53-1003713-03
14 September 2015

Brocade 5600 vRouter IPsec Site-to-Site VPN

Reference Guide

Supporting Brocade 5600 vRouter 3.5R6

BROCADE

Contents

Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Brocade technical documentation.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used in the flow of the text to highlight specific words or phrases.

Format	Description
bold text	Identifies command names
	Identifies keywords and operands
	Identifies the names of user-manipulated GUI elements
	Identifies text to enter at the GUI
italic text	Identifies emphasis
	Identifies variables
	Identifies document titles
`Courier font`	Identifies CLI output
	Identifies command syntax examples

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
value	In Fibre Channel products, a fixed value provided as input to a command option is printed in plain text, for example, **--show** WWN.

Convention	Description
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options. In Fibre Channel products, square brackets may be used instead for this purpose.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a “soft” line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Brocade resources

Visit the Brocade website to locate related documentation for your product and additional Brocade resources.

You can download additional publications supporting your product at www.brocade.com. Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to MyBrocade. You can register at no cost to obtain a user ID and password.

Release notes are available on MyBrocade under Product Downloads.

White papers, online demonstrations, and data sheets are available through the Brocade website.

Contacting Brocade Technical Support

As a Brocade customer, you can contact Brocade Technical Support 24x7 online, by telephone, or by e-mail. Brocade OEM customers contact their OEM/Solutions provider.

Brocade customers

For product support information and the latest information on contacting the Technical Assistance Center, go to http://www.brocade.com/services-support/index.html.

If you have purchased Brocade product support directly from Brocade, use one of the following methods to contact the Brocade Technical Assistance Center 24x7.

Online	Telephone	E-mail
Preferred method of contact for non-urgent issues: • My Cases through MyBrocade • Software downloads and licensing tools • Knowledge Base	Required for Sev 1-Critical and Sev 2-High issues: • Continental US: 1-800-752-8061 • Europe, Middle East, Africa, and Asia Pacific: +800-AT FIBREE (+800 28 34 27 33) • For areas unable to access toll free number: +1-408-333-6061 • Toll-free numbers are available in many countries.	support@brocade.com Please include: • Problem summary • Serial number • Installation details • Environment description

Brocade OEM customers

If you have purchased Brocade product support from a Brocade OEM/Solution Provider, contact your OEM/Solution Provider for all of your product support needs.

- OEM/Solution Providers are trained and certified by Brocade to support Brocade® products.
- Brocade provides backline support for issues that cannot be resolved by the OEM/Solution Provider.

- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information, contact Brocade or your OEM.
- For questions regarding service levels and response times, contact your OEM/Solution Provider.

Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. You can provide feedback in two ways:

- Through the online feedback form in the HTML documents posted on www.brocade.com.
- By sending your feedback to documentation@brocade.com.

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

About This Guide

This guide describes how to configure site-to-site IPsec VPNs on the Brocade 5600 vRouter (referred to as a virtual router, vRouter, or router in the guide).

IPsec VPN Overview

Benefits of IPsec VPNs

An IPsec Virtual Private Network (VPN) is a virtual network that operates across the public network, but remains “private” by establishing encrypted tunnels between two or more end points. VPNs provide:

- Data integrity: Data integrity ensures that no one has tampered with or modified data while it traverses the network. Data integrity is maintained with hash algorithms.
- Authentication: Authentication guarantees that data you receive is authentic; that is, that it originates from where it is supposed to, and not from someone masquerading as the source. Authentication is also ensured with hash algorithms.
- Confidentiality: Confidentiality ensures data is protected from being examined or copied while transiting the network. Confidentiality is accomplished using encryption.

An IP Security (IPsec) VPN secures communications and access to network resources for site-to-site access using encryption, authentication, and key management protocols. On a properly configured VPN, communications are secure, and the information that is passed is protected from attackers.

The Brocade vRouter currently supports site-to-site IPsec VPN connectivity on both IPv4 and IPv6 networks (IPv4 traffic over IPv4 IPsec tunnels, and IPv6 traffic over IPv6 IPsec tunnels). Site-to-site VPN connections are normally established between two (or more) VPN gateways and provide connectivity for user hosts, servers, and other devices at each location. Connectivity is normally based on IP source and destination network pairs, allowing multiple hosts to share the same tunnel between locations.

