The end with a smaller SA lifetime will initiate an SA negotiation when the lifetime expires.

9. Configure IKEv2 DPD to detect dead IKEv2 peers. You can also configure this feature in system view. If you configure IKEv2 DPD in both views, the IKEv2 DPD settings in IKEv2 profile view apply. If you do not configure IKEv2 DPD in IKEv2 profile view, the IKEv2 DPD settings in system view apply.

10. Specify an inside VPN instance. This setting determines where the device should forward received IPsec packets after it de-encapsulates them. If you specify an inside VPN instance, the device looks for a route in the specified VPN instance to forward the packets. If you do not specify an inside VPN instance, the internal and external networks are in the same VPN instance. The device looks for a route in this VPN instance to forward the packets.

11. Configure the NAT keepalive interval.

    Configure this task when the device is behind a NAT gateway. The device sends NAT keepalive packets regularly to its peer to prevent the NAT session from being aged because of no matching traffic.

12. Enable the configuration exchange feature.

    The configuration exchange feature enables the local and remote ends to exchange configuration data, such as gateway address, internal IP address, and route. The exchange includes data request and response, and data push and response.

    This feature typically applies to scenarios where branches and the headquarters communicate through virtual tunnels.

    This feature enables the IPsec gateway at a branch to send IP address requests to the IPsec gateway at the headquarters. When the headquarters receives the request, it sends an IP address to the branch in the response packet. The headquarters can also actively push an IP address to the branch. The branch uses the allocated IP address as the IP address of the virtual tunnel to communicate with the headquarters.

13. Enable AAA authorization.

    The AAA authorization feature enables IKEv2 to request authorization attributes, such as the IKEv2 address pool, from AAA. IKEv2 uses the address pool to assign IP addresses to remote users. For more information about AAA authorization, see "Configuring AAA."

To configure an IKEv2 profile:

Step	Command	Remarks
10. Enter system view.	**system-view**	N/A
11. Create an IKEv2 profile and enter IKEv2 profile view.	**ikev2 profile** *profile-name*	By default, no IKEv2 profiles exist.
12. Configure the local and remote identity authentication methods.	**authentication-method** { **local** | **remote** } { **dsa-signature** | **ecdsa-signature** | **pre-share** | **rsa-signature** }	By default, no local or remote identity authentication method is configured.
13. Specify a keychain.	**keychain***keychain-name*	By default, no keychain is specified for an IKEv2 profile. Perform this task when the pre-shared key authentication method is specified.
14. Specify a PKI domain.	**certificate domain***domain-name*[**sign** | **verify**]	By default, the device uses PKI domains configured in system view. Perform this task when the digital signature authentication method is specified.
15. Configure the local ID.	**identity local** { **address** { *ipv4-address* | **ipv6***ipv6-address* } | **dn** | **email***email-string* | **fqdn***fqdn-name* | **key-id***key-id-string* }	By default, no local ID is configured, and the device uses the IP address of the interface where the IPsec policy applies as the local ID.
16. Configure peer IDs.	**matchremote** { **certificate***policy-name* | **identity** { **address** { { *ipv4-address* [*mask* | *mask-length*] | **range***low-ipv4-address high-ipv4-address* } | **ipv6** { *ipv6-address* [*prefix-length*] | **range***low-ipv6-address high-ipv6-address* } } | **fqdn***fqdn-name* | **email***email-string* | **key-id** *key-id-string* } }	By default, no peer ID is configured. You must configure a minimum of one peer ID on each of the two peers.

Step	Command	Remarks
17. (Optional.) Specify the local interface or IP address to which the IKEv2 profile can be applied.	**match local address** { *interface-type interface-number* | { *ipv4-address* | **ipv6***ipv6-address*} }	By default, an IKEv2 profile can be applied to any local interface or IP address.
18. (Optional.) Specify a priority for the IKEv2 profile.	**priority***priority*	By default, the priority of an IKEv2 profile is 100.
19. (Optional.) Specify a VPN instance for the IKEv2 profile.	**match vrf** { **name***vrf-name*|**any** }	By default, an IKEv2 profile belongs to the public network.
20. (Optional.) Set the IKEv2 SA lifetime for the IKEv2 profile.	**sa duration***seconds*	By default, the IKEv2 SA lifetime is 86400 seconds.
21. (Optional.) Configure the DPD feature for the IKEv2 profile.	**dpd interval***interval* [**retry***seconds*] { **on-demand** | **periodic** }	By default, DPD is disabled for an IKEv2 profile. The global DPD settings in system view are used. If DPD is also disabled in system view, the device does not perform DPD.
22. (Optional.) Specify an inside VPN instance for the IKEv2 profile.	**inside-vrf** *vrf-name*	By default, no inside VPN instance is specified for an IKEv2 profile.The internal and external networks are in the same VPN instance. The device forwards protected data to this VPN instance.
23. (Optional.) Set the IKEv2 NAT keepalive interval.	**nat-keepalive***seconds*	By default, the global IKEv2 NAT keepalive setting is used.
24. (Optional.) Enable the configuration exchange feature.	**config-exchange** {**request** | **set** { **accept** | **send** } }	By default, all configuration exchange options are disabled.
25. (Optional.) Enable AAA authorization.	**aaa authorization domain***domain-name***username***user-name*	By default, AAA authorization is disabled for IKEv2.

Configuring an IKEv2 policy

During the IKE_SA_INIT exchange, each end tries to find a matching IKEv2 policy, using the IP address of the local security gateway as the matching criterion.

- If IKEv2 policies are configured, IKEv2 searches for an IKEv2 policy that uses the IP address of the local security gateway. If no IKEv2 policy uses the IP address or the policy is using an incomplete proposal, the IKE_SA_INIT exchange fails.
- If no IKEv2 policy is configured, IKEv2 uses the system default IKEv2 policy **default**.

The device matches IKEv2 policies in the descending order of their priorities. To determine the priority of an IKEv2 policy:

1. First, the device examines the existence of the **match local address** command. An IKEv2policy with the **match local address** command configured has a higher priority.
2. If a tie exists, the device compares the priority numbers. An IKEv2policy with a smaller priority number has a higher priority.
3. If a tie still exists, the device prefers an IKEv2policy configured earlier.

To configure an IKEv2 policy:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IKEv2 policy and enter IKEv2 policy view.	**ikev2 policy** *policy-name*	By default, an IKEv2policy named **default** exists.
3. Specify the local interface or address used for IKEv2 policy matching.	**match local address** { *interface-type interface-number* | { { *ipv4-address* | **ipv6***ipv6-address* } } }	By default, no local interface or address is used for IKEv2 policy matching, and the policy matches any local interface or address.
4. Specify a VPN instance for IKEv2 policy matching.	**match vrf** { **name***vrf-name* | **any** }	By default, no VPN instance is specified for IKEv2 policy matching. The IKEv2 policy matches all local addresses in the public network.
5. Specify an IKEv2 proposal for the IKEv2 policy.	**proposal***proposal-name*	By default, no IKEv2 proposal is specified for an IKEv2 policy.
6. Specify a priority for the IKEv2 policy.	**priority***priority*	By default, the priority of an IKEv2 policy is 100.

Configuring an IKEv2 proposal

An IKEv2 proposal contains security parameters used in IKE_SA_INIT exchanges, including the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups. An algorithm specified earlier has a higher priority.

A complete IKEv2 proposal must have at least one set of security parameters, including one encryption algorithm, one integrity protection algorithm, one PRF algorithm, and one DH group.

You can specify multiple IKEv2 proposals for an IKEv2 policy. A proposal specified earlier has a higher priority.

To configure an IKEv2 proposal:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A

Step	Command	Remarks
2. Create an IKEv2 proposal and enter IKEv2 proposal view.	**ikev2 proposal***proposal-name*	By default, an IKEv2 proposal named **default** exists. In non-FIPS mode, the default proposal uses the following settings: • Encryption algorithmsAES-CBC-128 and 3DES. • Integrity protection algorithms HMAC-SHA1 and HMAC-MD5. • PRF algorithms HMAC-SHA1 and HMAC-MD5. • DH groups2 and 5. In FIPS mode, the default proposal uses the following settings: • Encryption algorithmsAES-CBC-128 and AES-CTR-128. • Integrity protection algorithms HMAC-SHA1 and HMAC-SHA256. • PRF algorithms HMAC-SHA1 and HMAC-SHA256. • DH groups14 and 19.
3. Specify the encryption algorithms.	In non-FIPS mode: **encryption** { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256** | **camellia-cbc-128** | **camellia-cbc-192** | **camellia-cbc-256** | **des-cbc** } * In FIPS mode: **encryption**{**aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256**} *	By default, an IKEv2 proposal does not have any encryption algorithms.
4. Specify the integrity protection algorithms.	In non-FIPS mode: **integrity**{ **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512**} * In FIPS mode: **integrity**{ **sha1** | **sha256** | **sha384** | **sha512**} *	By default, an IKEv2 proposal does not have any integrity protection algorithms.
5. Specify the PRF algorithms.	In non-FIPS mode: **prf** { **aes-xcbc-mac** | **md5** | **sha1** | **sha256**| **sha384** |**sha512**} * In FIPS mode: **prf** { **sha1** | **sha256**| **sha384** | **sha512**} *	By default, an IKEv2 proposal uses the integrity protection algorithms as the PRF algorithms.

Step	Command	Remarks
6. Specify the DH groups.	In non-FIPS mode: **dh**{**group1** | **group14**| **group2** | **group24**| **group5** | **group19** | **group20**} * In FIPS mode: **dh**{**group14** | **group19** | **group20**} *	By default, an IKEv2 proposal does not have any DH groups.

Configuring an IKEv2 keychain

An IKEv2 keychain specifies the pre-shared keys used for IKEv2 negotiation.

An IKEv2 keychaincanhave multiple IKEv2 peers. Each peer has a symmetric pre-shared key or an asymmetric pre-shared key pair, and information for identifying the peer (such as the peer's host name, IP address or address range, or ID).

An IKEv2 negotiation initiator uses the peer host name or IP address/address range as the matching criterion to search for a peer. A responder uses the peer host IP address/address range or ID as the matching criterion to search for a peer.

To configure an IKEv2 keychain:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IKEv2 keychain and enter IKEv2 keychain view.	**ikev2 keychain***keychain-name*	By default, no IKEv2 keychains exist.
3. Create an IKEv2 peer and enter IKEv2 peer view.	**peer***name*	By default, no IKEv2 peers exist.
4. Configure the information for identifying the IKEv2 peer.	• To configure a host name for the peer: **hostname***host-name* • To configure a host IP address or address range for the peer: **address** { *ipv4-address* [*mask* | *mask-length*] | **ipv6***ipv6-address* [*prefix-length*] } • To configure an ID for the peer: **identity**{ **address** { *ipv4-address* | **ipv6** { *ipv6-address* } } | **fqdn***fqdn-name* | **email***email-string* | **key-id***key-id-string*}	By default, no hostname, host IP address, address range, or identity information is configured for an IKEv2 peer. You must configure different IP addresses/address ranges for different peers.
5. Configure a pre-shared key for the peer.	**pre-shared-key** [**local** | **remote**] {**ciphertext**| **plaintext**} *string*	By default, an IKEv2 peer does not have a pre-shared key.

Configure global IKEv2 parameters

Enabling the cookie challenging feature

Enable cookie challenging on responders to protect them against DoS attacks that use a large number of source IP addresses to forge IKE_SA_INIT requests.

To enable cookie challenging:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enable cookie challenging.	**ikev2 cookie-challenge***number*	By default, IKEv2 cookie challenging is disabled..

Configuring the IKEv2 DPDfeature

IKEv2 DPD detects dead IKEv2 peers in periodic or on-demand mode.

- **Periodic IKEv2 DPD**—Verifies the liveness of an IKEv2 peer by sending DPD messages at regular intervals.
- **On-demand IKEv2 DPD**—Verifies the liveness of an IKEv2 peer by sending DPD messages before sending data.
  - Before the device sends data, it identifies the time interval for which the last IPsec packet has been received from the peer. If the time interval exceeds the DPD interval, it sends a DPD message to the peer to detect its liveliness.
  - If the device has no data to send, it never sends DPD messages.

If you configure IKEv2 DPD in both IKEv2 profile view and system view, the IKEv2 DPD settings in IKEv2 profile view apply. If you do not configure IKEv2 DPD in IKEv2 profile view, the IKEv2 DPD settings in system view apply.

To configure global IKEv2 DPD:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Configure global IKEv2 DPD.	**ikev2 dpd interval***interval*[**retry***seconds*] { **on-demand** | **periodic** }	By default, global DPD is disabled.

Configuring the IKEv2 NAT keepalive feature

Configure this feature on the IKEv2 gateway behind the NAT device.The gateway then sends NAT keepalive packets regularly to its peer to keep the NAT session alive, so that the peer can access the device.

The NAT keepalive interval must be shorter than the NAT session lifetime.

This feature takes effect after the device detects the NAT device.

To configure the IKEv2 NAT keepalive feature:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A

Step	Command	Remarks
2. Set the IKEv2 NAT keepalive interval.	**ikev2 nat-keepalive** *seconds*	By default, the IKEv2 NAT keepalive interval is 10 seconds.

Configuring IKEv2 address pools

To perform centralized management on remote users, an IPsec gateway can use an address pool to assign private IP addresses to remote users.

You must use an IKEv2 address pool together with AAA authorization by specifying the IKEv2 address pool as an AAA authorization attribute. For more information about AAA authorization, see "Configuring AAA."

To configure IKEv2 address pools:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Configure an IKEv2 IPv4 address pool.	**ikev2 address-group***group-name**start-ipv4-address**end-ipv4-address* [*mask* | *mask-length*]	By default, no IKEv2 IPv4 address pool exists.
3. Configure an IKEv2 IPv6 address pool.	**ikev2 ipv6-address-group***group-name***prefix***prefix/prefix-len* **assign-len***assign-len*	By default, no IKEv2 IPv6 address pool exists.

Displaying and maintaining IKEv2

Execute**display** commands in any view and **reset** commands in user view.

Task	Command
Display the IKEv2 proposal configuration.	**display ikev2 proposal**[*name* | **default**]
Display the IKEv2 policy configuration.	**display ikev2 policy** [*policy-name* | **default**]
Display the IKEv2 profile configuration.	**display ikev2 profile** [*profile-name*]
Display the IKEv2 SA information.	**display ikev2 sa** [{ **local** | **remote** } { *ipv4-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*]] [**verbose**[**tunnel***tunnel-id*]]
Delete IKEv2 SAs and the child SAs negotiated through the IKEv2 SAs.	**reset ikev2 sa**[[{ **local** | **remote** } { *ipv4-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*]] | **tunnel***tunnel-id*] [**fast**]

IKEv2 configuration examples

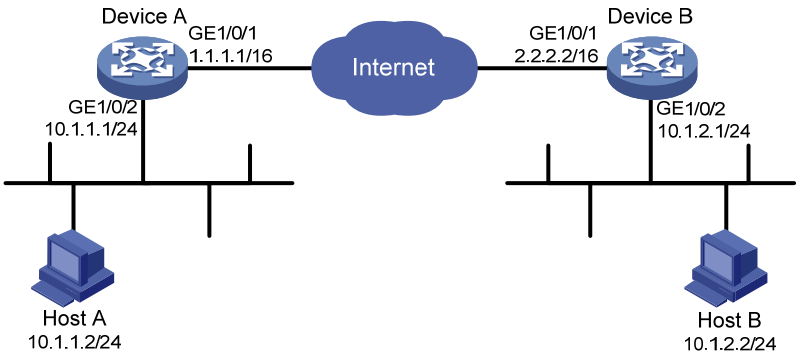
IKEv2 with pre-shared key authentication configuration example

Network requirements

As shown in Figure 2, configure an IKE-based IPsec tunnel between Device A and Deice B to securethe communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKEv2 proposal and the default IKEv2 policy in IKEv2 negotiation to set up IPsec SAs.
- Configure the two devicesto use the pre-shared key authentication method in IKEv2 negotiation.

Figure 2 Network diagram



Configuration procedures

1. Configure Device A:

Assign an IP address to each interface. (Details not shown.)

#ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

```
<DeviceA>system-view
[DeviceA] acl advanced 3101
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

#Create an IPsec transform set named**tran1**.

```
[DeviceA] ipsec transform-settran1
```

#Set the packet encapsulation mode to **tunnel**.

```
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
```

#Use the ESP protocol for the IPsec transform set.

```
[DeviceA-ipsec-transform-set-tran1] protocolesp
```

#Specify the encryption and authenticationalgorithms.

```
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithmsha1
[DeviceA-ipsec-transform-set-tran1] quit
```

#Create an IKEv2 keychain named**keychain1**.

```
[DeviceA] ikev2 keychain keychain1
```

Create an IKEv2 peer named **peer1**.

```
[DeviceA-ikev2-keychain-keychain1] peer peer1
```

Specify the peer IP address 2.2.2.2/24.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 24
```

Specify the peer ID, which is the IP address 2.2.2.2.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
```

Specify the plaintext **abcde**as the pre-shared key to be used with the peer at 2.2.2.2.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1]pre-shared-key plaintext abcde
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
```

```
[DeviceA-ikev2-keychain-keychain1] quit
```

#Create an IKEv2 profile named**profile1**.

```
[DeviceA] ikev2 profile profile1
```

Specify the local authentication method as pre-shared key.

```
[DeviceA-ikev2-profile-profile1] authentication-method local pre-share
```

Specify the remote authentication method as pre-shared key.

```
[DeviceA-ikev2-profile-profile1] authentication-method remote pre-share
```

#Specify the IKEv2 keychain **keychain1**.

```
[DeviceA-ikev2-profile-profile1] keychain keychain1
```

Specify the peer ID that the IKEv2 profile matches. The peer ID is the IP address 2.2.2.2/24.

```
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.255.0
[DeviceA-ikev2-profile-profile1] quit
```

Create an IKE-based IPsec policy entry with thename **map1**and the sequence number 10.

```
[DeviceA] ipsec policy map1 10 isakmp
```

#Specify the remote IP address 2.2.2.2 for the IPsec tunnel.

```
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceA-ipsec-policy-isakmp-map1-10] security acl3101
```

Specify the IPsec transform set **tran1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-map1-10] transform-settran1
```

Specify the IKEv2 profile **profile1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-map1-10] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

#Apply the IPsec policy **map1** to interface GigabitEthernet 1/0/1.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1]quit
```

#Configure a static route to the subnet where Host B resides.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 2.2.2.2
```

2. Configure Device B:

Assign an IP address to each interface. (Details not shown.)

#ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
<DeviceB>system-view
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceB-acl-ipv4-adv-3101] quit
```

#Create an IPsec transform set named**tran1**.

```
[DeviceB] ipsec transform-settran1
```

#Set the packet encapsulation mode to **tunnel**.

```
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
```

#Use the ESP protocol for the IPsec transform set.

```
[DeviceB-ipsec-transform-set-tran1] protocolesp
```

#Specify the encryption and authenticationalgorithms.

```
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1]esp authentication-algorithmsha1
```

```
[DeviceB-ipsec-transform-set-tran1] quit
```

Create an IKEv2 keychain named**keychain1**.

```
[DeviceB]ikev2 keychain keychain1
```

Create an IKEv2 peer named **peer1**.

```
[DeviceB-ikev2-keychain-keychain1] peer peer1
```

Specify the peer IP address 1.1.1.1/24.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1] address 1.1.1.1 24
```

Specify the peer ID, which is the IP address 1.1.1.1.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 1.1.1.1
```

Specify the plaintext **abcde**as the pre-shared key to be used with the peer at 1.1.1.1.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1]pre-shared-key plaintext abcde
[DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceB-ikev2-keychain-keychain1] quit
```

Create an IKEv2 profile named**profile1**.