Site-to-site VPNs enable enterprises to create low-cost connectivity between offices. These site-to-site VPNs frequently replace more expensive WAN technologies such as private lines or Frame Relay.

IPsec architecture

IPsec is a suite of protocols designed to provide end-to-end security at the network layer (Layer 3), using encryption and authentication techniques. From the point of view of IP networking equipment, encrypted packets can be routed just like any other ordinary IP packets. The only devices that require an IPsec implementation are the IPsec endpoints.

There are three main components of the IPsec architecture. These are:

- The Authentication Header (AH) protocol
- The Encapsulating Security Payload (ESP) protocol
- The Internet Key Exchange (IKE) protocol, formerly referred to as ISAKMP/Oakley

Of these, the Brocade vRouter currently supports ESP, which encrypts the packet payload and prevents it from being monitored, and IKE (IKEv1), which provides a secure method of exchanging cryptographic keys and negotiating authentication and encryption methods.

The set of IPsec parameters describing a connection is called a security policy. The security policy describes how both endpoints will use security services, such as encryption, hash algorithms, and Diffie-Hellman groups, to communicate securely.

The IPsec peers negotiate a set of security parameters, which must match on both sides. Then they create a security association (SA). An IPsec SA describes the connection in one direction. For packets to travel in both directions in a connection, both an inbound and an outbound SA are required.

IPsec phase 1 and phase 2

The establishment of an IPsec connection takes place in two phases, called IKE phases:

- In IKE Phase 1, the two endpoints authenticate one another and negotiate keying material. This results in an encrypted tunnel used by Phase 2 for negotiating the ESP security associations.
- In IKE Phase 2, the two endpoints use the secure tunnel created in Phase 1 to negotiate ESP SAs. The ESP SAs are what are used to encrypt the actual user data that is passed between the two endpoints.

IKE Phase 1 establishes an ISAKMP SA (typically called an IKE SA). The IKE protocol is used to dynamically negotiate and authenticate keying material and other security parameters required to provide secure communications. IKE itself uses a combination of four protocols (including ISAKMP and Oakley) to dynamically manage keys in the context of IPsec.

If the IKE Phase 1 negotiation is successful, then the ISAKMP SA is established. The ISAKMP SA essentially contains the information from the “winning proposal” of the negotiation, recording the security encryption and keying material that was successfully negotiated. This creates a secure “control channel” where keys and other information for protecting Phase 2 negotiation are maintained. The ISAKMP SA encrypts only Phase 2 ESP security association negotiations, plus any IKE messages between the two endpoints.

An ISAKMP SA is maintained for a pre-determined lifetime. This lifetime is configured, not negotiated or passed between peers. The configured lifetime may be different between peers. When the configured lifetime expires, a new ISAKMP SA is negotiated.

IKE Phase 2 negotiations are also managed by the IKE protocol. Using the encryption provided by the security association, the security policy is used to try and negotiate a Phase 2 SA. The security policy includes information about the communicating hosts and subnets, as well as the ESP information for providing security services for the connection, such as encryption cipher and hash algorithm. If the IKE Phase 2 negotiation process is successful, a pair of ESP SAs (typically called IPsec SAs) is established—one inbound and one outbound—between the two endpoints. This is the encrypted VPN

"tunnel" between the two endpoints. At this point, the user data can be exchanged through the encrypted tunnel.

Between any two IPsec VPN peers, there can be just one control channel for exchanging Phase 2 keying material. This means that between any two peers there will be just one ISAKMP SA on each peer.

However, between two VPN peers, any number of security policies can be defined. For example, you can define a security policy that creates a tunnel between two hosts, and a different security policy that creates a tunnel between a host and a subnet, or between two subnets. Since multiple tunnels can exist between two peers, this means that multiple IPsec SAs can be active at any time between two peers.

IKE key exchange

To be able to create an ISAKMP SA, the two devices must agree on all of the following:

- The encryption algorithm
- The bit-strength of the encryption key (Diffie-Hellman group)
- The authentication method
- The hash algorithm
- The authentication material (pre-shared secret)

All of this information is contained in an IKE Phase 1 proposal. A VPN gateway can be configured multiple Phase 1 proposals. Note that the SA lifetime is not negotiated.

During an IKE key exchange, one device (the initiator) sends the first packet in the exchange. This first packet consists of all the Phase 1 proposals configured for this VPN peer, in a sequence. This set of proposals informs the other gateway of what security and authentication policies it supports. The second device (the responder) inspects the set of proposals and returns the policy representing strongest security policy that both devices can agree on. If this process is successful, both devices agree on the parameter and the ISAKMP SA is established.

Once the ISAKMP SA has been established, the two devices can use this SA to encrypt the Phase 2 traffic where the two endpoints try to negotiate an IPsec SA for each matching security policy that has been configured between the two endpoints. Only after the IPsec SAs have been established can IPsec traffic be passed.

Different devices initiate IKE negotiation differently. Many VPN devices bring up VPN tunnels only on demand. These devices monitor traffic to see if it is "interesting"—that is, to see if it matches a configured security policy. Once the device receives traffic matching a specific security policy, the device will attempt to negotiate an IPsec SA that will be used to encrypt that traffic.

Other devices, including the Brocade vRouter, will attempt to initiate Phase 2 negotiations as soon as a correct policy configuration is entered. If both endpoints behave in this way, a race condition can occur, where duplicate IPsec SAs are created.

Encryption ciphers

Ciphers are used to encrypt data, so that it cannot be read or monitored during transit. The Brocade vRouter supports the following encryption ciphers.

TABLE 1 Supported encryption ciphers

Cipher	Description
AES	The Advanced Encryption Standard (AES) is a U.S. government standard that was developed to take the place of DES, which has become easier to break by using the more powerful computers available today. AES can run very quickly for a block cipher and can be implemented in a relatively small space. It has a block length that varies between 192 and 256 bits, and a key length that ranges between 128 and 256 bits in increments of 32 bits. The Brocade vRouter supports AES with a 128-bit key and with a 256-bit key.
3DES	Triple-DES is a variant of the Data Encryption Standard (DES). DES was formerly the most commonly used cipher, but in recent years has been compromised and is no longer recommended as a first choice. The Brocade vRouter supports only Triple-DES. Triple-DES is an iterative block cipher in which DES is used in three consecutive iterations on the same block of text and either two or three keys are used. The resulting cipher text is much harder to break than DES. Using two keys yields 112-bits key strength; using three keys yields 168-bits key strength.

Hash algorithms

A hash function is a cryptographic algorithm that is used for message authentication. A hash function takes a message of arbitrary length and produces an output of fixed length, called a message digest or fingerprint. Hash functions are used to verify that messages have not been tampered with.

The Brocade vRouter supports the following hash functions.

TABLE 2 Supported hash functions

Cipher	Description
MD5	MD5 is the most recent version of message digest algorithm. MD5 takes a message of arbitrary length and produces a 128-bit condensed digital representation, called a message digest. It is often used when a large file must be compressed and encrypted, then signed with a digital signature. Message digest is quite fast and efficient compared with SHA-1 because it uses primitive operations and produces a shorter message. However, it is not as secure as SHA-1, and has reportedly been compromised in some ways, though not yet in ways that make it insecure.
SHA-1	SHA stands for Secure Hash Algorithm, also known as the Secure Hash Standard. The SHA hash functions are five one-way cryptographic algorithms for computing a message digest. SHA-1 is an extension of the original SHA, and is the standard hash algorithm supported by the U.S. government. SHA-1 takes a message of arbitrary length (the message must be smaller than 2^64 bits) and produces a 160-bit message digest. SHA-1 is slower than MD5, but it is more secure because the additional bits in the message digest provide more protection from brute-force attacks.

Pre-shared keys

A preshared secret, or pre-shared key (PSK), is a method of authentication. The secret, or key, is a character string agreed upon beforehand by both parties as the key for authenticating the session. It generates a hash such that each VPN endpoint can authenticate the other.

Note that the pre-shared secret, although an ordinary character string, is not a “password.” It actually generates a hashed key to form a fingerprint that proves the identity of each endpoint. This means that long, complex character strings are more secure than short strings. Choose complex pre-shared secrets and avoid short ones, which can be more easily compromised by an attack.

The preshared secret is not passed during IKE negotiation. It is configured on both sides, and must match on both sides.

A preshared secret is an example of symmetric cryptography: the key is the same on both sides. Symmetric encryption algorithms are less computationally intensive than asymmetric algorithms, and are, therefore, faster. However, in symmetric cryptography, the two communicating parties must exchange keys in advance. Doing this securely can be a problem.