```
[DeviceB] ikev2 profile profile1
```

Specify the local authentication method as pre-shared key.

```
[DeviceB-ikev2-profile-profile1] authentication-method local pre-share
```

Specify the remote authentication method as pre-shared key.

```
[DeviceB-ikev2-profile-profile1] authentication-method remote pre-share
```

Specify the IKEv2 keychain **keychain1**.

```
[DeviceB-ikev2-profile-profile1] keychain keychain1
```

Specify the peer ID that the IKEv2 profile matches. The peer ID is the IP address 1.1.1.1/24.

```
[DeviceA-ikev2-profile-profile1] match remote identity address 1.1.1.1 255.255.255.0
[DeviceA-ikev2-profile-profile1] quit
```

Create an IKE-based IPsec policy entry with the name **use1** and the sequence number 10.

```
[DeviceB] ipsec policy use1 10 isakmp
```

#Specify the remote IP address 1.1.1.1 for the IPsec tunnel.

```
[DeviceB-ipsec-policy-isakmp-use1-10] remote-address 1.1.1.1
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceB-ipsec-policy-isakmp-use1-10]security acl3101
```

#Specify the IPsec transform set **tran1** for the IPsec policy.

```
[DeviceB-ipsec-policy-isakmp-use1-10] transform-settran1
```

Specify the IKEv2 profile **profile1** for the IPsec policy.

```
[DeviceB-ipsec-policy-isakmp-use1-10] ikev2-profile profile1
[DeviceB-ipsec-policy-isakmp-use1-10] quit
```

#Apply the IPsec policy **use1** to interface GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/1] quit
```

#Configure a static route to the subnet where Host A resides.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 1.1.1.1
```

Verifying the configuration

Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec protected.

Display the IKEv2 proposal and IKEv2 policy on Device A.

```
[DeviceA]display ikev2 proposal
IKEv2 proposal : default
Encryption : 3DES-CBC AES-CBC-128
Integrity : MD596 SHA96
PRF : MD5 SHA1
  DH Group : Group2 Group5
[DeviceA]display ikev2policy
IKEv2 policy : default
  Match VPN instance : any
```

Display the IKEv2 SA on Device A.

```
[DeviceA]display ikev2sa
Tunnel ID   Local                       Remote                      Status
------------------------------------------------------------------------------
1  1.1.1.1/500        2.2.2.2/500         EST
Status:
IN-NEGO: Negotiating, EST: Establish, DEL:Deleting
# Display the IPsec SAs on Device A.
[DeviceA]display ipsec sa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  -----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  -----------------------------
   Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1456
    Tunnel:
local  address: 1.1.1.1
remote address: 2.2.2.2
    Flow:
sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: IP
dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: IP

    [Inbound ESP SAs]
      SPI: 3264152513 (0xc28f03c1)
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: active
```

```
[Outbound ESP SAs]
  SPI: 738451674 (0x2c03e0da)
  Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
  SA duration (kilobytes/sec): 1843200/3600
  SA remaining duration (kilobytes/sec): 1843200/3484
  Max received sequence-number:
  Anti-replay check enable: Y
  Anti-replay window size: 64
  UDP encapsulation used for NAT traversal: N
  Status: active
```

Display the IKEv2 proposal, IKEv2 policy, IKEv2 SA and IPsec SAs on Device B.

```
[DeviceB]display ikev2 proposal
[DeviceB]display ikev2policy
[DeviceB]display ikev2sa
[DeviceB]display ipsec sa
```

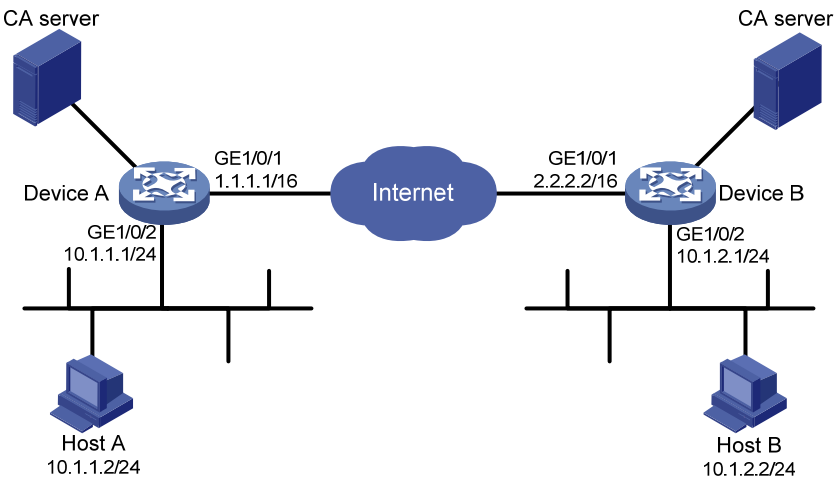
IKEv2 with RSA signature authentication configuration example

Network requirements

As shown in Figure 3, configure an IKE-based IPsec tunnel between Device A and Deice B to securethe communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

Configure Device A and Device B to use IKEv2 negotiation and RSA signature authentication. Device A acts as the initiator and the subnet where Device A resides uses IP addresses dynamically allocated.

Figure 3 Network diagram



Configuration procedure

1. Configure Device A:

   # Assign an IP address to each interface. (Details not shown.)

   #ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

   ```
   <DeviceA>system-view
   [DeviceA] acl advanced 3101
   ```

```
[DeviceA-acl-ipv4-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination
10.1.2.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

#Create an IPsec transform set named**tran1**.

```
[DeviceA] ipsec transform-settran1
```

#Set the packet encapsulation mode to **tunnel**.

```
[DeviceA-ipsec-transform-set-tran1] encapsulation-mode tunnel
```

#Use the ESP protocol for the IPsec transform set.

```
[DeviceA-ipsec-transform-set-tran1] protocolesp
```

#Specify the encryption and authenticationalgorithms.

```
[DeviceA-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceA-ipsec-transform-set-tran1] esp authentication-algorithmsha1
[DeviceA-ipsec-transform-set-tran1] quit
```

#Create a PKI entity named**entity1**.

```
[DeviceA]pki entityentity1
```

Set the common name to **routera** for the PKI entity.

```
[DeviceA-pki-entity-entity1] common-name routera
[DeviceA-pki-entity-entity1] quit
```

Create a PKI domain named**domain1**.

```
[DeviceA]pki domain domain1
```

#Set the certificate request mode to **auto** and set the password to **123**for certificate revocation.

```
[DeviceA-pki-domain-domain1] certificate request mode auto password simple 123
```

Set an MD5 fingerprint for verifying the validity of the CA root certificate.

```
[DeviceA-pki-domain-domain1] root-certificate fingerprint md5
50c7a2d282ea710a449eede6c56b102e
```

#Specify the trusted CA 8088.

```
[DeviceA-pki-domain-domain1] ca identifier 8088
```

Specify the URL of the registration server for certificate request through the SCEP protocol. This example uses the URL of http://192.168.222.1:446/eadbf9af4f2c4641e685f7a6021e7b298373feb7.

```
[DeviceA-pki-domain-domain1] certificate request url
http://192.168.222.1:446/eadbf9af4f2c4641e685f7a6021e7b298373feb7
```

#Specify the CAto accept certificate requests.

```
[DeviceA-pki-domain-domain1] certificate request from ca
```

Specify the PKI entity for certificate request as**entity1**.

```
[DeviceA-pki-domain-domain1] certificate request entity entity1
```

Specify the RSA key pair**rsa1**with the **general**purposefor certificate request.

```
[DeviceA-pki-domain-domain1] public-key rsa general name rsa1
[DeviceA-pki-domain-domain1] quit
```

#Create an IKEv2 profile named**profile1**.

```
[DeviceA] ikev2 profile profile1
```

Specify the local authentication method as RSA signatures.

```
[DeviceA-ikev2-profile-profile1] authentication-method local rsa-signature
```

Specify the remote authentication method as RSA signatures.

```
[DeviceA-ikev2-profile-profile1] authentication-method remote rsa-signature
```

#Specify the PKI domain **domain1** for the IKEv2 profile.

```
[DeviceA-ikev2-profile-profile1]certificate domain domain1
```

Set the local ID to the FQDN name **www.routera.com**.

```
[DeviceA-ikev2-profile-profile1] identitylocalfqdn www.routera.com
```

Specify the peer ID that the IKEv2 profile matches. The peer ID is the FQDN name **www.routerb.com**.

```
[DeviceA-ikev2-profile-profile1] match remote identity fqdn www.routerb.com
[DeviceA-ikev2-profile-profile1] quit
```

Create an IKEv2 proposal named **10**.

```
[DeviceA] ikev2proposal 10
```

#Specify the integrity protection algorithm as HMAC-MD5.

```
[DeviceA-ikev2-proposal-10] integrity md5
```

#Specify the encryption algorithm as 3DES-CBC.

```
[DeviceA-ikev2-proposal-10] encryption 3des-cbc
```

Specify the DH group as Group 1.

```
[DeviceA-ikev2-proposal-10] dh group1
```

#Specify the PRF algorithm as HMAC-MD5.

```
[DeviceA-ikev2-proposal-10] prf md5
[DeviceA-ikev2-proposal-10] quit
```

Create an IKEv2 policy named **1**.

```
[DeviceA] ikev2 policy 1
```

Specify the IKEv2 proposal **10** for the IKEv2 policy.

```
[DeviceA-ikev2-policy-1] proposal 10
[DeviceA-ikev2-policy-1] quit
```

Create an IKE-based IPsec policy entry with the name **map1** and the sequence number 10.

```
[DeviceA] ipsec policy map1 10 isakmp
```

#Specify the remote IP address 2.2.2.2 for the IPsec tunnel.

```
[DeviceA-ipsec-policy-isakmp-map1-10] remote-address 2.2.2.2
```

#Specify the IPsec transform set **tran1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-map1-10] transform-settran1
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceA-ipsec-policy-isakmp-map1-10] security acl3101
```

Specify the IKEv2 profile **profile1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-map1-10] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-map1-10] quit
```

#Apply the IPsec policy **map1** to interface GigabitEthernet 1/0/1.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy map1
[DeviceA-GigabitEthernet1/0/1]quit
```

#Configure a static route to the subnet where Host B resides.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 2.2.2.2
```

2. Configure Device B:

Assign an IP address to each interface. (Details not shown.)

#ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
<DeviceB>system-view
[DeviceB] acl advanced 3101
[DeviceB-acl-ipv4-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
```

```
[DeviceB-acl-ipv4-adv-3101] quit
```

#Create an IPsec transform set named**tran1**.

```
[DeviceB] ipsec transform-settran1
```

#Set the packet encapsulation mode to **tunnel**.

```
[DeviceB-ipsec-transform-set-tran1] encapsulation-mode tunnel
```

#Use the ESP protocol for the IPsec transform set.

```
[DeviceB-ipsec-transform-set-tran1] protocolesp
```

#Specify the encryption and authenticationalgorithms.

```
[DeviceB-ipsec-transform-set-tran1] esp encryption-algorithm des-cbc
[DeviceB-ipsec-transform-set-tran1]esp authentication-algorithmsha1
[DeviceB-ipsec-transform-set-tran1] quit
```

#Create a PKI entity named**entity2**.

```
[DeviceB]pki entityentity2
```

Set the common name to **routerb** for the PKI entity.

```
[DeviceB-pki-entity-entity2] common-name routerb
[DeviceB-pki-entity-entity2] quit
```

#Create a PKI domain named**domain2**.

```
[DeviceB] pki domain domain2
```

#Set the certificate request mode to **auto** and set the password to **123**for certificate revocation.

```
[DeviceB-pki-domain-domain2]certificate request mode auto password simple 123
```

Set an MD5 fingerprint for verifying the validity of the CA root certificate.

```
[DeviceB-pki-domain-domain2] root-certificate fingerprint md5
50c7a2d282ea710a449eede6c56b102e
```

#Specify the trusted CA 8088.

```
[DeviceB-pki-domain-domain2] ca identifier 8088
```

Specify the URL of the registration server for certificate request through the SCEP protocol. This example uses the URL of http://192.168.222.1:446/eadbf9af4f2c4641e685f7a6021e7b298373feb7.

```
[DeviceB-pki-domain-domain2] certificate request url
http://192.168.222.1:446/eadbf9af4f2c4641e685f7a6021e7b298373feb7
```

#Specify the CAto accept certificate requests.

```
[DeviceB-pki-domain-domain2] certificate request from ca
```

Specify the PKI entity for certificate request as**entity2**.

```
[DeviceB-pki-domain-domain2] certificate request entity entity2
```

Specify the RSA key pair**rsa1**with the **general**purposefor certificate request.

```
[DeviceB-pki-domain-domain2] public-key rsa general name rsa1
[DeviceB-pki-domain-domain2] quit
```

Create an IKEv2 profile named**profile2**.

```
[DeviceB] ikev2 profile profile2
```

Specify the local authentication method as RSA signatures.

```
[DeviceB-ikev2-profile-profile2] authentication-method local rsa-signature
```

Specify the remote authentication method as RSA signatures.

```
[DeviceB-ikev2-profile-profile2] authentication-method remote rsa-signature
```

#Set the local identity to the FQDN name **www.routerb.com**.

```
[DeviceB-ikev2-profile-profile2] identity local fqdn www.routerb.com
```

#Specify the peer ID that the IKEv2 profile matches. The peer ID is the FQDN name **www.routera.com**.

```
[DeviceB-ikev2-profile-profile2] match remote identity fqdn www.routera.com
[DeviceB-ikev2-profile-profile2] quit
```

#Create an IKEv2 proposal named**10**.

```
[DeviceB] ikev2proposal 10
```

#Specify the integrity protection algorithm as HMAC-MD5.

```
[DeviceB-ikev2-proposal-10] integrity md5
```

#Specify the encryption algorithm as 3DES-CBC.

```
[DeviceB-ikev2-proposal-10] encryption 3des-cbc
```

Specify the DH group as Group 1.

```
[DeviceB-ikev2-proposal-10] dh group1
```

#Specify the PRF algorithm as HMAC-MD5.

```
[DeviceB-ikev2-proposal-10] prf md5
[DeviceB-ikev2-proposal-10] quit
```

Create an IKEv2 policy named **1**.

```
[DeviceB] ikev2 policy 1
```

Specify the IKEv2 proposal **10** for the IKEv2 policy.

```
[DeviceB-ikev2-policy-1] proposal 10
[DeviceB-ikev2-policy-1] quit
```

Create an IPsec policy template entry with the name **template1** and the sequence number 1.

```
[DeviceB] ipsec policy-template template1 1
```

#Specify the remote IP address 1.1.1.1 for the IPsec tunnel.

```
[DeviceB-ipsec-policy-template-template1-1] remote-address 1.1.1.1
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceB-ipsec-policy-template-template1-1]security acl3101
```

#Specify the IPsec transform set **tran1** for the IPsec policy template.

```
[DeviceB-ipsec-policy-template-template1-1] transform-set tran1
```

Specify the IKEv2 profile **profile2** for the IPsec policy template.

```
[DeviceB-ipsec-policy-template-template1-1] ikev2-profile profile2
[DeviceB-ipsec-policy-template-template1-1] quit
```

Create an IKE-based IPsec policy entry with thename **use1** and the sequence number 1 by using the IPsec policy template **template1**.

```
[DeviceB] ipsec policy use1 1 isakmp template template1
```

#Apply IPsec policy **use1** to interface GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy use1
[DeviceB-GigabitEthernet1/0/1] quit
```

#Configure a static route to the subnet where Host A resides.

```
[DeviceB] ip route-static 10.1.1.0 255.255.255.0 1.1.1.1
```

Verifying the configuration

Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec protected.