A preshared secret and a digital signature are the most common methods of IKE authentication. A preshared secret is an easy and effective way to quickly set up authentication with little administrative overhead. However, it has several drawbacks.

- If a preshared key is captured and no one is aware of it, the attacker has access to your network as long as that key is in use.
- A preshared secret is manually configured, so it should be regularly changed. However, this task often falls off the list of busy network administrators. Using preshared key values with remote users is equivalent to giving them a password to your network.

NOTE
You should restrict the use of pre-shared keys to smaller, low-risk environments.

Digital signatures

Along with pre-shared key, RSA digital signatures are the most common means of IKE authentication.

An RSA digital signature is based on a cryptographic key that has two parts: a public part and a private part. One part (the public key) is widely shared, and may even be publicly distributed. The other part (the private key) remains secret. These keys are mathematically related but are independent, so that neither key is derivable from the other.

The key is used as input to a hash function; together, the key and the hash function form a signing function that, when applied to a document, creates a digital signature.

An RSA key can be used either to encrypt or authenticate, and this is based on two facts:

- Data encrypted with the agent's public key can only be decrypted by the agent, using the private key. This means that any peer can send information securely by encrypting it with the public key and forwarding it to the agent.
- Data processed with a hash function can be encrypted with the signer's private key—such data is said to be digitally signed. Since anyone with the public key can verify the digital signature, this communication can be accepted as authentically coming from the agent.

The algorithms that encrypt using RSA keys are very secure but extremely slow—so slow that it would be impracticable to encrypt an entire set of data using them. Instead, the agent produces a digital signature for the data, as follows:

1. A hash function is applied to the data to generate a message digest. The message digest is much shorter than the original data, and any peer possessing the same hash function can produce the identical message digest.
2. The private key is used to encrypt the message digest. This encrypted message digest is the digital signature.
3. The original message and the digital signature are all sent to the peer in an encrypted packet. (The encryption of the packet is independent of the digital signature.)
4. When the peer receives the packet, it decrypts the packet. Then it uses the sending agent's public key to decrypt the digital signature. This recovers the message digest.
5. The peer applies the hash function to the original message (which was also sent in the packet) and compares the resulting message digest to the message digest recovered from the digital signature.

   When the system generates an RSA digital signature, it stores it in a file. The file that contains the digital signature contains both the public key part and the private key part of the digital signature. When you view the RSA key, by looking at VPN configuration or by using the **show vpn ike rsa-keys** command, only the public key is displayed (along with any public keys configured for VPN peers). It is the public key that you should share with the other VPN peer.

   By default, the RSA digital signature file for the local host is stored in the /etc/ipsec.d/rsa-keys/localhost.key directory. When the key is required to authenticate the VPN peer, the system looks for the key in this directory. You can change the location and name of the file through configuration.

   You can have only one RSA digital signature configured for the local host. If you generate a new key, it overwrites the previous key.

   - If the message digests match, the peer can accept the communication as authentic.
   - If the message digests do not match, the peer must consider the communication to have been tampered with, or corrupted in some other way, and reject it.

Diffie-Hellman groups

Diffie-Hellman key exchange is a cryptographic protocol for securely exchanging encryption keys over an insecure communications channel, such as the Internet. Diffie-Hellman key exchange was developed in 1976 by Whitfield Diffie and Martin Hellman. It is based on two facts.

- Asymmetric encryption algorithms are much more secure than symmetric algorithms, which require that two parties exchange secret keys in advance.
- However, asymmetric algorithms are much slower and much more computationally expensive than symmetric algorithms.

In a Diffie-Hellman key exchange, asymmetric cryptography is used at the outset of the communication (IKE Phase 1) to establish a shared key. After the key has been exchanged, it can then be used symmetrically to encrypt subsequent communications (IKE Phase 2).

Diffie-Hellman key exchange uses a group of standardized global unique prime numbers and generators to provide secure asymmetric key exchange. The original specification of IKE defined four of these groups, called Diffie-Hellman groups or Oakley groups. Since then, a fifth has been defined.

The Brocade vRouter supports the following Diffie-Hellman groups.

TABLE 3 Supported Diffie-Hellman groups

Diffie-Hellman Group	Description
2	Diffie-Hellman group 2 is a modular exponentiation group (MODP). This group has a 1024-bit modulus.
5	Diffie-Hellman group 5 is a 1536-bit modular exponentiation (MODP) group. This group has a 1536-bit modulus.

IPsec modes

IPsec, in general, supports two modes of operation: *aggressive mode* and *main mode*.

Aggressive mode

Aggressive mode was created to reduce latency during Phase 1 negotiation but it is vulnerable to attack. For this reason, the Brocade vRouter does not support aggressive mode.

Main mode

Under ordinary conditions, establishing the ISAKMP SA requires several packets to be sent and received:

- The first two messages determine communications policy.
- The next two messages exchange Diffie-Hellman public data.
- The last two messages authenticate the Diffie-Hellman exchange.

This is the normal method of establishing a successful Phase 1 connection, and it is called *main mode*. This method provides the most security and privacy, because authentication information is not exchanged until a full Diffie-Hellman exchange has been negotiated and encryption has been enabled. The Brocade vRouter supports main mode.

Perfect forward secrecy

In Perfect Forward Secrecy (PFS), the private key is used to generate a temporary key (the session key) that is used for a short time and then discarded. Subsequent keys are independent of any previously created keys. This way, if a key is compromised, it does not affect any further keys, or compromise the security of data protected by other keys.

PFS provides a way to optimize both efficiently and security. Reasonably-sized keys are much more computationally efficient than large keys, but are also less secure. In PFS, you can use reasonably-sized keys and refresh them frequently.

Committing VPN configuration changes

An IPsec VPN connection includes multiple components, some of which are interdependent. For example, a VPN connection configuration requires a valid IKE group configuration, a valid ESP group configuration, and a valid tunnel configuration. In addition, the interface specified in the connection must be enabled for IPsec VPN. When you commit a VPN configuration, the Brocade vRouter performs a full verification on the configuration. If any required component is missing or incorrectly specified, the commit will fail.

For an IPsec VPN site-to-site connection configuration to successfully commit, all the following must be correctly configured:

- The interface and IP address must already be configured.
- The interface must be enabled for IPsec VPN.
- The peer must be configured.
- The IKE group specified in the peer configuration must be defined.
- The tunnel must be configured.
- The ESP group specified in the tunnel must be defined.
- The local IP address specified for the peer must be configured on the VPN-enabled interface.
- The **peer-address** type, **local-address** type, **tunnel local prefix** network type, and **tunnel remote prefix** network type, must all match. They must all be IPv4 or all be IPv6.

In addition, note that modifying global parameters (such as **auto-update** or **nat-traversal**) requires an IPsec restart, and therefore restarts all tunnels.

Adding, modifying, or deleting a tunnel restarts only the modified tunnel. Modifying an existing IKE group or ESP group restarts any tunnel using the group. Changing authentication information (pre-shared key or RSA signature) does not result in a tunnel restart.

Supported standards for IPsec VPN

The Brocade vRouter implementation of IPsec complies with the following standards:

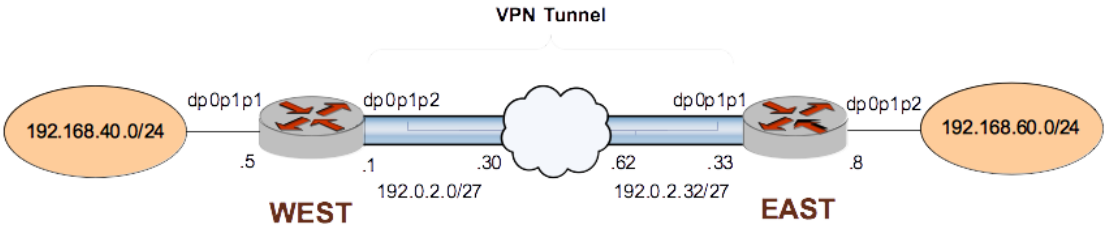
- RFC 2401, Security Architecture for the Internet Protocol
- RFC 2402, IP Authentication Header (AH)
- RFC 2406, IP Encapsulating Security Payload (ESP)
- RFC 2407, The Internet IP Security Domain of Interpretation for ISAKMP
- RFC 2408, Internet Security Association and Key Management Protocol (ISAKMP)
- RFC 2409, The Internet Key Exchange (IKE)
- RFC 2412, The OAKLEY Key Determination Protocol

IPsec Site-to-Site VPN Configuration

Basic site-to-site connection

This section presents a sample configuration for a basic IPsec tunnel between WEST and EAST Brocade vRouters on an IPv4 network. First WEST is configured, and then EAST. When you have finished, these peers will be configured as shown in the following section.