Display the IKEv2 proposal configuration on Device A and Device B.

```
[DeviceA]display ikev2 proposal 10
IKEv2 proposal : 10
Encryption : 3DES-CBC
```

```
Integrity : MD596
PRF : MD5
  DH Group : Group1
[DeviceB]display ikev2 proposal 10
IKEv2 proposal : 10
Encryption : 3DES-CBC
Integrity : MD596
PRF : MD5
  DH Group : Group1
```

Display the IKEv2 policy configuration Device A and Device B.

```
[DeviceA]display ikev2policy 1
IKEv2 policy : 1
  Match Local :any
  Match VPN instance : public
Proposal : 1
[DeviceB]display ikev2policy 1
IKEv2 policy : 1
  Match Local :any
  Match VPN instance : public
Proposal : 1
```

Display the IKEv2 SA on Device A.

```
[DeviceA]display ikev2sa
Tunnel ID   Local                       Remote                      Status
------------------------------------------------------------------------------
1  1.1.1.1/500        2.2.2.2/500         EST
Status:
IN-NEGO: Negotiating, EST: Establish, DEL:Deleting
```

Display information about the CA certificate on Device A.

```
[DeviceA] display pki certificate domain domain1 ca
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            b9:14:fb:25:c9:08:2c:9d:f6:94:20:30:37:4e:00:00
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=cn, O=rnd, OU=sec, CN=8088
        Validity
            Not Before: Sep  6 01:53:58 2012 GMT
            Not After : Sep  8 01:50:58 2015 GMT
        Subject: C=cn, O=rnd, OU=sec, CN=8088
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:de:81:f4:42:c6:9f:c2:37:7b:21:84:57:d6:42:
                    00:69:1c:4c:34:a4:5e:bb:30:97:45:2b:5e:52:43:
                    c0:49:1f:e1:d8:0f:5c:48:c2:39:69:d1:84:e4:14:
                    70:3d:98:41:28:1c:20:a1:9a:3f:91:67:78:77:27:
```

```
                    d9:08:5f:7a:c4:36:45:8b:f9:7b:e7:7d:6a:98:bb:
                    4e:a1:cb:2c:3d:92:66:bd:fb:80:35:16:c6:35:f0:
ff:0b:b9:3c:f3:09:94:b7:d3:6f:50:8d:83:f1:66:
                    2f:91:0b:77:a5:98:22:b4:77:ac:84:1d:03:8e:33:
                    1b:31:03:78:4f:77:a0:db:af
                Exponent: 65537 (0x10001)
    Signature Algorithm: sha1WithRSAEncryption
        9a:6d:8c:46:d3:18:8a:00:ce:12:ee:2b:b0:aa:39:5d:3f:90:
        08:49:b9:a9:8f:0d:6e:7b:e1:00:fb:41:f5:d4:0c:e4:56:d8:
        7a:a7:61:1d:2b:b6:72:e3:09:0b:13:9d:fa:c8:fc:c4:65:a7:
        f9:45:21:05:75:2c:bf:36:7b:48:b4:4a:b9:fe:87:b9:d8:cf:
        55:16:87:ec:07:1d:55:5a:89:74:73:68:5e:f9:1d:30:55:d9:
        8a:8f:c5:d4:20:7e:41:a9:37:57:ed:8e:83:a7:80:2f:b8:31:
        57:3a:f2:1a:28:32:ea:ea:c5:9a:55:61:6a:bc:e5:6b:59:0d:
        82:16
```

Display the local certificate on Device A.

```
[DeviceA]display pki certificate domain domain1 local
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            a1:f4:d4:fd:cc:54:c3:07:c4:9e:15:2d:5f:64:57:77
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=cn, O=rnd, OU=sec, CN=8088
        Validity
            Not Before: Sep 26 02:06:43 2012 GMT
            Not After : Sep 26 02:06:43 2013 GMT
        Subject: CN=devicea
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (1024 bit)
                Modulus:
                    00:b0:a1:cd:24:6e:1a:1d:51:79:f0:2a:3e:9f:e9:
                    84:07:16:78:49:1b:7d:0b:22:f0:0a:ed:75:91:a4:
                    17:fd:c7:ef:d0:66:5c:aa:e3:2a:d9:71:12:e4:c6:
                    25:77:f0:1d:97:bb:92:a8:bd:66:f8:f8:e8:d5:0d:
                    d2:c8:01:dd:ea:e6:e0:80:ad:db:9d:c8:d9:5f:03:
                    2d:22:07:e3:ed:cc:88:1e:3f:0c:5e:b3:d8:0e:2d:
ea:d6:c6:47:23:6a:11:ef:3c:0f:6b:61:f0:ca:a1:
                    79:a0:b1:02:1a:ae:8c:c9:44:e0:cf:d1:30:de:4c:
                    f0:e5:62:e7:d0:81:5d:de:d3
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 CRL Distribution Points:

                Full Name:
                  URI:http://xx.rsa.com:447/8088.crl
```

```
    Signature Algorithm: sha1WithRSAEncryption
        73:ac:66:f9:b8:b5:39:e1:6a:17:e4:d0:72:3e:26:9e:12:61:
        9e:c9:7a:86:6f:27:b0:b9:a3:5d:02:d9:5a:cb:79:0a:12:2e:
cb:e7:24:57:e6:d9:77:12:6b:7a:cf:ee:d6:17:c5:5f:d2:98:
        30:e0:ef:00:39:4a:da:ff:1c:29:bb:2a:5b:60:e9:33:8f:78:
        f9:15:dc:a5:a3:09:66:32:ce:36:cd:f0:fe:2f:67:e5:72:e5:
        21:62:85:c4:07:92:c8:f1:d3:13:9c:2e:42:c1:5f:0e:8f:ff:
        65:fb:de:7c:ed:53:ab:14:7a:cf:69:f2:42:a4:44:7c:6e:90:
        7e:cd
```

Display the IPsec SAs on Device A.

```
[DeviceA] display ipsec sa
------------------------------
Interface: GigabitEthernet1/0/1
------------------------------

  ----------------------------
  IPsec policy: map1
  Sequence number: 10
  Mode: ISAKMP
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1456
    Tunnel:
local  address: 1.1.1.1
remote address: 2.2.2.2
    Flow:
sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: ip
dest addr: 10.1.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3264152513 (0xc28f03c1)
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: active

    [Outbound ESP SAs]
      SPI: 738451674 (0x2c03e0da)
      Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/3484
      Max received sequence-number:
```

```
Anti-replay check enable: Y
Anti-replay window size: 64
UDP encapsulation used for NAT traversal: N
Status: active
```

Display the information about the CA certificate, local certificate, IKEv2 SA, and IPsec SA on Device B.

```
[DeviceB]display ikev2sa
[DeviceB] display pki certificate domain domain2 ca
[DeviceB] display pki certificate domain domain2 local
[DeviceB] display ipsec sa
```

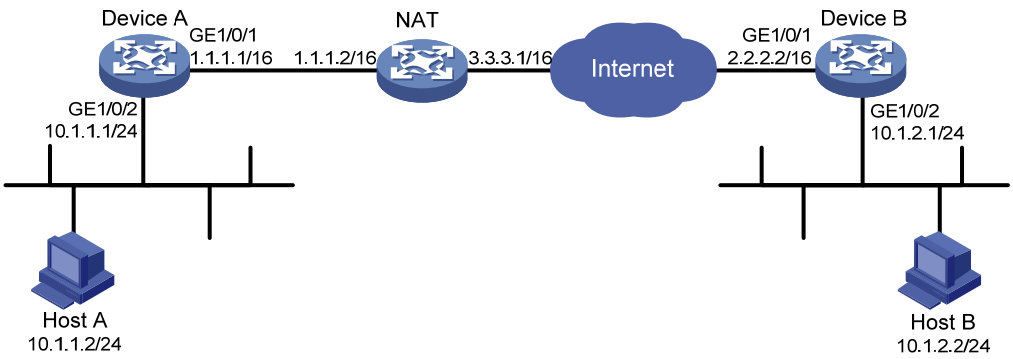
IKEv2 with NAT traversal configuration example

Network requirements

As shown in Figure 4, Device A is behind the NAT device. Configure an IKE-based IPsec tunnel between Device A and Deice B to securethe communication between subnet 10.1.1.0/24 and subnet 10.1.2.0/24.

- Configure Device A and Device B to use the default IKEv2 proposal and the default IKEv2 policy in IKEv2 negotiation to set up IPsec SAs.
- Configure the two devicesto use the pre-shared key authentication method in IKEv2 negotiation.

Figure 4 Network diagram



Configuration procedure

1. Configure Device A:

   # Assign an IP address to each interface. (Details not shown.)

   #ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.1.0/24 to subnet 10.1.2.0/24.

   ```
   <DeviceA>system-view
   [DeviceA] acl advanced 3101
   [DeviceA-acl-ipv4-adv-3101] rule 0 permit ip source 10.1.1.0 0.0.0.255 destination
   10.1.2.0 0.0.0.255
   [DeviceA-acl-ipv4-adv-3101] quit
   ```

   #Create an IPsec transform set named**transform1**.

   ```
   [DeviceA] ipsec transform-set transform1
   ```

   #Use the ESP protocol for the IPsec transform set.

   ```
   [DeviceA-ipsec-transform-set-transform1] protocolesp
   ```

   #Specify the encryption and authenticationalgorithms.

```
[DeviceA-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceA-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceA-ipsec-transform-set-transform1] quit
```

#Create an IKEv2 keychain named**keychain1**.

```
[DeviceA]ikev2 keychain keychain1
```

Create an IKEv2 peer named **peer1**.

```
[DeviceA-ikev2-keychain-keychain1] peer peer1
```

Specify the peer IP address 2.2.2.2/24.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1] address 2.2.2.2 24
```

Specify the peer ID, which is the IP address 2.2.2.2.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1] identity address 2.2.2.2
```

Specify the plaintext **123**as the pre-shared key to be used with the peer.

```
[DeviceA-ikev2-keychain-keychain1-peer-peer1]pre-shared-key plaintext 123
[DeviceA-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceA-ikev2-keychain-keychain1] quit
```

#Create an IKEv2 profile named**profile1**.

```
[DeviceA] ikev2 profile profile1
```

#Specify the IKEv2 keychain **keychain1**.

```
[DeviceA-ikev2-profile-profile1] keychain keychain1
```

#Set the local ID to the FQDN name **www.devicea.com**.

```
[DeviceA-ikev2-profile-profile1]identitylocal fqdn www.devicea.com
```

Specify the peer ID that the IKEv2 profile matches. The peer ID is the IP address 2.2.2.2/24.

```
[DeviceA-ikev2-profile-profile1] match remote identity address 2.2.2.2 255.255.255.0
[DeviceA-ikev2-profile-profile1] quit
```

Create an IKE-based IPsec policy entry with thename **policy1** and the sequence number 1.

```
[DeviceA] ipsec policy policy1 1 isakmp
```

#Specify the remote IP address 2.2.2.2 for the IPsec tunnel.

```
[DeviceA-ipsec-policy-isakmp-policy1-1] remote-address 2.2.2.2
```

#Specify the IPsec transform set **transform1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-policy1-1] transform-set transform1
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceA-ipsec-policy-isakmp-policy1-1] security acl3101
```

Specify the IKEv2 profile **profile1** for the IPsec policy.

```
[DeviceA-ipsec-policy-isakmp-policy1-1] ikev2-profile profile1
[DeviceA-ipsec-policy-isakmp-policy1-1] quit
```

#Apply the IPsec policy **policy1** to interface GigabitEthernet 1/0/1.

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceA-GigabitEthernet1/0/1]quit
```

Configure a static route to the subnet where Host B resides.

```
[DeviceA] ip route-static 10.1.2.0 255.255.255.0 2.2.2.2
```

2. Configure Device B:

Assign an IP address to each interface. (Details not shown.)

#ConfigureIPv4 advanced ACL 3101 to identify traffic from subnet 10.1.2.0/24 to subnet 10.1.1.0/24.

```
<DeviceA>system-view
[DeviceA] acl advanced 3101
```

```
[DeviceA-acl-ipv4-adv-3101] rule 0 permit ip source 10.1.2.0 0.0.0.255 destination
10.1.1.0 0.0.0.255
[DeviceA-acl-ipv4-adv-3101] quit
```

#Create an IPsec transform set named **transform1**.

```
<DeviceB> system-view
[DeviceB] ipsec transform-set transform1
```

#Use the ESP protocol for the IPsec transform set.

```
[DeviceB-ipsec-transform-set-transform1] protocolesp
```

#Specify the encryption and authenticationalgorithms.

```
[DeviceB-ipsec-transform-set-transform1] esp encryption-algorithm 3des-cbc
[DeviceB-ipsec-transform-set-transform1] esp authentication-algorithm md5
[DeviceB-ipsec-transform-set-transform1] quit
```

#Create an IKEv2 keychain named **keychain1**.

```
[DeviceB]ikev2 keychain keychain1
```

Create an IKEv2 peer named **peer1**.

```
[DeviceB-ikev2-keychain-keychain1] peer peer1
```

Specify the peer IP address 1.1.1.1/24.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1] address 1.1.1.1 24
```

Specify the peer ID, which is the IP address 1.1.1.1.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1] identity address 1.1.1.1
```

Specify the plaintext **123**as the pre-shared key to be used with the peer.

```
[DeviceB-ikev2-keychain-keychain1-peer-peer1]pre-shared-key plaintext 123
[DeviceB-ikev2-keychain-keychain1-peer-peer1] quit
[DeviceB-ikev2-keychain-keychain1] quit
```

#Create an IKEv2 profile named**profile1**.

```
[DeviceB] ikev2 profile profile1
```

Specify the IKEv2 keychain **keychain1**.

```
[DeviceB-ikev2-profile-profile1] keychain keychain1
```

Specify the peer ID that the IKEv2 profile matches. The peer ID is the FQDN name **www.devicea.com**.

```
[DeviceB-ikev2-profile-profile1] match remote identity fqdn www.devicea.com
[DeviceB-ikev2-profile-profile1] quit
```

Create an IPsec policy template entry with the name **template1** and the sequence number 1.

```
[DeviceB] ipsec policy-template template1 1
```

#Specify the remote IP address 1.1.1.1 for the IPsec tunnel.

```
[DeviceB-ipsec-policy-template-template1-1] remote-address 1.1.1.1
```

#Specify ACL 3101 to identify the traffic to be protected.

```
[DeviceB-ipsec-policy-template-template1-1]security acl3101
```

#Specify the IPsec transform set **transform1** for the IPsec policy template.

```
[DeviceB-ipsec-policy-template-template1-1] transform-set transform1
```

Specify the IKEv2 profile **profile1** for the IPsec policy template.

```
[DeviceB-ipsec-policy-template-template1-1] ikev2-profile profile1
[DeviceB-ipsec-policy-template-template1-1] quit
```

Create an IKE-based IPsec policy entry with the name **policy1** and the sequence number 1 byusing the IPsec policy template **template1**.

```
[DeviceB] ipsec policy policy1 1 isakmp template template1
```

#Apply the IPsec policy **policy1** to interface GigabitEthernet 1/0/1.

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] ipsec apply policy policy1
[DeviceB-GigabitEthernet1/0/1]quit
```

Configure a static route to the subnet where Host A resides. The next hop is the IP address of the output interface on the NAT device. (Details not shown.)

Verifying the configuration

Initiate a connection from subnet 10.1.1.0/24 to subnet 10.1.2.0/24 to trigger IKEv2 negotiation. After IPsec SAs are successfully negotiated by IKEv2, traffic between the two subnets is IPsec protected.

Display the IKEv2 SA on Device A.

```
[DeviceA]display ikev2sa
Tunnel ID   Local                       Remote                      Status
------------------------------------------------------------------------------
1  1.1.1.1/500        2.2.2.2/500         EST
Status:
IN-NEGO: Negotiating, EST: Establish, DEL:Deleting
 [DeviceA]display ikev2sa verbose
   -----------------------------------------------
   Connection ID: 13
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Initiator
   -----------------------------------------------
   Local IP: 1.1.1.1
   Local ID type: FQDN
   Local ID: www.devicea.com

   Remote IP: 2.2.2.2
   Remote ID type: IPV4_ADDR
   Remote ID: 2.2.2.2

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: MD5
   Encryption-algorithm: 3DES-CBC

   Life duration(sec): 86400
   Remaining key duration(sec): 84565
   Exchange-mode: Aggressive
   Diffie-Hellman group: Group 1
   NAT traversal: Detected
```

Display the IPsec SAs on Device A.

```
[DeviceA]displayipsecsa
-------------------------------
Interface: GigabitEthernet1/0/1
-------------------------------

  ----------------------------
```

```
  IPsec policy: policy1
  Sequence number: 1
  Mode: ISAKMP
  -----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1435
    Tunnel:
local  address: 1.1.1.1
remote address: 2.2.2.2
    Flow:
sour addr: 10.1.1.0/255.255.255.0  port: 0  protocol: IP
dest addr: 10.2.1.0/255.255.255.0  port: 0  protocol: IP

    [Inbound ESP SAs]
      SPI: 830667426 (0x3182faa2)
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for nat traversal: Y
      Status: active

    [Outbound ESP SAs]
      SPI: 3516214669 (0xd1952d8d)
      Transform set: ESP-ENCRYPT-3DES-CBC ESP-AUTH-MD5
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843200/2313
      Max received sequence-number:
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for nat traversal: Y
      Status: active
```

Troubleshooting IKEv2

IKEv2 negotiation failed because no matching IKEv2 proposals were found

Symptom

The IKEv2 SA is in IN-NEGO status.

```
<Sysname> display ikev2sa
Tunnel ID   Local                     Remote                    Status
  ------------------------------------------------------------------------------
  5           123.234.234.124/500         123.234.234.123/500         IN-NEGO
Status:
```

```
IN-NEGO: Negotiating, EST: Establish, DEL:Deleting
```

Analysis

Certain IKEv2 proposal settings are incorrect.

Solution

1. Examine the IKEv2 proposal configuration to see whether the two ends have matching IKEv2 proposals.
2. Modify the IKEv2 proposal configuration to make sure the two ends have matching IKEv2 proposals.

IPsec SA negotiation failed because no matching IPsec transform sets were found

Symptom

The **display ikev2sa**command shows that the IKEv2 SA negotiation succeeded and the IKEv2 SA is in EST status. The **display ipsec sa**command shows that the expected IPsec SAs have not been negotiated yet.

Analysis

Certain IPsec policy settings are incorrect.

Solution

1. Examine the IPsec configuration to see whether the two ends have matching IPsec transform sets.
2. Modify the IPsec configuration to make sure the two ends have matching IPsec transform sets.

IPsec tunnel establishment failed

Symptom

The ACLs and IKEv2 proposals are correctly configured on both ends. The two ends cannot establish an IPsec tunnel or cannot communicate through the established IPsec tunnel.

Analysis

The IKEv2 SA or IPsec SAs on either end are lost. The reason might be that the network is unstable and the device reboots.

Solution

1. Use the **display ikev2 sa**command to examine whether an IKEv2 SA exists on both ends. If the IKEv2 SA on one end is lost, delete the IKEv2 SA on the other end by using the **reset ikev2 sa**command and trigger new negotiation. If an IKEv2 SA exists on both ends, go to the next step.
2. Use the **display ipsec sa**command to examine whether IPsec SAs exist on both ends. If the IPsec SAs on one end are lost, delete the IPsec SAs on the other end by using the **reset ipsec sa**command and trigger new negotiation.

Command reference

aaa authorization

Use **aaa authorization** to enable IKEv2 AAA authorization.

Use **undo aaa authorization** to disable IKEv2 AAA authorization.

Syntax

aaa authorization domain*domain-name* **username***user-name*

undo aaa authorization

Default

IKEv2 AAA authorization is disabled.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

domain*domain-name*: Specifies the ISP domain used for requesting authorization attributes. The ISP domain name is a case-insensitive string of 1 to 255 characters and must meet the following requirements:

- The name cannot contain a forward slash (/), backslash (\), vertical bar (|), quotation mark ("), colon (:), asterisk (*), question mark (?), left angle bracket (<), right angle bracket (>), or an at sign (@).
- The name cannot be **d**, **de**, **def**, **defa**, **defau**, **defaul**, **default**, **i**, **if**, **if-**, **if-u**, **if-un**, **if-unk**, **if-unkn**, **if-unkno**, **if-unknow**, or **if-unknown**.

username*user-name*: Specifies the username used for requesting authorization attributes. The username is a case-sensitive string of 1 to 55 characters and must meet the following requirements:

- The username cannot contain the domain name.
- The username cannot contain a forward slash (/), backslash (\), vertical bar (|), colon (:), asterisk (*), question mark (?), left angle bracket (<), right angle bracket (>), or an at sign (@).
- The username cannot be **a**, **al**, or **all**.

Usage guidelines

The AAA authorization feature enables IKEv2 to request authorization attributes, such as the IKEv2 IPv4 address pool, from AAA.

IKEv2 uses the ISP domain and username to request authorization attributes. AAA uses the authorization settings in the ISP domain to request the user's authorization attributes from the remote AAA server or the local user database. After IKEv2 passes the username authentication, it obtains the authorization attributes.

This feature is applicable when AAA is used to centrally manage and deploy authorization attributes.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Enable AAA authorization. Specify the ISP domain name **abc** and the username **test**.

```
[Sysname-ikev2-profile-profile1] aaa authorization domain abc username test
```

Related commands

display ikev2 profile

address

Use **address** to specify the IP address or IP address range of an IKEv2 peer.

Use **undo address** to restore the default.

Syntax

address{*ipv4-address* [*mask* | *mask-length*] | **ipv6***ipv6-address* [*prefix-length*] }

undo address

Default

The IKEv2 peer's IP address or IP address range is not specified.

Views

IKEv2 peer view

Predefined user roles

network-admin

mdc-admin

Parameters

ipv4-address: Specifies the IPv4 address of the IKEv2 peer.

mask: Specifies the subnet mask of the IPv4 address.

mask-length: Specifies the subnet mask length of the IPv4 address, in the range of 0 to 32.

ipv6*ipv6-address*: Specifies the IPv6 address of the IKEv2 peer.

prefix-length: Specifies the prefix length of the IPv6 address, in the range of 0 to 128.

Usage guidelines

Both the initiator and the responder can look up an IKEv2 peer by IP address in IKEv2 negotiation.

The IP addresses of different IKEv2 peers in the same IKEv2 keychain cannot be the same.

Examples

Create an IKEv2 keychain named **key1**.

```
<Sysname> system-view
[Sysname] ikev2 keychain key1
```

Create an IKEv2 peer named **peer1**.

```
[Sysname-ikev2-keychain-key1] peer peer1
```

Specifythe IKEv2 peer's IP address 3.3.3.3 with the subnet mask 255.255.255.0.

```
[Sysname-ikev2-keychain-key1-peer-peer1] address3.3.3.3 255.255.255.0
```

Related commands

- **ikev2 keychain**
- **peer**

authentication-method

Use **authentication-method**to specify the local or remote identity authentication method.

Use **undo authentication-method**to remove the local or remote identity authentication method.

Syntax

authentication-method { **local** | **remote** } { **dsa-signature** | **ecdsa-signature** | **pre-share** | **rsa-signature** }

undo authentication-method local

undo authentication-method remote{ **dsa-signature** | **ecdsa-signature**| **pre-share** | **rsa-signature** }

Default

No local or remote identity authentication method is specified.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

local: Specifies the local identity authentication method.

remote: Specifies the remote identity authentication method.

dsa-signature: Specifies the DSA signatures as the identity authentication method.

ecdsa-signature: Specifies the ECDSA signatures as the identity authentication method.

pre-share: Specifies the pre-shared key as the identity authentication method.

rsa-signature: Specifies the RSA signatures as the identity authentication method.

Usage guidelines

The local and remote identity authentication methods must both be specified and they can be different.

You can specify only one local identity authentication method. You can specify multiple remote identity authentication methods by executing this command multiple times when there are multiple remote ends whose authentication methods are unknown.

If you use RSA, DSA, or ECDSA signature authentication, you must specify PKI domains for obtaining certificates. You can specify PKI domains by using the**certificate domain** command in IKEv2 profile view. If you do not specify PKI domains in IKEv2 profile view, the PKI domains configured by the **pki domain** command in system view will be used.

If you specify the pre-shared key method, you must specify a pre-shared key for the IKEv2 peer in the keychain used by the IKEv2 profile.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Specify the pre-shared key and RSA signatures as the local and remote authentication methods, respectively.

```
[Sysname-ikev2-profile-profile1] authentication local pre-share
[Sysname-ikev2-profile-profile1] authentication remote rsa-signature
```

Specify the PKI domain**genl** as the PKI domain for obtaining certificates.

```
[Sysname-ikev2-profile-profile1] certificate domain genl
```

Specify the keychain **keychain1**.

```
[Sysname-ikev2-profile-profile1] keychain keychain1
```

Related commands

- **display ikev2 profile**
- **certificate domain** (ikev2 profile view)
- **keychain** (ikev2 profile view)

certificate domain

Use **certificate domain** to specify a PKI domain for signature authentication in IKEv2 negotiation.

Use **undo certificate domain** to remove a PKI domain for signature authentication in IKEv2 negotiation.

Syntax

certificate domain *domain-name*[**sign** | **verify**]

undo certificate domain *domain-name*

Default

PKI domains configured in system view are used.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

domain-name: Specifies a PKI domain by its name, a case-insensitive string of 1 to 31 characters.

sign: Uses the local certificate in the PKI domain to generate a signature.

verify: Uses the CA certificate in the PKI domain to verify the remote end's certificate.

Usage guidelines

If you do not specify the **sign** or **verify** keyword, the PKI domain is used for both **sign** and **verify** purposes. You can specify a PKI domain for each purpose by executing this command multiple times. If you specify the same PKI domain for both purposes, the later configuration takes effect. For example, if you execute **certificate domain abc sign** and **certificate domain abc verify** successively, the PKI domain**abc** will be used only for verification.

If the local end uses RSA, DSA, or ECDSA signature authentication, you must specify a PKI domain for signature generation. If the remote end uses RSA, DSA, or ECDSA signature authentication, you must specify a PKI domain for verifying the remote end's certificate. If you do not specify PKI domains, the PKI domains configured in system view will be used.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Specify the PKI domain **abc** for signature. Specify the PKI domain**def** for verification.

```
[Sysname-ikev2-profile-profile1] certificate domain abc sign
[Sysname-ikev2-profile-profile1] certificate domain def verify
```

Related commands

- **authentication-method**
- **pki domain**

config-exchange

Use **config-exchange** to enable configuration exchange.

Use **undo config-exchange** to disable configuration exchange.

Syntax

config-exchange {**request** | **set** { **accept** | **send** } }

undo config-exchange {**request** | **set** { **accept** | **send** } }

Default

Configuration exchange is disabled.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

request: Enables the device to send request messages carrying the configuration request payload during theIKE_AUTH exchange.

set: Specifies the configuration set payload exchange.

accept: Enables the device to accept the configuration set payload carried in Info messages.

send: Enables the device to send Info messages carrying the configuration set payload.

Usage guidelines

The configuration exchange feature enables the local and remote ends to exchange configuration data, such as gateway address, internal IP address, and route. The exchange includes data request and response, and data push and response. The enterprise center can push IP addresses to branches. The branches can request IP addresses, but the requested IP addresses cannot be used.

You can specify both **request** and **set** for the device.

If you specify **request** for the local end, the remote end will respond if it can obtain the requested data through AAA authorization.

If you specify **set send** for the local end, you must specify **set accept** for the remote end.

The device with **set send** specified pushes an IP address after the IKEv2 SA is set up if it does not receive any configuration request from the peer.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Enable the local end to add the configuration request payload to the request message of IKE_AUTH exchange.

```
[Sysname-ikev2-profile-profile1] config-exchange request
```

Related commands

- **aaa authorization**
- **configuration policy**
- **display ikev2 profile**

display ikev2 policy

Use **display ikev2 policy** to display the IKEv2 policy configuration.

Syntax

display ikev2 policy [*policy-name* | **default**]

Views

Any view

Predefined user roles

network-admin

network-operator

mdc-admin

mdc-operator

Parameters

policy-name: Specifies an IKEv2 policy by its name, a case-insensitive string of 1 to 63 characters.

default: Specifies the default IKEv2 policy.

Usage guidelines

If you do not specify any parameters, this command displays the configuration of all IKEv2 policies.

Examples

Display the configuration of all IKEv2 policies.

```
<Sysname> display ikev2 policy
IKEv2 policy: 1
Priority: 100
  Match local address: 1.1.1.1
  Match local address ipv6: 1:1::1:1
Match VRF: vpn1
Proposal: 1
Proposal: 2
IKEv2 policy: default
  Match local address: Any
Match VRF: Any
  Proposal: default
```

Table 1 Command output

Field	Description
IKEv2 policy	Name of the IKEv2 policy.
Priority	Priority of the IKEv2 policy.
Match local address	IPv4 address to which the IKEv2 policy can be applied.
Match local address ipv6	IPv6 address to which the IKEv2 policy can be applied.

Field	Description
Match VRF	VPN instance to which the IKEv2 policy can be applied.
Proposal	IKEv2 proposal that the IKEv2 policy uses.

Related commands

ikev2policy

display ikev2 profile

Use **display ikev2 profile** to display the IKEv2 profile configuration.

Syntax

display ikev2 profile [*profile-name*]

Views

Any view

Predefined user roles

network-admin

network-operator

mdc-admin

mdc-operator

Parameters

profile-name: Specifies an IKEv2 profile by its name, a case-insensitive string of 1 to 63 characters. If you do not specify an IKEv2 profile, this command displays the configuration of all IKEv2 profiles.

Examples

Display the configuration of all IKEv2 profiles.

```
<Sysname> display ikev2 profile
IKEv2 profile: 1
Priority: 100
  Matchcriteria:
Local address 1.1.1.1
    Local address Ethernet0/1
    Local address 1:1::1:1
Remote identity address 3.3.3.3/32
VRFvrf1
Local identity:address 1.1.1.1
  Local authentication method: pre-share
  Remote authentication methods: pre-share
Keychain: Keychain1
Sign certificate domain:
     Domain1
abc
  Verify certificate domain:
     Domain2
yy
SA duration: 500 seconds
```

```
 DPD: Interval 32 secs, retry-interval 23 secs, periodic
Configexchange: request, set accept, set send
  NAT keepalive: 10 seconds
  Inside VRF: vrf1
  AAA authorization: Domain domain1, username ikev2
```

Table 2 Command output

Field	Description
IKEv2 profile	Name of the IKEv2 profile.
Priority	Priority of the IKEv2 profile.
Match criteria	Criteria for looking up the IKEv2 profile.
Local identity	ID of the local end.
Local authentication method	Method that the local end uses for authentication.
Remote authentication methods	Methods that the remote end uses for authentication.
Keychain	IKEv2 keychain that the IKEv2 profile uses.
Sign certificate domain	PKI domain used for signature generation.
Verify certificate domain	PKI domain used for verifying the remote end's certificate.
SA duration	Lifetime of the IKEv2 SA.
DPD	DPD settings: • Detection interval in seconds. • Retry interval in seconds. • Detection mode, on demand or periodically. If DPD is disabled, this field displays **Disabled**.
Config exchange	Configuration exchange settings: • **request**—The local end sends request messages carrying the configuration request payload during theIKE_AUTH exchange. • **set accept**—The local end accepts the configuration set payload carried in Info messages. • **set send**—The local end sends Info messages carrying the configuration set payload.
NAT keepalive	NAT keepalive interval in seconds.
Inside vrf	Inside VPN instance.
AAA authorization	AAA authorization settings: • ISP domain name. • Username.

Related commands

ikev2profile

display ikev2 proposal

Use **display ikev2 proposal** to display the IKEv2 proposal configuration.

Syntax

display ikev2 proposal [*name* | **default**]

Views

Any view

Predefined user roles

network-admin

network-operator

mdc-admin

mdc-operator

Parameters

name: Specifies an IKEv2 proposal by its name, a case-insensitive string of 1 to 63 characters.

default: Specifies the default IKEv2 proposal.

Usage guidelines

This command displays IKEv2 proposals in descending order of priorities. If you do not specify any parameters, this command displays the configuration of all IKEv2 proposals.

Examples

Display the configuration of all IKEv2 proposals.

```
<Sysname> display ikev2 proposal
IKEv2 proposal: 1
  Encryption: 3DES-CBC, AES-CBC-128, AES-CTR-192, CAMELLIA-CBC-128
  Integrity: MD5, SHA256, AES-XCBC
  PRF: MD5, SHA256, AES-XCBC
  DH group: MODP1024/Group 2, MODP1536/Group 5

IKEv2 proposal: default
  Encryption: AES-CBC-128, 3DES-CBC
  Integrity: SHA1, MD5
  PRF: SHA1, MD5
  DH group: MODP1536/Group 5, MODP1024/Group 2
```

Table 3 Command output

Field	Description
IKEv2 proposal	Name of the IKEv2 proposal.
Encryption	Encryption algorithms that the IKEv2 proposal uses.
Integrity	Integrity protection algorithms that the IKEv2 proposal uses.
PRF	PRF algorithms that the IKEv2 proposal uses.
DH group	DH groups that the IKEv2 proposal uses.

Related commands

ikev2proposal

display ikev2 sa

Use **display ikev2 sa** to display the IKEv2 SA information.

Syntax

display ikev2 sa [{ **local** | **remote** } { *ipv4-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*]] [**verbose**[**tunnel***tunnel-id*]]

Views

Any view

Predefined user roles

network-admin

network-operator

mdc-admin

mdc-operator

Parameters

local: Displays IKEv2 SA information for a local IP address.

remote: Displays IKEv2 SA information for a remote IP address.

ipv4-address: Specifies a local or remote IPv4 address.

ipv6*ipv6-address*: Specifies a local or remote IPv6 address.

vpn-instance*vpn-instance-name*: Displays informationabout theIKEv2 SAsin an MPLS L3VPN instance. The *vpn-instance-name*argument represents the VPN instance name, a case-sensitive string of 1 to 31 characters. If you do not specify a VPN instance, this command displays information about IKEv2 SAs for the public network.

verbose: Displays detailed information. If you do not specify this keyword, the command displays the summary information.

tunnel*tunnel-id*: Displays detailed IKEv2 SA information for an IPsec tunnel. The *tunnel-id* argument specifies an IPsec tunnel by its ID in the range of 1 to 2000000000.

Usage guidelines

If you do not specify any parameters, this command displays summary information about all IKEv2 SAs.

Examples

Display summary information about all IKEv2 SAs.

```
<Sysname> display ikev2 sa
     Tunnel ID          Local             Remote             Status
  ------------------------------------------------------------------------
     1                  1.1.1.1/500       1.1.1.2/500        EST
     2                  2.2.2.1/500       2.2.2.2/500        EST
  Status:
  IN-NEGO: Negotiating, EST: Established, DEL: Deleting
# Display summary IKEv2 SA information for the remote IP address 1.1.1.2.
<Sysname> display ikev2 sa remote 1.1.1.2
     Tunnel ID          Local             Remote             Status
  ------------------------------------------------------------------------
     1                  1.1.1.1/500       1.1.1.2/500        EST
  Status:
  IN-NEGO: Negotiating, EST: Established, DEL: Deleting
```

Table 4 Command output

Field	Description
Tunnel ID	ID of the IPsec tunnel to which the IKEv2 SA belongs.
Local	Local IP address of the IKEv2 SA.
Remote	Remote IP address of the IKEv2 SA.
Status	Status of the IKEv2 SA: • **IN-NEGO** (**Negotiating**)—The IKEv2 SA is under negotiation. • **EST** (**Established**)—The IKEv2 SA has been set up. • **DEL** (**Deleting**)—The IKEv2 SA is about to be deleted.

Display detailed information about all IKEv2 SAs.

```
<Sysname> display ikev2 sa verbose
  Tunnel ID: 1
  Local IP/Port: 1.1.1.1/500
  Remote IP/Port: 1.1.1.2/500
  Outside VRF: -
  Inside VRF: -
  Local SPI: 8f8af3dbf5023a00
  Remote SPI: 0131565b9b3155fa

  Local ID type: FQDN
  Local ID: router_a
  Remote ID type: FQDN
  Remote ID: router_b

  Auth sign method: Pre-shared key
  Auth verify method: Pre-shared key
  Integrity algorithm: HMAC_MD5
  PRF algorithm: HMAC_MD5
  Encryption algorithm: AES-CBC-192

  Life duration: 86400 secs
  Remaining key duration: 85604 secs
  Diffie-Hellman group: MODP1024/Group2
  NAT traversal: Not detected
  DPD: Interval 20 secs, retry interval 2 secs
  Transmitting entity: Initiator

  Local window: 1
  Remote window: 1
  Local request message ID: 2
  Remote request message ID:2
  Local next message ID: 0
  Remote next message ID: 0

  Pushed IP address: 192.168.1.5
```

```
  Assigned IP address: 192.168.2.24
```

Display detailed IKEv2 SA information for the remote IP address 1.1.1.2.