FIGURE 1 Basic site-to-site IPsec VPN connection



Before you begin:

- In this set of examples, we assume that you have two Brocade vRouters, with host names configured WEST and EAST. (The example systems are configured with the host name in upper case.)
- Any data plane interface used for IPsec VPN must already be configured. In this example, you need dp0p1p2 on WEST and dp0p1p1 on EAST, plus internal subnet information.
- The interface must be configured with the IP address you want to use as the source IP for packets sent to the peer VPN gateway. In this example, IP address 192.0.2.1 is defined on dp0p1p2 of WEST, and 192.0.2.33 is defined on dp0p1p1 of EAST. In examples where the interface is configured as a DHCP client, the interface address is set to **dhcp**.

NOTE
The sending and receiving of ICMP redirects is disabled when IPsec VPN is configured.

NOTE
In the Brocade vRouter, a data plane interface is an abstraction that represents the underlying physical or virtual Ethernet interface of the system.The terms Ethernet interface and data plane interface are synonymous in this guide.

Configure WEST

This section presents the following topics:

This section presents the following examples:

Configure an IKE group on WEST

The IKE group allows you to pre-define a set of one or more proposals to be used in IKE Phase 1 negotiation, after which the ISAKMP security association (SA) can be set up. For each proposal in the group, the following information is defined:

- Cipher to encrypt packets during IKE Phase 1
- Hash function to authenticate packets during IKE Phase 1

The IKE group also has a configured lifetime, which is the duration of the ISAKMP SA. When the lifetime of the ISAKMP SA expires, a new Phase 1 negotiation takes place, and new encryption, hash, and keying information is established in a new pair of ISAKMP SAs.

The lifetime is an attribute of the IKE group as a whole. If the IKE group contains multiple proposals, the lifetime applies regardless of which proposal in the group is accepted.

Table 4 creates IKE group IKE-1W on WEST. This IKE group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm
- Proposal 2 uses AES-128 as the encryption cipher and SHA-1 as the hash algorithm

The lifetime of a proposal from this IKE group is set to 3600 seconds.

To create this IKE group, perform the following steps on WEST in configuration mode.

TABLE 4 Configuring an IKE group on WEST

Step	Command
Create the configuration node for proposal 1 of IKE group IKE-1W.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W proposal 1 hash sha1`

TABLE 4 Configuring an IKE group on WEST (Continued)

Step	Command
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of IKE group IKE-1W.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W proposal 2 encryption aes128`
Set the hash algorithm for proposal 2.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W proposal 2 hash sha1`
Set the lifetime for the whole IKE group.	`vyatta@WEST# set security vpn ipsec ike-group IKE-1W lifetime 3600`

View the configuration for the IKE group. Don't commit yet.	`vyatta@WEST# show security vpn ipsec ike-group IKE-1W`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  proposal 2 {`
	`>  encryption aes128`
	`>  hash sha1`
	`>  }`
	`>  lifetime 3600`

Configure an ESP group on WEST

Encapsulated Security Payload (ESP) is an authentication protocol that provides authentication for IP packets, and it also encrypts them.

The ESP protocol negotiates a unique number for the session connection, called the Security Parameter Index (SPI). It also starts a numbering sequence for the packets and negotiates the hashing algorithm that authenticates packets.

The Brocade vRouter allows you to pre-define multiple ESP configurations. Each configuration is known as an ESP group. An ESP group includes the Phase 2 proposals, which contain the parameters that are needed to negotiate an IPsec security association:

- Cipher to encrypt user data across the IPsec tunnel
- Hashing function to authenticate packets in the IPsec tunnel
- Lifetime of the IPsec security association

Table 5 creates ESP group ESP-1W on Brocade vRouter WEST. This ESP group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm
- Proposal 2 uses Triple-DES as the encryption cipher and MD5 as the hash algorithm

The lifetime of a proposal from this ESP group is set to 1800 seconds.

To create this ESP group, perform the following steps on WEST in configuration mode.