```
<Sysname> display ikev2 saremote 1.1.1.2 verbose
  Tunnel ID: 1
  Local IP/Port: 1.1.1.1/500
  Remote IP/Port: 1.1.1.2/500
  Outside VRF: -
  Inside VRF: -
  Local SPI: 8f8af3dbf5023a00
  Remote SPI: 0131565b9b3155fa

  Local ID type: FQDN
  Local ID: router_a
  Remote ID type: FQDN
  Remote ID: router_b

  Auth sign method: Pre-shared key
  Auth verify method: Pre-shared key
  Integrity algorithm: HMAC_MD5
  PRF algorithm: HMAC_MD5
  Encryption algorithm: AES-CBC-192

  Life duration: 86400 secs
  Remaining key duration: 85604 secs
  Diffie-Hellman group: MODP1024/Group2
  NAT traversal: Not detected
  DPD: Interval 30 secs, retry 10 secs
  Transmitting entity: Initiator

  Local window: 1
  Remote window: 1
  Local request message ID: 2
  Remote request message ID: 2
  Local next message ID: 0
  Remote next message ID: 0

  Pushed IP address: 192.168.1.5
  Assigned IP address: 192.168.2.24
```

Table 5 Command output

Field	Description
Tunnel ID	ID of the IPsec tunnel to which the IKEv2 SA belongs.
Local IP/Port	IP address and port number of the local security gateway.
Remote IP/Port	IP address and port number of the remote security gateway.

Field	Description
Outside VRF	Name of the VPN instance to which the protected outbound data flow belongs. If the protected outbound data flow belongs to the public network, this field displays a hyphen (**-**).
Inside VRF	Name of the VPN instance to which the protected inbound data flow belongs. If the protected inbound data flow belongs to the public network, this field displays a hyphen (**-**).
Local SPI	SPI that the local end uses.
Remote SPI	SPI that the remote end uses.
Local ID type	ID type of the local security gateway.
Local ID	ID of the local security gateway.
Remote ID type	ID type of the remote security gateway.
Remote ID	ID of the remote security gateway.
Auth sign method	Signature methodthat the IKEv2 proposal usesin authentication.
Auth verify method	Verification methodthat the IKEv2 proposal usesin authentication.
Integrity algorithm	Integrity protection algorithms that the IKEv2 proposal uses.
PRF algorithm	PRF algorithms that the IKEv2 proposal uses.
Encryption algorithm	Encryption algorithms that the IKEv2 proposal uses.
Life duration	Lifetime of the IKEv2 SA, in seconds.
Remaining key duration	Remaining lifetime of the IKEv2 SA, in seconds.
Diffie-Hellman group	DH groups used in IKEv2 key negotiation.
NAT traversal	Whether a NAT gateway is detected between the local and remote ends.
DPD	DPD settings: • Detection interval in seconds. • Retry interval in seconds. If DPD is disabled, this field displays **Disabled**.
Transmitting entity	Role of the local end in IKEv2 negotiation, initiator or responder.
Local window	Window size that the local end uses.
Remote window	Window size that the remote end uses.
Local request messageID	ID of the request message that the local end is about to send.
Remote request messageID	ID of the request message that the remote end is about to send.
Local next messageID	ID of the message that the local end expects to receive.
Remote next messageID	ID of the message that the remote end expects to receive.
Pushed IP address	IP address pushed to the local end by the remote end.
Assigned IP address	IP address assigned to the remote end by the local end .

dh

Use **dh** to specify DH groups to be used in IKEv2 key negotiation.

Use **undo group** to restore the default.

Syntax

In non-FIPS mode:

dh{**group1** | **group14**| **group2** | **group24**| **group5** | **group19** | **group20**} *

undo dh

In FIPS mode:

dh{**group14** | **group19** | **group20**} *

undo dh

Default

No DH group is specified for an IKEv2 proposal.

Views

IKEv2 proposal view

Predefined user roles

network-admin

mdc-admin

Parameters

group1: Uses the 768-bitDiffie-Hellman group.

group2: Uses the 1024-bitDiffie-Hellman group.

group5: Uses the 1536-bitDiffie-Hellman group.

group14: Uses the 2048-bit Diffie-Hellman group.

group24: Uses the 2048-bit Diffie-Hellman group with the 256-bit prime order subgroup.

group19: Usesthe 256-bit ECP Diffie-Hellman group.

group20: Usesthe 384-bit ECP Diffie-Hellman group.

Usage guidelines

A DH group with a higher group number provides higher security but needs more time for processing. To achieve the best trade-off between processing performance and security, choose proper DH groups for your network.

You must specify a minimum of one DH group for an IKEv2 proposal. Otherwise, the proposal is incomplete and useless.

You can specify multiple DH groups for an IKEv2 proposal. A group specified earlier has a higher priority.

Examples

Specify DH groups 1 for the IKEv2 proposal 1.

```
<Sysname> system-view
[Sysname] ikev2 proposal 1
[Sysname-ikev2-proposal-1] dh group1
```

Related commands

ikev2 proposal

dpd

Use **dpd** to configure IKEv2 DPD.

Use **undo dpd** to disable IKEv2 DPD.

Syntax

dpd interval*interval* [**retry***seconds*] { **on-demand** | **periodic** }

undo dpd interval

Default

IKEv2 DPD is disabled. The global IKEv2 DPD settings are used.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

interval*interval*: Specifies a DPD triggering interval in the range of 10 to 3600 seconds.

retry *seconds*: Specifies the DPD retry interval in the range of 2 to 60 seconds. The default is 5 seconds.

on-demand: Triggers DPD on demand. The device triggers DPD if it has IPsec traffic to send and has not received any IPsec packets from the peer for the specified interval.

periodic: Triggers DPD at regular intervals. The device triggers DPD at the specified interval.

Usage guidelines

DPD is triggered periodically or on-demand. As a best practice, use the on-demand mode when the device communicates with a large number of IKEv2 peers. For an earlier detection of dead peers, use the periodic triggering mode, which consumes more bandwidth and CPU.

The triggering interval must be longer than the retry interval, so that the device will not trigger a new round of DPD during a DPD retry.

Examples

Configure on-demand IKEv2 DPD. Set the DPD triggering interval to 10 seconds and the retry interval to 5 seconds.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
[Sysname-ikev2-profile-profile1] dpd interval 10 retry 5 on-demand
```

Related commands

ikev2 dpd

encryption

Use **encryption** to specify encryption algorithms for an IKEv2 proposal.

Use **undo encryption** to restore the default.

Syntax

In non-FIPS mode:

encryption { **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256** | **camellia-cbc-128** | **camellia-cbc-192** | **camellia-cbc-256** | **des-cbc** } *

undo encryption

In FIPS mode:

encryption{**aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** | **aes-ctr-128** | **aes-ctr-192** | **aes-ctr-256**} *

undo encryption

Default

No encryption algorithm is specified for an IKEv2 proposal.

Views

IKEv2 proposal view

Predefined user roles

network-admin

mdc-admin

Parameters

3des-cbc: Specifies the 3DESalgorithm in CBC mode, which uses a168-bit key.

aes-cbc-128: Specifies the AESalgorithm in CBC mode, which uses a 128-bit key.

aes-cbc-192: Specifies the AESalgorithm in CBC mode, which uses a 192-bit key.

aes-cbc-256: Specifies the AESalgorithm in CBC mode, which uses a 256-bit key.

aes-ctr-128: Specifies the AESalgorithm in CTR mode, which uses a 128-bit key.

aes-ctr-192: Specifies the AESalgorithm in CTR mode, which uses a 192-bit key.

aes-ctr-256: Specifies the AESalgorithm in CTR mode, which uses a 256-bit key.

camellia-cbc-128: Specifies theCamellia algorithm in CBC mode, which uses a 128-bit key.

camellia-cbc-192: Specifies the Camellia algorithm in CBC mode, which uses a 192-bit key.

camellia-cbc-256: Specifies the Camellia algorithm in CBC mode, which uses a 256-bit key.

des-cbc: Specifies the DESalgorithm in CBC mode, which uses a56-bit key.

Usage guidelines

You must specify a minimum of one encryption algorithm for an IKEv2 proposal. Otherwise, the proposal is incomplete and useless. You can specify multiple encryption algorithms for an IKEv2 proposal. An algorithm specified earlier has a higher priority.

Examples

Specifythe 168-bit 3DES algorithm in CBC mode as the encryption algorithm for the IKE proposal **prop1**.

```
<Sysname> system-view
[Sysname] ikev2 proposal prop1
[Sysname-ikev2-proposal-prop1] encryption-algorithm 3des-cbc
```

Related commands

ikev2 proposal

hostname

Use **hostname** to specify the host name of an IKEv2 peer.

Use **undohostname** to restore the default.

Syntax

hostname*name*

undohostname

Default

The IKEv2 peer's host name is not specified.

Views

IKEv2 peer view

Predefined user roles

network-admin

mdc-admin

Parameters

name: Specifies the host name of the IKEv2 peer, a case-insensitive string of 1 to 253 characters.

Usage guidelines

Only the initiator can look up an IKEv2 peer by host name in IKEv2 negotiation, and the initiator must use an IPsec policy rather than an IPsec profile.

Examples

Create an IKEv2 keychain named **key1**.

```
<Sysname> system-view
[Sysname] ikev2keychainkey1
```

Create an IKEv2 peer named **peer1**.

```
[Sysname-ikev2-keychain-key1] peer peer1
```

Specify the host name **test** of the IKEv2 peer.

```
[Sysname-ikev2-keychain-key1-peer-peer1] hostname test
```

Related commands

- **ikev2 keychain**
- **peer**

identity

Use **identity** to specify the ID of an IKEv2 peer.

Use **undo identity** to restore the default.

Syntax

identity{ **address** { *ipv4-address* | **ipv6** { *ipv6-address* } } | **fqdn***fqdn-name* | **email***email-string* | **key-id***key-id-string*}

undoidentity

Default

The IKEv2 peer's ID is not specified.

Views

IKEv2 peer view

Predefined user roles

network-admin

mdc-admin

Parameters

ipv4-address: Specifies the IPv4 address of the peer.

ipv6 *ipv6-address*: Specifies the IPv6 address of the peer.

fqdn*fqdn-name*: Specifies the FQDN of the peer. The *fqdn-name* argument is a case-sensitive string of 1 to 255 characters, such as www.test.com.

email*email-string*: Specifies the email address of the peer. The *email-string* argument is a case-sensitive string of 1 to 255 characters in the format defined by RFC 822, such as esec@test.com.

key-id*key-id-string*: Specifies the remote gateway's key ID. The *key-id-string* argument is a case-sensitive string of 1 to 255 characters, and is usually a vendor-specific string for doing proprietarytypes of identification.

Usage guidelines

Only the responder can look up an IKEv2 peer by ID in IKEv2 negotiation. The initiator does not know the peer ID when initiating the IKEv2 negotiation, so it cannot use an ID for IKEv2 peer lookup.

Examples

Create an IKEv2 keychain named **key1**.

```
<Sysname> system-view
[Sysname] ikev2 keychain key1
```

Create an IKEv2 peer named **peer1**.

```
[Sysname-ikev2-keychain-key1] peer peer1
```

Specify the peer IPv4 address 1.1.1.2 as the ID of the IKEv2 peer.

```
[Sysname-ikev2-keychain-key1-peer-peer1] identity address 1.1.1.2
```

Related commands

- **ikev2 keychain**
- **peer**

identity local

Use **identity local** to configure the local ID, the ID that the device uses to identify itself to the peer during IKEv2 negotiation..

Use **undoidentity local** to restore the default.

Syntax

identity local { **address** { *ipv4-address* | i**pv6***ipv6-address* } | **dn** | **email***email-string* | **fqdn***fqdn-name* | **key-id***key-id-string* }

undo identity local

Default

No local ID is configured. The IP address of the interface to which the IPsec policy is applied is used as the local ID.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

address { *ipv4-address* | **ipv6***ipv6-address* }: Uses an IPv4 or IPv6 address as the local ID.

dn: Uses the DN in the local certificate as the local ID.

email*email-string*: Uses an email address as the local ID. The *email-string* argument is a case-sensitive string of 1 to 255 characters in the format defined by RFC 822, such as sec@abc.com.

fqdn*fqdn-name*: Uses an FQDN as the local ID. The *fqdn-name* argument is a case-sensitive string of 1 to 255 characters, such as www.test.com.

key-id*key-id-string*: Uses the device's key ID as the local ID. The *key-id-string* argument is a case-sensitive string of 1 to 255 characters, and is usually a vendor-specific string for doing proprietarytypes of identification.

Usage guidelines

Peers exchange local IDs for identifying each other in negotiation.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Use the IP address 2.2.2.2 as the local ID.

```
[Sysname-ikev2-profile-profile1] identity local address 2.2.2.2
```

Related commands

peer

ikev2 address-group

Use **ikev2 address-group** to configure an IKEv2 IPv4 address pool for assigning IPv4 addresses to remote peers.

Use **undo ikev2 address-group** to delete an IKEv2 IPv4 address pool.

Syntax

ikev2 address-group*group-namestart-ipv4-addressend-ipv4-address* [*mask* | *mask-length*]

undoikev2 address-group*group-name*

Default

No IKEv2 IPv4 address pools exist.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

group-name: Specifies an name for the IKEv2 IPv4 address pool. The *group-name* argument is a case-insensitive string of 1 to 63 characters.

start-ipv4-addressend-ipv4-address: Specifies an IPv4 address range. The *start-ipv4-address* argument specifies the start IPv4 address. The *end-ipv4-address* argument specifies the end IPv4 address.

mask: Specifies the IPv4 address mask.

mask-length: Specifies the length of the IPv4 address mask.

Usage guidelines

An IKE IPv4 address pool can contain a maximum of 8192 IPv4 addresses.

Examples

Configure an IKEv2 IPv4 address pool with the name **ipv4group**, address range 1.1.1.1 to 1.1.1.2, and the mask 255.255.255.0.

```
<Sysname> system-view
[Sysname] ikev2 address-group ipv4group 1.1.1.1 1.1.1.2 255.255.255.0
```

Configure an IKEv2 IPv4 address pool with the name **ipv4group**, address range 1.1.1.1 to 1.1.1.2, and the mask length 32.

```
<Sysname> system-view
[Sysname] ikev2 address-group ipv4group 1.1.1.1 1.1.1.2 32
```

Related commands

address-group

ikev2 cookie-challenge

Use **ikev2 cookie-challenge** to enable the cookie challenging feature.

Use **undoikev2 cookie-challenge** to disable the cookie challenging feature.

Syntax

ikev2 cookie-challenge*number*

undoikev2 cookie-challenge

Default

The cookiechallenging feature is disabled.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

number: Specifies the threshold for triggering the cookie challenging feature. The value range for this argument is 0 to 1000 half-open IKE SAs.

Usage guidelines

When an IKEv2 responder maintains a threshold number of half-open IKE SAs, it starts the cookie challenging mechanism.The responder generates a cookie and includes it in the response sent to the initiator. If the initiator initiates a new IKE_SA_INIT request that carries the correct cookie, the responder considers the initiator valid and proceeds with the negotiation. If the carried cookie is incorrect, the responder terminates the negotiation.

This feature can protectthe responder against DoS attacks which aim to exhaust the responder's system resources by using a large number of IKE_SA_INIT requests with forged source IP addresses.

Examples

Enable the cookiechallenging feature and set the threshold to 450.

```
<Sysname> system-view
[Sysname] ikev2 cookie-challenge 450
```

ikev2 dpd

Use **ikev2 dpd** to configure global IKEv2 DPD.

Use **undoikev2 dpd** to disable global IKEv2 DPD.

Syntax

ikev2 dpd interval*interval*[**retry***seconds*] { **on-demand** | **periodic** }

undo ikev2 dpd interval

Default

The global IKEv2 DPD feature is disabled.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

interval*interval*: Specifies a DPD triggering interval in the range of 10 to 3600 seconds.

retry *seconds*: Specifies the DPD retry interval in the range of 2 to 60 seconds. The default is 5 seconds.

on-demand: Triggers DPD on demand. The device triggers DPD if it has IPsec traffic to send and has not received any IPsec packets from the peer for the specified interval.

periodic: Triggers DPD at regular intervals. The device triggers DPD at the specified interval.

Usage guidelines

DPD is triggered periodically or on-demand. As a best practice, use the on-demand mode when the device communicates with a large number of IKEv2 peers. For an earlier detection of dead peers, use the periodic triggering mode, which consumes more bandwidth and CPU.

The triggering interval must be longer than the retry interval, so that the device will not trigger a new round of DPD during a DPD retry.

You can configure IKEv2 DPD in both IKEv2 profile view and system view. The IKEv2 DPD settings in IKEv2 profile view apply. If you do not configure IKEv2 DPD in IKEv2 profile view, the IKEv2 DPD settings in system view apply.

Examples

Configure the device to trigger IKEv2 DPD if it has IPsec traffic to send and has not received any IPsec packets from the peer for 15 seconds.

```
<Sysname> system-view
[Sysname] ikev2 dpd interval 15 on-demand
```

Configure the device to trigger IKEv2 DPD every 15 seconds.

```
<Sysname> system-view
[Sysname] ikev2 dpd interval 15 periodic
```

Related commands

dpd (IKEv2 profile view)

ikev2 ipv6-address-group

Use **ikev2 ipv6-address-group** to configure an IKEv2 IPv6 address pool for assigning IPv6 addresses to remote peers.

Use **undo ikev2 ipv6-address-group** to delete an IKEv2 IPv6 address pool.

Syntax

ikev2 ipv6-address-group*group-name***prefix***prefix/prefix-len* **assign-len***assign-len*

undoikev2 ipv6-address-group*group-name*

Default

No IKEv2 IPv6 address pools exist.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

group-name: Specifies a name for the IKEv2 IPv6 address pool. The *group-name* argument is a case-insensitive string of 1 to 63 characters.

prefix*prefix/prefix-len*: Specifies an IPv6 prefix in the format of *prefix*/*prefix length*. The value range for the *prefix-len* argument is 1 to 128.

assign-len *assign-len*: Specifies the assigned prefix length. The valuerange for the *assign-len* argument is 1 to 128, and the value must be greater than or equal to *prefix-len.* The differencebetween *assign-len* and *prefix-len* must be no more than 16.

Usage guidelines

Different from the IKEv2 IPv4 address pool, the device assigns an IPv6 subnet to a peer from the IKEv2 IPv6 address pool. The peer can use the assigned IPv6 subnet to assign IPv6 addresses to other devices.

IKEv2 IPv6 address pools cannot overlap with each other.

Examples

Configure an IKEv2 IPv6 address pool with the name **ipv6group**, prefix 1:1::/64, and the assigned prefix length 80.

```
<Sysname> system-view
[Sysname] ikev2 ipv6-address-group ipv6groupprefix:1:1::/64assign-len 80
```

Related commands

ipv6-address-group

ikev2 keychain

Use **ikev2 keychain** to create an IKEv2 keychain and enter its view, or enter the view of an existing IKEv2 keychain.

Use **undo ikev2 keychain** to delete an IKEv2 keychain.

Syntax

ikev2 keychain*keychain-name*

undoikev2 keychain*keychain-name*

Default

No IKEv2 keychains exist.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

keychain-name: Specifies a name for the IKEv2 keychain. The keychain name is a case-insensitive string of 1 to 63 characters and cannot contain a hyphen (-).

Usage guidelines

An IKEv2 keychain is required on both ends if either end uses pre-shared key authentication. The pre-shared key configured on both ends must be the same.

You can configure multiple IKEv2 peers in an IKEv2 keychain.

Examples

Create an IKEv2 keychain named **key1** and enter IKEv2 keychain view.

```
<Sysname> system-view
[Sysname] ikev2 keychain key1
[Sysname-ikev2-keychain-key1]
```

ikev2 nat-keepalive

Use **ikev2 nat-keepalive** to set the NAT keepalive interval.

Use **undoikev2 nat-keepalive** to restore the default.