TABLE 5 Configuring an ESP group on Brocade vRouter WEST

Step	Command
Create the configuration node for proposal 1 of ESP group ESP-1W.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W proposal 1`

TABLE 5 Configuring an ESP group on Brocade vRouter WEST (Continued)

Step	Command
Set the encryption cipher for proposal 1.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W proposal 1 hash sha1`
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of ESP group ESP-1W.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W proposal 2 encryption 3des`
Set the hash algorithm for proposal 2.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W proposal 2 hash md5`
Set the lifetime for the whole ESP group.	`vyatta@WEST# set security vpn ipsec esp-group ESP-1W lifetime 1800`

View the configuration for the ESP group. Don't commit yet.	`vyatta@WEST# show security vpn ipsec esp-group ESP-1W`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  proposal 2 {`
	`>  encryption 3des`
	`>  hash md5`
	`>  }`
	`>  lifetime 1800`

Create the connection to EAST

In defining a site-to-site connection, you specify IPsec policy information (most of which is pre-configured as an IKE and ESP group) and the routing information for the two endpoints of the IPsec tunnel.

The local endpoint is the Brocade vRouter. The remote endpoint is the peer VPN gateway—this gateway can be another Brocade vRouter, or it can be another IPsec-compliant router, an IPsec-capable firewall, or a VPN concentrator. For each end of the tunnel, you define the IP address and subnet mask of the local and remote subnets or hosts.

In all, you must specify the following:

- IP address of the remote peer.
- Authentication mode that the peers use to authenticate one another. The Brocade vRouter supports peer authentication by pre-shared secret (pre-shared key, or PSK), so you must also supply the character string to use to generate the hashed key. Digital signatures and X.509 certificates are also supported.
- IKE group to use in the connection.
- ESP group to use in the connection.
- IP address on this Brocade vRouter to use for the tunnel. This IP address must be pre-configured on the interface that is enabled for VPN.
- Communicating subnet or host for each end of the tunnel. You can define multiple tunnels for each VPN peer, and each tunnel can use a different security policy.

When supplying a preshared secret, keep the following in mind:

A preshared secret, or pre-shared key (PSK), is a method of authentication. The secret, or key, is a character string agreed upon beforehand by both parties as the key for authenticating the session. It generates a hash such that each VPN endpoint can authenticate the other.

Note that the pre-shared secret, although an ordinary character string, is not a “password.” It actually generates a hashed key to form a fingerprint that proves the identity of each endpoint. This means that long, complex character strings are more secure than short strings. Choose complex pre-shared secrets and avoid short ones, which can be more easily compromised by an attack.

The preshared secret is not passed during IKE negotiation. It is configured on both sides, and must match on both sides.

A preshared secret is an example of symmetric cryptography: the key is the same on both sides. Symmetric encryption algorithms are less computationally intensive than asymmetric algorithms, and are, therefore, faster. However, in symmetric cryptography, the two communicating parties must exchange keys in advance. Doing this securely can be a problem.

A preshared secret and a digital signature are the most common methods of IKE authentication. A preshared secret is an easy and effective way to quickly set up authentication with little administrative overhead. However, it has several drawbacks.

- If a preshared key is captured and no one is aware of it, the attacker has access to your network as long as that key is in use.
- A preshared secret is manually configured, so it should be regularly changed. However, this task often falls off the list of busy network administrators. Using preshared key values with remote users is equivalent to giving them a password to your network.

NOTE
You should restrict the use of pre-shared keys to smaller, low-risk environments.

The following example defines a site-to-site connection to EAST.

- This connection is configured with a single tunnel:
  - Tunnel 1 communicates between 192.168.40.0/24 on WEST and 192.168.60.0/24 on EAST, using ESP group ESP-1W.
- WEST uses IP address 192.0.2.1 on dp0p1p2.
- EAST uses IP address 192.0.2.33 on dp0p1p1.
- The IKE group is IKE-1W.
- The authentication mode is pre-shared secret. The pre-shared secret is “test_key_1”.

To configure this connection, perform the following steps on Brocade vRouter WEST in configuration mode.