Syntax

ikev2 nat-keepalive*seconds*

undoikev2 nat-keepalive

Default

The NAT keepalive interval is 10 seconds.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

seconds: Specifies the NAT keepalive interval in seconds, in the range of 5 to 3600.

Usage guidelines

This command takes effect when the device resides in the private network behind a NAT device. The device must send NAT keepalive packets regularly to its peer to keep the NAT session alive, so that the peer can access the device.

The NAT keepalive interval must be shorter than the NAT session lifetime.

Examples

#Setthe NAT keepalive interval to 5 seconds.

```
<Sysname> system-view
[Sysname] ikev2nat-keepalive 5
```

ikev2 policy

Use **ikev2 policy** to create an IKEv2 policy and enter its view, or enter the view of an existing IKEv2 policy.

Use **undo ikev2 policy** to delete an IKEv2 policy.

Syntax

ikev2 policy*policy-name*

undoikev2 policy*policy-name*

Default

An IKEv2policy named **default** exists, which uses the default IKEv2 proposal and matches any local addresses.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

policy-name: Specifies a name for the IKEv2 policy. The policy name is a case-insensitive string of 1 to 63 characters.

Usage guidelines

Each end must have an IKEv2 policy for the IKE_SA_INIT exchange. The initiator looks up an IKEv2 policy by the IP address of the interface to which the IPsec policy is applied and the VPN instance to which the interface belongs. The responder looks up an IKEv2 policy by the IP address of the interface that receives the IKEv2 packet and the VPN instance to which the interface belongs. An IKEv2 policy uses IKEv2 proposals to define the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups to be used for negotiation.

You can configure multiple IKEv2 policies. An IKEv2 policy must have a minimum of one IKEv2 proposal. Otherwise, the policy is incomplete.

If the initiator uses an IPsec policy that is bound to a source interface, the initiator looks up an IKEv2 policy by the IP address of the source interface.

You can set priorities to adjust the match order of IKEv2 policies that have the same match criteria.

If no IKEv2 policy is configured, the default IKEv2 policy is used. You cannot enter the view of the default IKEv2 policy, nor modify it.

Examples

Create an IKEv2 policy named **policy1** and enter IKEv2 policy view.

```
<Sysname> system-view
[Sysname] ikev2 policy policy1
[Sysname-ikev2-policy-policy1]
```

Related commands

display ikev2 policy

ikev2 profile

Use **ikev2 profile** to create an IKEv2 profile and enter its view, or enter the view of an existing IKEv2 profile.

Use **undo ikev2 profile** to delete an IKEv2 profile.

Syntax

ikev2 profile*profile-name*

undo ikev2 profile*profile-name*

Default

No IKEv2 profiles exist.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

profile-name: Specifies a name for the IKEv2 profile. The profile name is a case-insensitive string of 1 to 63 characters.

Usage guidelines

An IKEv2 profile contains the IKEv2 SA parameters that are not negotiated, such as the identity information and authentication methods of the peers, and the matching criteria for profile lookup.

Examples

Create an IKEv2 profile named **profile1** and enter IKEv2 profile view.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
[Sysname-ikev2-profile-profile1]
```

Related commands

display ikev2 profile

ikev2 proposal

Use **ikev2 proposal** to create an IKEv2 proposal and enter its view, or enter the view of an existing IKEv2 proposal.

Use **undo ikev2 proposal** to delete an IKEv2 proposal.

Syntax

ikev2 proposal*proposal-name*

undoikev2 proposal*proposal-name*

Default

An IKEv2 proposal named **default** exists, which has the lowest priority and uses the following settings:

- In non-FIPS mode:
  - **Encryption algorithm**—AES-CBC-128 and 3DES.
  - **Integrity protection algorithm**—HMAC-SHA1 and HMAC-MD5.
  - **PRF algorithm**—HMAC-SHA1 and HMAC-MD5.
  - **DH group**—Group 5 and group 2.
- In FIPS mode:
  - **Encryption algorithm**—AES-CBC-128 and AES-CTR-128.
  - **Integrity protection algorithm**—HMAC-SHA1 and HMAC-SHA256.
  - **PRF algorithm**—HMAC-SHA1 and HMAC-SHA256.
  - **DH group**—Group 14 and group 19.

Views

System view

Predefined user roles

network-admin

mdc-admin

Parameters

proposal-name: Specifies a name for the IKEv2 proposal. The proposal name is a case-insensitive string of 1 to 63 characters and cannot be **default**.

Usage guidelines

An IKEv2 proposal contains security parameters used in IKE_SA_INIT exchanges, including the encryption algorithms, integrity protection algorithms, PRF algorithms, and DH groups.

An IKEv2 proposal must have a minimum of one set of security parameters, including one encryption algorithm, one integrity protection algorithm, one PRF algorithm, and one DH group.

In an IKEv2 proposal, you can specify multiple parameters of the same type. The parameters of different types combine and form multiple sets of security parameters. If you want to use only one set of security parameters, configure only one set of security parameters for the IKEv2 proposal.

Examples

Create an IKEv2 proposal named **prop1**. Specify the encryption algorithm AES-CBC-128, integrity protection algorithm SHA1, PRF algorithm SHA1, and DH group 2.

```
<Sysname> system-view
[Sysname] ikev2 proposal prop1
[Sysname-ikev2-proposal-prop1] encryption-algorithm aes-cbc-128
[Sysname-ikev2-proposal-prop1] authentication-algorithm sha1
[Sysname-ikev2-proposal-prop1] prf sha1
[Sysname-ikev2-proposal-prop1] dh group2
```

Related commands

- **encryption-algorithm**
- **integrity**
- **prf**
- **dh**

inside-vrf

Use **inside-vrf**to specify an inside VPN instance.

Use **undo inside-vrf**to restore the default.

Syntax

inside-vrf *vrf-name*

undoinside-vrf

Default

No inside VPN instance is specified. The internal and external networks are in the same VPN instance. The device forwards protected data to this VPN instance.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

vrf-name: Specifies the VPN instance to which the protected data belongs. The *vrf-name*argument represents the VPN instance name, a case-sensitive string of 1 to 31 characters.

Usage guidelines

This command determines where the device should forward received IPsec packets after it de-encapsulates them. If you configure this command, the device looks for a route in the specified VPN to forward the packets. If you do not configure this command, the internal and external networks are in the same VPN instance. The device looks for a route in this VPN instance to forward the packets.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Specify the inside VPN instance **vpn1**.

```
[Sysname-ikev2-profile-profile1] inside-vrf vpn1
```

integrity

Use **integrity**to specify integrity protection algorithms for an IKEv2 proposal.

Use **undo integrity**to restore the default.

Syntax

In non-FIPS mode:

integrity{ **aes-xcbc-mac** | **md5** | **sha1** | **sha256** | **sha384** | **sha512**} *

undo integrity

In FIPS mode:

integrity{ **sha1** | **sha256** | **sha384** | **sha512**} *

undo integrity

Default

No integrity protection algorithm is specified for an IKEv2 proposal.

Views

IKEv2 proposal view

Predefined user roles

network-admin

mdc-admin

Parameters

aes-xcbc-mac: Uses the HMAC-AES-XCBC-MAC algorithm.

md5: Uses the HMAC-MD5 algorithm.

sha1: Uses the HMAC-SHA1 algorithm.

sha256: Uses the HMAC-SHA256 algorithm.

sha384: Uses the HMAC-SHA384 algorithm.

sha512: Uses the HMAC-SHA512 algorithm.

Usage guidelines

You must specify a minimum of one integrity protection algorithm for an IKEv2 proposal. Otherwise, the proposal is incomplete and useless. You can specify multiple integrity protection algorithms for an IKEv2 proposal. An algorithm specified earlier has a higher priority.

Examples

Create an IKEv2 proposal named **prop1**.

```
<Sysname> system-view
[Sysname] ikev2 proposal prop1
```

Specify HMAC-SHA1 and HMAC-MD5 as theintegrity protection algorithms, with HMAC-SHA1 preferred.

```
[Sysname-ikev2-proposal-prop1] integrity sha1 md5
```

Related commands

ikev2 proposal

keychain

Use **keychain** to specify an IKEv2 keychain for pre-shared key authentication.

Use **undo keychain** to restore the default.

Syntax

keychain*keychain-name*

undo keychain

Default

No IKEv2 keychain is specified for an IKEv2 profile.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

keychain-name: Specifies an IKEv2 keychain by its name. The keychain name is a case-insensitive string of 1 to 63 characters and cannot contain a hyphen (-).

Usage guidelines

An IKEv2 keychain is required on both ends if either end uses pre-shared key authentication. You can specify only one IKEv2 keychain for an IKEv2 profile.

You can specify the same IKEv2 keychain for different IKEv2 profiles.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Specify the IKEv2 keychain **keychain1**.

```
[Sysname-ikev2-profile-profile1] keychain keychain1
```

Related commands

- **display ikev2 profile**
- **ikev2 keychain**

match local (IKEv2 profile view)

Use **match local**to specify a local interface or a local IP address to which an IKEv2 profile can be applied.

Use **undo match local**to remove a local interface or a local IP address to which an IKEv2 profile can be applied.

Syntax

match local address { *interface-type interface-number* | { *ipv4-address* | **ipv6***ipv6-address* } }

undo match local address { *interface-type interface-number* | { *ipv4-address* | **ipv6***ipv6-address* } }

Default

An IKEv2 profile can be applied to any local interface or IP address.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

address: Specifies a local interface or IP address to which an IKEv2 profile can be applied.

interface-type interface-number: Specifies a local interface by its type and number. It can be any Layer 3 interface.

ipv4-address: Specifies the IPv4 address of a local interface.

ipv6*ipv6-address*: Specifies the IPv6 address of a local interface.

Usage guidelines

Use this command to specify which address or interface can use the IKEv2 profile for IKEv2 negotiation. The interface is the interface that receives IKEv2 packets. The IP address is the IP address of the interface that receives IKEv2 packets.

An IKEv2 profile configured earlier has a higher priority. To give an IKEv2 profile that is configured later a higher priority, you can configure the **priority**command or this command for the profile. For example, suppose you configured IKEv2 profile A before configuring IKEv2 profile B, and you configured the **match remote identity address range 2.2.2.1 2.2.2.100** command for IKEv2 profile A and the**match remote identity address range 2.2.2.1 2.2.2.10** command for IKEv2 profile B. For the local interface with the IP address 3.3.3.3 to negotiate with the peer 2.2.2.6, IKEv2 profile A is preferred because IKEv2 profile A was configured earlier. To use IKEv2 profile B, you can use this command to restrict the application scope of IKEv2 profile B to IPv4 address 3.3.3.3.

You can specify multiple applicable local interfaces or IP addresses for an IKEv2 profile.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Apply the IKEv2 profile **profile1** to the interface whose IP address is 2.2.2.2.

```
[Sysname-ikev2-profile-profile1] match local address 2.2.2.2
```

Related commands

match remote

matchlocaladdress (IKEv2 policy view)

Use **match local address**to specify a local interface or a local address that an IKEv2 policy matches.

Use **undo match local address**to remove a local interface or a local address that an IKEv2 policy matches.

Syntax

match local address { *interface-typeinterface-number* | { *ipv4-address* | **ipv6***ipv6-address* } }

undomatch local address { *interface-typeinterface-number* | { *ipv4-address* | **ipv6***ipv6-address* } }

Default

No local interface or address is specified, and the IKEv2 policy matches any local interface or local address.

Views

IKEv2 policy view

Predefined user roles

network-admin

mdc-admin

Parameters

interface-type interface-number: Specifies a local interface by its type and number. It can be any Layer 3 interface.

ipv4-address: Specifies the IPv4 address of a local interface.

ipv6*ipv6-address*: Specifies the IPv6 address of a local interface.

Usage guidelines

IKEv2 policies with this command configured are looked up before those that do not have this command configured.

Examples

Configure the IKEv2 policy **policy1** to match the local address 3.3.3.3.

```
<Sysname> system-view
[Sysname] ikev2 policy policy1
[Sysname-ikev2-policy-policy1] match localaddress 3.3.3.3
```

Related commands

- **display ikev2 policy**
- **match vrf**

match remote

Use **match remote** to configure a peer ID that an IKEv2 profile matches.

Use **undo match remote** to delete a peer ID that an IKEv2 profile matches.

Syntax

matchremote { **certificate***policy-name* | **identity** { **address** { { *ipv4-address* [*mask* | *mask-length*] | **range***low-ipv4-address high-ipv4-address* } | **ipv6** { *ipv6-address* [*prefix-length*] | **range***low-ipv6-address high-ipv6-address* } } | **fqdn***fqdn-name* | **email***email-string* | **key-id***key-id-string* } }

undo matchremote { **certificate***policy-name* | **identity** { **address** { { *ipv4-address* [*mask* |*mask-length*] | **range***low-ipv4-address high-ipv4-address* } | **ipv6** { *ipv6-address* [*prefix-length*] | **range***low-ipv6-address high-ipv6-address* } } | **fqdn***fqdn-name* | **email***email-string* | **key-id***key-id-string* } }

Default

No matching peer ID is configured for the IKEv2 profile.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

certificate*policy-name*: Uses the information in the peer's digital certificate as the peer ID for IKEv2 profile matching. The *policy-name* argument specifies a certificate-based access control policy by its name, a case-insensitive string of 1 to 31 characters.

identity: Uses the specified information as the peer ID for IKEv2 profile matching. The specified information is configured on the peer by using the **local-identity** command.

- **address***ipv4-address* [*mask* | *mask-length*]: Uses an IPv4 host address or an IPv4 subnet address as the peer ID for IKEv2 profile matching. The value range for the *mask-length* argument is 0 to 32.
- **address range***low-ipv4-address high-ipv4-address*: Uses a range of IPv4 addresses as the peer ID for IKEv2 profile matching. The end address must be higher than the start address.
- **addressipv6***ipv6-address* [*prefix-length*]: Uses an IPv6 host address or an IPv6 subnet address as the peer ID for IKEv2 profile matching. The value range for the*prefix-length* argument is 0 to 128.
- **addressipv6 range***low-ipv6-address high-ipv6-address*: Uses a range of IPv6 addresses as the peer ID for IKEv2 profile matching. The end address must be higher than the start address.

- **fqdn***fqdn-name*: Uses the peer's FQDN as the peer ID for IKEv2 profile matching. The *fqdn-name* argument is a case-sensitive string of 1 to 255 characters, such as www.test.com.
- **email***email-string*: Uses peer's email address as the peer ID for IKEv2 profile matching. The *email-string* argument is a case-sensitive string of 1 to 255 characters in the format defined by RFC 822, such as sec@abc.com.
- **key-id***key-id-string*: Uses the peer's key ID as the peer ID for IKEv2 profile matching. The *key-id-string* argument is a case-sensitive string of 1 to 255 characters, and is usually a vendor-specific string for doing proprietarytypes of identification.

Usage guidelines

The device compares the received peer ID with the peer IDs configured in local IKEv2 profiles. If a match is found, it uses the IKEv2 profile with the matching peer ID for IKEv2 negotiation. If you have configured the **match local address** and **match vrf** commands, the IKEv2 profile must also match the specified local interface or address and the specified VPN instance.

To make sure only one IKEv2 profile is matched for a peer, do not configure the same peer ID for two or more IKEv2 profiles. If you configure the same peer ID for two or more IKEv2 profiles, which IKEv2 profile is selected for IKEv2 negotiation is unpredictable.

You can configure an IKEv2 profile to match multiple peer IDs. A peer ID configured earlier has a higher priority.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Configure the IKEv2 profile to match the peer ID that is the FQDN name **www.test.com**.

```
[Sysname-ikev2-profile-profile1] match remote identity fqdn www.test.com
```

Configure the IKEv2 profile to match the peer ID that is the IP address 10.1.1.1.

```
[Sysname-ikev2-profile-profile1]match remote identity address 10.1.1.1
```

Related commands

- **identity local**
- **match local address**
- **match vrf**

match vrf (IKEv2 policy view)

Use **match vrf** to specify a VPN instance that an IKEv2 policy matches.

Use **undo match vrf** to restore the default.

Syntax

match vrf { **name** *vrf-name* |**any** }

undo matchvrf

Default

No VPN instance is specified, and the IKEv2 policy matches all local IP addresses in the public network.

Views

IKEv2 policy view

Predefined user roles

network-admin

mdc-admin

Parameters

name *vrf-name*: Specifies a VPN instance by its name, a case-sensitive string of 1 to 31 characters.

any: Specifies the public network and all VPN instances.

Usage guidelines

Each end must have an IKEv2 policy for the IKE_SA_INIT exchange. The initiator looks up an IKEv2 policy by the IP address of the interface to which the IPsec policy is applied and the VPN instance to which the interface belongs. The responder looks up an IKEv2 policy by the IP address of the interface that receives the IKEv2 packet and the VPN instance to which the interface belongs.

IKEv2 policies with this command configured are looked up before those that do not have this command configured.

Examples

Create an IKEv2 policy named **policy1**.

```
<Sysname> system-view
[Sysname] ikev2 policy policy1
```

Configure the IKEv2 policy to match the VPN instance **vpn1**.

```
[Sysname-ikev2-policy-policy1] match vrf name vpn1
```

Related commands

- **display ikev2 policy**
- **match local address**

match vrf (IKEv2 profile view)

Use **match vrf** to specify a VPN instance for an IKEv2 profile.

Use **undo match vrf** to restore the default.

Syntax

match vrf { **name***vrf-name* |**any** }

undo match vrf

Default

The IKEv2 profile belongs to the public network.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

name*vrf-name*: Specifies a VPN instance by its name, a case-sensitive string of 1 to 31 characters.

any: Specifies the public network and all VPN instances.

Usage guidelines

If an IKEv2 profile belongs to a VPN instance, only interfaces in the VPN instance can use the IKEv2 profile for IKEv2 negotiation. The VPN instance is the VPN instance to which the interface that receives IKEv2 packets belongs. If you specify the **any** keyword, interfaces in any VPN instance can use the IKEv2 profile for IKEv2 negotiation.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Specify **vrf1** as the VPN instance that theIKEv2 profile belongs to.

```
[Sysname-ikev2-profile-profile1] match vrf name vrf1
```

Related commands

match remote

nat-keepalive

Use **nat-keepalive** to set the NAT keepalive interval.

Use **ikev2 nat-keepalive** to restore the default.

Syntax

nat-keepalive*seconds*

undo nat-keepalive

Default

The NAT keepalive interval set in system view is used.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

seconds: Specifies the NAT keepalive interval in seconds, in the range of 5 to 3600.

Usage guidelines

This command takes effect when the device resides in the private network behind a NAT device. The device must send NAT keepalive packets regularly to its peer to keep the NAT session alive, so that the peer can access the device.

The NAT keepalive interval must be shorter than the NAT session lifetime.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Set the NAT keepalive interval to 1200 seconds.

```
[Sysname-ikev2-profile-profile1]nat-keepalive 1200
```

Related commands

- **display ikev2 profile**
- **ikev2 nat-keepalive**

peer

Use **peer** to create an IKEv2 peer and enter its view, or enter the view of an existing IKEv2 peer.

Use **undo peer** to delete an IKEv2 peer.

Syntax

peer*name*

undo peer*name*

Default

No IKEv2 peers exist.

Views

IKEv2 keychain view

Predefined user roles

network-admin

mdc-admin

Parameters

name: Specifies a name for the IKEv2 peer. The peer name is a case-insensitive string of 1 to 63 characters.

Usage guidelines

An IKEv2 peer contains a pre-shared key and the criteria for looking up the peer. The criteria for peer lookup includes the peer's host name, IP address, IP address range, and ID. The IKEv2 negotiation initiator uses the peer's host name, IP address, or IP address range to look up its peer. The responder uses the peer's IP address, IP address range, or ID to look up its peer.

Examples

Create an IKEv2 keychain named **key1** and enter IKEv2 keychain view.

```
<Sysname> system-view
[Sysname] ikev2 keychain key1
```

Create an IKEv2 peer named **peer1**.

```
[Sysname-ikev2-keychain-key1] peer peer1
```

Related commands

- **address**
- **hostname**
- **identity**
- **ikev2 keychain**

pre-shared-key

Use **pre-shared-key** to configure a pre-shared key.

Use **undopre-shared-key** to delete a pre-shared key.

Syntax

pre-shared-key [**local** | **remote**] {**ciphertext**| **plaintext**} *string*

undo pre-shared-key [**local** | **remote**]

Default

No pre-shared key exists.

Views

IKEv2 peer view

Predefined user roles

network-admin

mdc-admin

Parameters

local: Specifies a pre-shared key for certificate signing.

remote: Specifies a pre-shared key for certificate authentication.

ciphertext: Specifies apre-shared key in encrypted form.

plaintext: Specifies apre-shared key in plaintext form.For security purposes, the key specified in plaintext form will be stored in encrypted form.

string: Specifies the pre-shared key.The key is case sensitive. In non-FIPS mode, its plaintext form is a string of 1 to 128 characters and its encrypted form is a string of 1 to 201 characters. In FIPS mode, its plaintext form is a string of 15 to 128 characters and its encrypted form is a string of 15 to 201 characters.

Usage guidelines

If you specify the **local** or **remote** keyword, you configure an asymmetric key. If you specify neither the **local** nor the **remote** keyword, you configure a symmetric key.

To delete a key by using the **undo** command, you must specify the correct key type. For example, if you configure a key by using the **pre-shared-key local** command, you cannot delete the key by using the **undo pre-shared-key** or **undo pre-shared-key remote** command.

If you execute this command multiple times, the most recent configuration takes effect.

Examples

- On the initiator:

  # Create an IKEv2 keychain named **key1**.

  ```
  <Sysname> system-view
  [Sysname] ikev2 keychain key1
  ```

  # Create an IKEv2 peer named **peer1**.

  ```
  [Sysname-ikev2-keychain-key1] peer peer1
  ```

  # Configure the symmetric plaintext pre-shared key **111-key**.

  ```
  [Sysname-ikev2-keychain-key1-peer-peer1] pre-shared-key plaintext 111-key
  [Sysname-ikev2-keychain-key1-peer-peer1] quit
  ```

  # Create an IKEv2 peer named **peer2**.

  ```
  [Sysname-ikev2-keychain-key1] peer peer2
  ```

  # Configure asymmetric plaintext pre-shared keys. The key for certificate signing is **111-key-a** and the key for certificate authentication is **111-key-b**.

  ```
  [Sysname-ikev2-keychain-key1-peer-peer2]pre-shared-key local plaintext 111-key-a
  [Sysname-ikev2-keychain-key1-peer-peer2]pre-shared-keyremote plaintext 111-key-b
  ```

- On the responder:

  # Create an IKEv2 keychain named **telecom**.

  ```
  <Sysname> system-view
  [Sysname] ikev2 keychain telecom
  ```

  # Create an IKEv2 peer named **peer1**.

  ```
  [Sysname-ikev2-keychain-telecom] peer peer1
  ```

  # Configure the symmetric plaintext pre-shared key **111-key**.

  ```
  [Sysname-ikev2-keychain-telecom-peer-peer1] pre-shared-key plaintext 111-key
  [Sysname-ikev2-keychain-telecom-peer-peer1] quit
  ```

Create an IKEv2 peer named **peer2**.

```
[Sysname-ikev2-keychain-telecom] peer peer2
```

Configure asymmetric plaintext pre-shared keys. The key for certificate signing is **111-key-b** and the key for certificate authentication is **111-key-a**.

```
[Sysname-ikev2-keychain-telecom-peer-peer2] pre-shared-key local plaintext
111-key-b
[Sysname-ikev2-keychain-telecom-peer-peer2] pre-shared-key remote plaintext
111-key-a
```

Related commands

- **ikev2 keychain**
- **peer**

prf

Use **prf**to specify pseudo-random function (PRF) algorithms for an IKEv2 proposal.

Use **undoprf**to restore the default.

Syntax

In non-FIPS mode:

prf { **aes-xcbc-mac** | **md5** | **sha1** |**sha256** | **sha384** | **sha512** } *

undoprf

In FIPS mode:

prf { **sha1 | sha256** | **sha384** | **sha512** } *

undoprf

Default

An IKEv2 proposal uses the integrity protection algorithms as the PRF algorithms.

Views

IKEv2 proposal view

Predefined user roles

network-admin

mdc-admin

Parameters

aes-xcbc-mac: Uses the HMAC-AES-XCBC-MAC algorithm.

md5: Uses the HMAC-MD5 algorithm.

sha1: Uses the HMAC-SHA1 algorithm.

sha256: Uses the HMAC-SHA256 algorithm.

sha384: Uses the HMAC-SHA384 algorithm.

sha512: Uses the HMAC-SHA512 algorithm.

Usage guidelines

You can specify multiple PRF algorithms for an IKEv2 proposal. An algorithm specified earlier has a higher priority.

Examples

Create an IKEv2 proposal named **prop1**.

```
<Sysname> system-view
[Sysname] ikev2 proposal prop1
```

Specify HMAC-SHA1 and HMAC-MD5 as thePRF algorithms, with HMAC-SHA1 preferred.

```
[Sysname-ikev2-proposal-prop1] prf sha1 md5
```

Related commands

- **ikev2 proposal**
- **integrity**

priority (IKEv2 policy view)

Use **priority** to set a priority for an IKEv2 policy.

Use **undopriority** to restore the default.

Syntax

priority*priority*

undopriority

Default

The priority of an IKEv2 policy is 100.

Views

IKEv2 policy view

Predefined user roles

network-admin

mdc-admin

Parameters

priority: Specifies the priority of the IKEv2 policy, in the range of 1 to 65535. A smaller number represents a higher priority.

Usage guidelines

The priority set by this command can only be used to adjust the match order of IKEv2 policies.

Examples

#Set the priority to 10 for the IKEv2 policy **policy1**.

```
<Sysname> system-view
[Sysname] ikev2 policy policy1
[Sysname-ikev2-policy-policy1] priority 10
```

Related commands

display ikev2 policy

priority (IKEv2 profile view)

Use **priority** to set a priority for an IKEv2 profile.

Use **undopriority** to restore the default.

Syntax

priority*priority*

undopriority

Default

The priority of an IKEv2 profile is 100.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

priority: Specifies the priority of the IKEv2 profile, in the range of 1 to 65535. A smaller number represents a higher priority.

Usage guidelines

The priority set by this command can only be used to adjust the match order of IKEv2 profiles.

Examples

#Set the priority to 10 for the IKEv2 profile **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
[Sysname-ikev2-profile-profile1] priority 10
```

proposal

Use **proposal**to specify an IKEv2 proposal for an IKEv2 policy.

Use **undo proposal**to remove an IKEv2 proposal from an IKEv2 policy.

Syntax

proposal*proposal-name*

undo proposal*proposal-name*

Default

No IKEv2 proposal is specified for an IKEv2 policy.

Views

IKEv2 policy view

Predefined user roles

network-admin

mdc-admin

Parameters

proposal-name: Specifies an IKEv2 proposal by its name, a case-insensitive string of 1 to 63 characters.

Usage guidelines

You can specify multiple IKEv2 proposals for an IKEv2 policy. A proposal specified earlier has a higher priority.

Examples

Specify the IKEv2 proposal **proposal1** for the IKEv2 policy **policy1**.

```
<Sysname> system-view
[Sysname] ikev2 policy policy1
```

```
[Sysname-ikev2-policy-policy1] proposal proposal1
```

Related commands

- **display ikev2 policy**
- **ikev2 proposal**

reset ikev2 sa

Use **reset ikev2sa** to delete IKEv2 SAs.

Syntax

reset ikev2 sa[[{ **local** | **remote** } { *ipv4-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*]] | **tunnel***tunnel-id*] [**fast**]

Views

User view

Predefined user roles

network-admin

mdc-admin

Parameters

local: Deletes IKEv2 SAs for a local IP address.

remote: Deletes IKEv2 SAs for a remote IP address.

ipv4-address: Specifies a local or remote IPv4 address.

ipv6*ipv6-address*: Specifies a local or remote IPv6 address.

vpn-instance*vpn-instance-name*: Deletes IKEv2 SAsin an MPLS L3VPN instance. The *vpn-instance-name*argument represents the VPN instance name, a case-sensitive string of 1 to 31 characters. If you do not specify a VPN instance, this command deletes IKEv2 SAs for the public network.

tunnel*tunnel-id*: Deletes IKEv2 SAs for an IPsec tunnel. The *tunnel-id* argument specifies an IPsec tunnel by its ID in the range of 1 to 2000000000.

fast: Notifies the peers of the deletion and deletes IKEv2 SAs directly before receiving the peers' responses. If you do not specify this keyword, the device notifies the peers of the deletion and deletes IKEv2 SAs after it receives the peers' responses.

Usage guidelines

Deleting an IKEv2 SA will also delete the child SAs negotiated through the IKEv2 SA.

If you do not specify any parameters, this command deletes all IKEv2 SAs and the child SAs negotiated through the IKEv2 SAs.

Examples

Display information about IKEv2 SAs.

```
<Sysname> display ikev2 sa
     Tunnel ID          Local             Remote             Status
  ----------------------------------------------------------------------
     1                  1.1.1.1/500       1.1.1.2/500        EST
     2                  2.2.2.1/500       2.2.2.2/500        EST
  Status:
  IN-NEGO: Negotiating EST: Established, DEL: Deleting
```

Delete the IKEv2 SA whose remote IP address is 1.1.1.2.

```
<Sysname> reset ikev2saremote 1.1.1.2
```

Display information about IKEv2 SAs again. Verify that the IKEv2 SA is deleted.

```
<Sysname> display ikev2 sa
     Tunnel ID          Local             Remote             Status
  ------------------------------------------------------------------------
     2                  2.2.2.1/500       2.2.2.2/500        EST
  Status:
  IN-NEGO: Negotiating EST: Established, DEL: Deleting
```

Related commands

display ikev2 sa

sa duration

Use **sa duration** to set the IKEv2 SA lifetime.

Use **undo sa duration** to restore the default.

Syntax

sa duration*seconds*

undo sa duration

Default

The IKEv2 SA lifetime is 86400 seconds.

Views

IKEv2 profile view

Predefined user roles

network-admin

mdc-admin

Parameters

seconds: Specifies the IKEv2 SA lifetime in seconds, in the range of 120 to 86400.

Usage guidelines

An IKEv2 SA can be used for subsequent IKEv2 negotiations before its lifetime expires, saving a lot of negotiation time. However, the longer the lifetime, the higher the possibility that attackers collect enough information and initiate attacks.

Two peers can have different IKEv2 SA lifetime settings, and they do not perform lifetime negotiation. The peer with a shorter lifetime always initiates the rekeying.

Examples

Create an IKEv2 profile named **profile1**.

```
<Sysname> system-view
[Sysname] ikev2 profile profile1
```

Set the IKEv2 SA lifetime to 1200 seconds.

```
[Sysname-ikev2-profile-profile1] sa duration 1200
```

Related commands

display ikev2 profile

Modified feature: SSHv2

Feature change description

None

Command changes

Modified command: ssh user *username*

Syntax

ssh user *username* **service-type { all | netconf | scp | sftp | stelnet } authentication-type { password | { any | password-publickey | publickey } assign { pki-domain***domain-name***| publickey***keyname* **} }**

Views

System view

Change description

Before modification:

- *username*: Specifies an SSH username, a case-sensitive string of 1 to 80 characters. If the username contains an ISP domain name, use the format *pureusername@domain*. The *pureusername* argument is a string of 1 to 55 characters. The *domain* argument is a string of 1 to 24 characters.

After modification:

- *username: Specifies an SSH username, a case-sensitive string of 1 to 80 characters.The name cannot contain “\”, ”|”, ”/’, ”:”, “*”,”?”,”<”,”>” and “@”.The name can not be “a”,”al”, “all”. If the username contains an ISP domain name, use the format pureusername@domain.The pureusername argument is a string of 1 to 55 characters. The domain argument is a string of 1 to 24 characters.*

Release HP VSR1000_HP-CMW710-E0321-X64

This release has the following changes:

New feature:Intelligent Resilient Framework (IRF)

Intelligent Resilient Framework (IRF) technology virtualizes two devices at the same layer into one virtual fabric to provide data center class availability and scalability. IRF virtualization technology offers processing power, interaction, unified management, and uninterrupted maintenance of two devices.

Configuring Intelligent Resilient Framework (IRF)

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature: Reth Interface and Redundancy Group

A redundant Ethernet (Reth) interface is a virtual Layer 3 interface that uses two member interfaces to ensure link availability. One member interface is active and the other is inactive. When the active interface fails, the inactive interface becomes active. The member interface switchover does not interrupt traffic.

A redundancy group has two redundancy group nodes. Each node is bound to an IRF member device. The redundancy group contains members that are on the bound IRF member devices. The members include Reth interfaces, failover groups, and interfaces of the IRF member devices.

A redundancy group node is a collection of objects on its bound IRF member device. The objects include member interfaces of Reth interfaces, CPUs of failover groups, and member interfaces of the redundancy group. The state of the objects is consistent with the state of the node.

In a redundancy group, one node is in primary state, and the other node is in secondary state. Only the primary node forwards traffic. When the primary node fails, the redundancy group switches over to the secondary node. This mechanism ensures path symmetry for traffic.

If an IRF fabric contains more than two member devices, you can configure multiple independent redundancy groups. The node state and switchovers in a redundancy group does not affect another redundancy group.

Configuring Reth Interface and Redundancy Group

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature:Address Family Translation (AFT)

Address Family Translation (AFT) translates an IP address of one address family into an IP address of the other address family. It enables an IPv4 network and IPv6 network to communicate with each other.

Configuring Address Family Translation (AFT)

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature:Monitor Link

Monitor Link associates the state of downlink interfaces with the state of uplink interfaces in a monitor link group. When Monitor Link shuts down the downlink interfaces because of an uplink failure, the downstream device changes connectivity to another link.

Configuring Monitor Link

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature:Service Chain

Service chain is a forwarding and routing technology used to guide network service packets through service nodes. It is based on the Overlay technology and combines the software defined network (SDN) centralized management theory. You can configure service chains by using an H3C virtual converged framework controller (VCFC). For more information about H3C VCF controllers, see the VCF controller configuration guide.

Service chain implements the following functions:

- Decoupling the tenant logical network and the physical network, and separating the control plane from the forwarding plane.
- Service resource allocation and deployment on demand with no physical topology restrictions.
- Dynamic creation and automatic deployment of network function virtualization (NFV) resource pools.
- Tenant-specific service arrangement and modification without affecting the physical topology and other tenants.

Configuring Service Chain

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: Layer 7 Load Balancing

Layer 7 server load balancing—Identifies network layer, transport layer, and application layer information, and is implemented based on contents. It analyzes packet contents, distributes packets one by one based on the contents, and distributes connections to the specified server according to the predefined policies.

Configuring Layer 7 Load Balancing

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: Outbound Link Load Balancing

Outbound link load balancing load balances traffic among the links from the internal network to the external network. .

Configuring Outbound Link Load Balancing

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: WAAS

The Wide Area Application Services (WAAS) feature is a set of services that can optimize WAN traffic. WAAS solves WAN issues such as high delay and low bandwidth by using optimization services. WAAS provides the following optimization services:

- Transport Flow Optimization (TFO).
- Data Redundancy Elimination(DRE).
- Lempel-Ziv compression(LZcompression).

Configuring WAAS

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: VXLAN

Virtual eXtensible LAN (VXLAN) is a MAC-in-UDP technology that provides Layer 2 connectivity between distant network sites across an IP network. VXLAN is typically used in data centers for multitenant services.

VXLAN provides the following benefits:

- Support for more virtual switched domains than VLANs—Each VXLAN is uniquely identified by a 24-bit VXLAN ID. The total number of VXLANs can reach 16777216 (224). This specification makes VXLAN a better choice than 802.1Q VLAN to isolate traffic for VMs.
- Easydeployment andmaintenance—VXLAN requires deployment only on the edge devices of the transport network. Devices in the transport network perform typical Layer 3 forwarding.

The device supports only IPv4-based VXLAN. IPv6-based VXLAN is not supported.

Configuring VXLAN

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: ADVPN

Auto Discovery Virtual Private Network (ADVPN) enables enterprise branches that use dynamic public addresses to establish a VPN network. ADVPN uses the VPN Address Management (VAM) protocol to collect, maintain, and distribute dynamic public addresses.

VAM uses the client/server model. All VAM clients register their public addresses on the VAM server. A VAM client obtains the public addresses of other clients from the server to establish ADVPN tunnels.

Configuring ADVPN

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: LLDP

The Link Layer Discovery Protocol (LLDP) is specified in IEEE 802.1AB. The protocol operates on the data link layer to exchange device information between directly connected devices. With LLDP, a device sends local device information as TLV (type, length, and value) triplets in LLDP Data Units (LLDPDUs) to the directly connected devices. Local device information includes its system capabilities, management IP address, device ID, port ID, and so on. The device stores the device information in LLDPDUs from the LLDP neighbors in a standard MIB. For more information about MIBs, see *Network Management and Monitoring Configuration Guide*. LLDP enables a network management system to quickly detect and identify Layer 2 network topology changes.

Configuring LLDP

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: OpenFlow

OpenFlow is the communications interface defined between the control and forwarding layers of a Software-Defined Networking architecture. With OpenFlow, you can perform centralized data forwarding management for physical and virtual devices through controllers.

OpenFlow separates the data forwarding and routing decision functions. It keeps the flow-based forwarding function and employs a separate controller to make routing decisions. An OpenFlow switch communicates with the controller through an OpenFlow channel. An OpenFlow channel can be encrypted by using TLS or run directly over TCP. VSR exchanges control messages with the controller through an OpenFlow channel to perform the following operations:

- Receive flow table entries or datafrom the controller.

- Report informationto the controller.

Configuring OpenFlow

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: DCN

Feature change description

Data communication network (DCN) is built for the network management system (NMS) to implement operation, administration, and maintenance (OAM) on the network elements (NEs).

On large-scaled networks with DCN configured, the NMS remotely manages and controls all NEs through the gateway network element (GNE), which reduces operation and maintenance costs.

On the DCN network, all NEs must operate in Area 0 of OSPF process 65535.

Configuring DCN

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Newfeature: MAC-based quick portal authentication

Feature change description

MAC-based quick portal authentication is applicable to scenarios where users access the network frequently. It allows users to pass authentication without entering username and password. MAC-based quick portal authentication is also called MAC-trigger authentication or transparent portal authentication.

A MAC binding server is required for MAC-trigger authentication. The MAC binding server records the MAC-to-account bindings of portal users for authentication. The account contains the portal authentication information of the user, including username and password.

Configuring MAC-based quick portal authentication

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Modified feature: BGP summary route redistribution

Feature change description

Before modification: The summary route generated by a protocol cannot be redistributed by another protocol.

After modification: The summary route generated by a protocol can be redistributed by another protocol.

Upgrade impact: Determine whether route summarization has been configured and the summary route has been redistributed by another protocol. If route summarization and redistribution are configured, the summary route will be advertised and routing loops might occur. If they are not configured, the summary route will not be advertised.

Command changes

None

Modified feature: EBGP peer relationship establishment

Feature change description

Before modification: For BGP to use a loopback interface to establish an EBGP session, you only need to execute the **peer connect-interface** command. You do not need to execute the **peer ebgp-max-hop** command.

After modification: For BGP to use a loopback interface to establish an EBGP session, you must execute the **peer connect-interface** command and the **peer ebgp-max-hop** command.

Upgrade impact: After the upgrade, execute the**peer ebgp-max-hop** command for the peers to which BGP has established EBGP sessions through loopback interfaces.

Command changes

None

Modified feature: SSH

Feature change description

Before modification: The SSH2 server supports SSH1 clients by default.

After modification: For security purposes, the SSH2 server does not support SSH1 clients by default.

Upgrade impact: To enable the SSH2 server to support SSH1 clients after the upgrade, use the **ssh server compatible-ssh1x enable** command.

Command changes

None

Modified feature: EVI

Feature change description

None

Command changes

Modified command: gre key

Old syntax

gre key [**key-number** | **network-id**]

New syntax

gre key [**key-number** | **vlan-id**]

Views

Tunnel interface view

Change description

The **network-id** keyword was deleted, and the **vlan-id** keyword was added. An EVI tunnel interface can generate a GRE key for tunneled packets based on their VLAN IDs.

Modified feature: ACL

Feature change description

Before modification: The **packet-filter** command in interface view supports the **hardware-count** keyword for rule match counting. You do not need to specify the **counting** keyword in the**rule** command.

After modification: The **packet-filter** command in interface view does not support the **hardware-count** keyword. You must specify the **counting** keyword in the**rule** command to enable rule match counting.

Upgrade impact: None.

Command changes

None

Modified feature: AAA

Feature change description

None

Command changes

Modified command: authorization-attribute

Syntax

authorization-attribute { **acl***acl-number* | **callback-number***callback-number* | **idle-cut***minute* | **ip***ipv4-address* | **ipv6***ipv6-address* | **ipv6-pool***ipv6-pool-name* | **ipv6-prefix***ipv6-prefix prefix-length* | { **primary-dns** | **secondary-dns** } { **ip***ip-address* | **ipv6***ipv6-address* } | **url***url-string* | **user-profile***profile-name* | **user-role***role-name* | **vlan***vlan-id* | **vpn-instance***vpn-instance-name* | **work-directory***directory-name* } *

Views

Local user view

User group view

Change description

Before modification:

- The *profile-name* argument is a case-sensitive string of 1 to 31 characters.
- The maximum length of the *directory-name* argument is 512 characters.

After modification:

- The*profile-name* argument is a case-sensitive string of 1 to 31 characters.The name can contain only letters, digits, and underscores (_), and it must start with a letter.
- The maximum length of the *directory-name* argument is 255 characters.

Modified feature: NTP

Feature change description

None

Command changes

Modified command: ntp-service inbound

Old syntax

ntp-service inbound disable

undo ntp-service inbound disable

New syntax

ntp-service inbound enable

undo ntp-service inbound enable

Views

Interface view

Change description

Usentp-service inbound enableto enable an interface to process NTP messages.

Useundontp-service inbound enableto disable an interface from processing NTP messages.

Modified command: ntp-service ipv6 inbound

Old syntax

ntp-service ipv6 inbound disable

undo ntp-service ipv6 inbound disable

New syntax

ntp-service ipv6 inbound enable

undo ntp-service ipv6 inbound enable

Views

Interface view

Change description

Usentp-service ipv6 inbound enableto enable an interface to process IPv6 NTP messages.

Useundontp-service ipv6 inbound enableto disable an interface from processing IPv6 NTP messages.

Release HP VSR1000_HP-CMW710-E0301-X64

This release has the following changes:

New feature:OVF with Start-up Configuration Deployment

VSR OVF with Start-up Configuration Deployment can:

1. Provide GUI to input the basic VSR parameters (e.g. the management IP address, username, password, Telnet/SSH/SNMP, etc.) while deploying VSR with VM tool (e.g., OVA in VMware vCenter). Then you can directly manage VSR remotely (e.g. SSH, NMS) after the VSR boots up.

2. Provide a script tool to remotely deploy a VSR on a specific server with VM profile (e.g. vCPU, vMemory, vNIC, etc.) and basic VSR parameters. You can also customize VSR OVA based on the released VSR OVA. Batch auto-deployment can also be achieved by leveraging the script tool.

Configuring OVF with Start-up Configuration Deployment

Please refer to the Virtual Services Router Installation and Getting Started Guide.

Command reference

None

New feature:Ethernet Virtual Interconnect (EVI) Support

EVI is a MAC-in-IP technology that provides Layer 2 connectivity between distant Layer 2 network sites across an IP routed network. It is used for connecting geographically dispersed sites of a virtualized large-scale data center that requires Layer 2 adjacency. EVI enables long-distance virtual machine workload mobility and data mobility, disaster recovery, and business continuity. For example, virtual machines can move between data center sites without changing their IP addresses, so their movements are transparent to users and do not disrupt traffic.

NOTE: To usethis feature, please purchase and install a suitableDCI feature license.

Configuring Ethernet Virtual Interconnect (EVI)

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature: Multi-Control Plane Support

With this release, you assign more CPU resource to the control plane applications.

NOTE: To usethis feature, please purchase and install a suitableECP feature license.

Configuring Multi-Control Plane

None

Command reference

None

New feature:Multi-Topology Routing (MTR) Support

MTR splits a base topology into multiple topologies, which intersect or overlap with one another. Route calculation is performed on a per-topology basis. For example, IS-IS MTR enables IS-IS to perform separate route calculation for the IPv4 and IPv6 topologies in an IS-IS routing domain.

Configuring Multi-Topology Routing (MTR)

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

New feature:Zone-based Security Management Support

The traditional security management technology is based on interfaces. You must apply security policies on the inbound and outbound interfaces of a firewall to filter packets. This traditional technology has the following problems:

- When the firewall is connected to multiple network segments, deploying security policies is time consuming and complicated.
- If the network topology changes, you might have to reconfigure security policies.

The security zone-based security management solved the above problems, and is better when the firewall is connected to multiple network segments or the network topology might change. It includes security zone, object group, and object policy.

A security zone is a collection of interfaces that have the same security requirements.

An object group is a group of objects that can be referenced by an ACL, object policy, or object group to identify packets.

An object policy is a set of rules for security control over packets between a source and a destination security zone. These two zones define an interzone instance. The object policy matches the first packet of a traffic flow against the rules. If a match is found, the device stops the match process and takes the action defined in the rule over the packet and all subsequent packets of the flow.

Configuring Zone-based Security Management

Please refer to the Virtual Services Router Configuration Guides.

Command reference

Please refer to the Virtual Services Router Command References.

Modified feature: MPLS TE

Feature change description

None

Command changes

Modified command: MPLS TE

Old syntax

mpls te igp advertise [*hold-time value*]

undo mpls te igp advertise

New syntax

mpls te igp advertise

undo mpls te igp advertise

Views

Tunnel bundle interface view

Change description

The hold-time configuration is clipped for tunnel bundle interfaces, because hold-time does not apply for tunnel bundle interfaces.

Modified feature: Firewall-AttackDefend

Feature change description

None

Command changes

Modified command: blacklist ip source-ip-address

Old syntax

blacklist ip *source-ip-address* **[vpn-instance** *vpn-instance-name* **] [timeout** *minutes* **]**

New syntax

blacklist ip*source-ip-address* [**vpn-instance***vpn-instance-name*] [**ds-lite-peer***ds-lite-peer-address*] [**timeout***minutes*]

Views

System view

Change description

Config blacklist.specifies the Ds-lite tunnel peer address to distinguish between different B4.

Modified command: reset blacklist ip

Old syntax

reset blacklist ip { *source-ip-address* [**vpn-instance***vpn-instance-name*] | **all** }

New syntax

reset blacklist ip { *source-ip-address* [**vpn-instance***vpn-instance-name*] [**ds-lite-peer** *ds-lite-peer-address*] | **all** }

Views

System view

Change description

Reset blacklist, specifies the Ds-lite tunnel peer address to distinguish between different B4.

Modified command: XXX-flood detect

Old syntax

XXX-flood detect { **ip***ip-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*] [**threshold** *rate-number*] [**action** { **client-verify** | **drop** | **logging** } *]

New syntax

XXX-flood detect { **ip***ip-address* | **ipv6***ipv6-address* } [**vpn-instance***vpn-instance-name*] [**threshold***rate-number*] [**action** { { **client-verify** | **drop** | **logging** } * } | **none**]

Views

Attack defense policy view

Change description

Specifies the none action when a XXX flood attack is detected. Takes no action.

Modified feature: FIB4/FIB6 Feature

Feature change description

None

Command changes

Modified command: apply next-hop

Old syntax

apply next-hop [**vpn-instance***vpn-instance-name* | **inbound-vpn**] { *ip-address* [**direct**] [**track** track-entry-number] }&<1-n>

New syntax

apply next-hop [vpn-instance vpn-instance-name | inbound-vpn] { ip-address [direct] [track *track-entry-number*] [**weight***weight-value*] }&<1-n>

Views

Policy node view

Change description

Specifies a weight-value for load sharing among multiple next hops.

Modified feature: BGP Feature

Feature change description

None

Command changes

Modified command: peer

Old syntax

peer { *ip-address* | *ipv6-address* } **as-number***as-number*

New syntax

peer{ *ip-address* [*mask-length*] | *ipv6-address* [*prefix-length*] } **as-number***as-number*

Views

BGP view, BGP-VPN instance view

Change description

The ip-address and mask-length (ipv6-address and prefix-length)arguments together specify a network, representing all dynamic peers in the network.

Modified feature: Interface Feature

Feature change description

None

Command changes

Modified command: ip address

Old syntax

ip address ip-address { *mask-length* | *mask* } [**sub**]

New syntax

ip address ip-address { *mask-length* | *mask* } [**sub**]

Views

Interface view

Change description

The total number of IP addresses for one interface can be changed to 256.

Modified feature: APR and NAT feature

Feature change description

Some services' name has been unified, the services showed as Table 6:

Table 6 The services' name change list

Service	Origin Name	Other Name	New Name
Domain Name Server	dns	domain	dns
IBMLotusNotes	notes	lotusnote	notes
RoutingInformationProtocol	rip	router	rip
MicrosoftRPC	msrpc	epmap	msrpc
H.323CallSetup	h225	h323hostcall	h225
MSNMessenger	msn-messenger	msnp	msn-messenger
H.323GatekeeperRegistration AdmissionStatus	ras	h323gatestat	ras
RemoteShellCommands	rsh	shell	rsh
SkinnyClientControlProtocol	sccp	cisco-sccp	sccp
XWindowSystem	x11	xwindows	x11
TencentQQ	irdmi	qq	irdmi
SQL*NETforOracle	sqlnet	ncube-lm	sqlnet

Command changes

Modified command: display port-mapping

Old syntax

display port-mapping pre-defined

Views

User view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: include application

Old syntax

include application*application-name*

Views

Application group view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: port-mapping

Old syntax

port-mapping application*application-name***port** *port-number* [**protocol***protocol-name*]

undoport-mapping application*application-name***port** *port-number* [**protocol***protocol-name*]

Views

System view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: port-mapping acl

Old syntax

port-mapping application*application-name***port***port-number* [**protocol** *protocol-name*] **acl** [**ipv6**] *acl-number*

undoport-mapping application*application-name***port***port-number* [**protocol** *protocol-name*] **acl** [**ipv6**] *acl-number*

Views

System view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: port-mapping host

Old syntax

port-mapping application*application-name* **port***port-number* [**protocol***protocol-name*] **host** { **ip** | **ipv6** } *start-ip-address* [*end-ip-address*] [**vpn-instance***vpn-instance-name*]

undoport-mapping application*application-name* **port***port-number* [**protocol***protocol-name*] **host** { **ip** | **ipv6** } *start-ip-address* [*end-ip-address*] [**vpn-instance***vpn-instance-name*]

Views

System view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: port-mapping subnet

Old syntax

port-mapping application*application-name***port***port-number* [**protocol***protocol-name*] **subnet** { **ip***ipv4-address* { *mask-length* | *mask* } | **ipv6***ipv6-addressprefix-length* } [**vpn-instance***vpn-instance-name*]

undoport-mapping application*application-name***port***port-number* [**protocol***protocol-name*] **subnet** { **ip***ipv4-address* { *mask-length* | *mask* } | **ipv6***ipv6-addressprefix-length* } [**vpn-instance***vpn-instance-name*]

Views

System view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: nat server

Old syntax

Common internal server:

- A single public address with no or a single public port:

  nat server protocol *pro-type***global** { *global-address* | **current-interface** | *interface interface-type interface-number* } [*global-port*] [**vpn-instance***global-name*] **inside***local-address* [*local-port*] [**vpn-instance***local-name*] [**acl***acl-number*]

  undonat server protocol *pro-type***global** { *global-address* | **current-interface** | *interface interface-type interface-number* } [*global-port*] [**vpn-instance***global-name*] **inside***local-address* [*local-port*] [**vpn-instance***local-name*]

- A single public address with consecutive public ports:

  nat server protocol*pro-type***global** { *global-address* | **current-interface** | *interface interface-type interface-number* } *global-port1 global-port2* [**vpn-instance***global-name*] **inside** { { *local-address* | *local-address1 local-address2* } *local-port* | *local-address local-port1 local-port2* } [**vpn-instance***local-name*] [**acl***acl-number*]

  undo nat server protocol*pro-type***global** { *global-address* | **current-interface** | **interface***interface-type interface-number* } *global-port1 global-port2* [**vpn-instance***global-name*]

- Consecutive public addresses with no or a single public port:

  nat server protocol*pro-type* **global***global-address1 global-address2* [*global-port*] [**vpn-instance***global-name*] **inside** { *local-address* | *local-address1 local-address2* } [*local-port*] [**vpn-instance***local-name*] [**acl***acl-number*]

  undo nat server protocol*pro-type***global***global-address1 global-address2* [*global-port*] [**vpn-instance***global-name*]

- Consecutive public addresses with a single public port:

nat server protocol*pro-type***global***global-address1 global-address2 global-port* [**vpn-instance***global-name*] **inside***local-address local-port1 local-port2* [**vpn-instance***local-name*] [**acl** *acl-number*]

undonat server protocol*pro-type***global***global-address1 global-address2 global-port* [**vpn-instance***global-name*] **inside***local-address local-port1 local-port2* [**vpn-instance***local-name*]

Load sharing internal server:

nat server protocol*pro-type***global** { { *global-address* | **current-interface** | **interface** *interface-type interface-number* } { *global-port* | *global-port1 global-port2* } | *global-address1 global-address2 global-port* } [**vpn-instance***global-name*] **insideserver-group***group-number* [**vpn-instance***local-name*] [**acl***acl-number*]

undonat server protocol*pro-type***global** { { *global-address* | **current-interface** | **interface** *interface-type interface-number* } { *global-port* | *global-port1 global-port2* } | *global-address1 global-address2 global-port* } [**vpn-instance***global-name*] **insideserver-group***group-number* [**vpn-instance***local-name*]

Views

Interface view

Change description

Some services' name has been changed. For details, see Table 6.

Modified command: nat dns-map

Old syntax

nat dns-map domain*domain-name***protocol***pro-type* { **interface***interface-type interface-number* | **ip***global-ip* } **port***global-port*

Views

System view

Change description

Some services' name has been changed. For details, see Table 6.

Release HP VSR1000_HP-CMW710-R0202-X64

This release has the following changes:

- New feature:Intel 82599 VF Support
- New feature: VirtIO Disk Support
- New feature:VMware ESXi 5.5 Support
- Modified feature: None
- Removed feature: None

New feature:Intel 82599 VF Support

Configuring Intel 82599 VF Support

None

Command reference

None

New feature: VirtIO Disk Support

Configuring VirtIO Disk Support

None

Command reference

None

New feature:VMware ESXi 5.5 Support

Configuring VMware ESXi 5.5 Support

None

Command reference

None

Release HP VSR1000_HP-CMW710-E0102-X64

This release is the maintenance version for VSR1000_HP-CMW710-E0101P01-X64. There is no software feature have been changed.

Release HP VSR1000_HP-CMW710-E0101P01-X64

This release is the first release of VSR software.