TABLE 6 Creating a site-to-site connection from WEST to EAST

Step	Command
Create the node for EAST and set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 192.0.2.33` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Provide the string that will be used to generate encryption keys.	`vyatta@WEST# set authentication pre-shared-secret test_key_1` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`

TABLE 6 Creating a site-to-site connection from WEST to EAST (Continued)

Step	Command
Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Create a tunnel configuration, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 1 local prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Provide the remote subnet for the tunnel.	`vyatta@WEST# set tunnel 1 remote prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Return to the top of the configuration tree.	`vyatta@WEST# top`
Now commit the configuration.	`vyatta@WEST# commit`

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33`
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`

View data plane interface dp0p1p2 address configuration. **local-address** is set to this address.	`vyatta@WEST# show interfaces dataplane dp0p1p2 address` `address 192.0.2.1/27`

Configure EAST

This section presents the following examples:

- Configure an IKE group on EAST on page 27
- Configure an ESP group on EAST on page 27
- Create the connection to WEST on page 28

Configure an IKE group on EAST

Table 7 creates IKE group IKE-1E on EAST. This IKE group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm
- Proposal 2 uses AES-128 as the encryption cipher and SHA-1 as the hash algorithm

The lifetime of a proposal from this IKE group is set to 3600.

Note that these parameters correspond to those set in IKE-1W on WEST. You must ensure, in defining proposals, that the encryption ciphers and hash algorithms are such that the two peers will be able to agree on at least one combination.

To create this IKE group, perform the following steps on EAST in configuration mode.

TABLE 7 Configuring an IKE group on EAST

Step	Command
Create the configuration node for proposal 1 of IKE group IKE-1E.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E proposal 1 hash sha1`
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of IKE group IKE-1E.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E proposal 2 encryption aes128`
Set the hash algorithm for proposal 2.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E proposal 2 hash sha1`
Set the lifetime for the whole IKE group.	`vyatta@EAST# set security vpn ipsec ike-group IKE-1E lifetime 3600`

View the configuration for the IKE group. Don't commit yet.	`vyatta@EAST# show security vpn ipsec ike-group IKE-1E`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  proposal 2 {`
	`>  encryption aes128`
	`>  hash sha1`
	`>  }`
	`>  lifetime 3600`

Configure an ESP group on EAST

Table 8 creates ESP group ESP-1E on EAST. This ESP group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm
- Proposal 2 uses Triple-DES as the encryption cipher and MD5 as the hash algorithm

The lifetime of a proposal from this ESP group is set to 1800 seconds.

To create this ESP group, perform the following steps on EAST in configuration mode.

TABLE 8 Configuring an ESP group on EAST

Step	Command
Create the configuration node for proposal 1 of ESP group ESP-1E.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E proposal 1 hash sha1`
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of ESP group ESP-1E.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E proposal 2 encryption 3des`
Set the hash algorithm for proposal 2.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E proposal 2 hash md5`
Set the lifetime for the whole ESP group.	`vyatta@EAST# set security vpn ipsec esp-group ESP-1E lifetime 1800`

View the configuration for the ESP group. Don't commit yet.	`vyatta@EAST# show security vpn ipsec esp-group ESP-1E`
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  proposal 2 {`
	`>  encryption 3des`
	`>  hash md5`
	`>  }`
	`>  lifetime 1800`

Create the connection to WEST

Table 9 defines a site-to-site connection to WEST. In this example:

- This connection is configured with a single tunnel:
  - Tunnel 1 communicates between 192.168.60.0/24 on EAST and 192.168.40.0/24 on WEST, using ESP group ESP-1E.
- EAST uses IP address 192.0.2.33 on dp0p1p1.
- WEST uses IP address 192.0.2.1 on dp0p1p2.
- The IKE group is IKE-1E.
- The authentication mode is pre-shared secret. The pre-shared secret is “test_key_1.”

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 9 Creating a site-to-site connection from EAST to WEST

Step	Command
Create the node for WEST and set the authentication mode.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@EAST# edit security vpn ipsec site-to-site peer 192.0.2.1` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`

TABLE 9 Creating a site-to-site connection from EAST to WEST (Continued)

Step	Command

Provide the string that will be used to generate encryption keys.	`vyatta@EAST# set authentication pre-shared-secret test_key_1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify the default ESP group for all tunnels.	`vyatta@EAST# set default-esp-group ESP-1E`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify the IKE group.	`vyatta@EAST# set ike-group IKE-1E`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@EAST# set local-address 192.0.2.33`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Create a tunnel configuration, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 1 local prefix 192.168.60.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Provide the remote subnet for the tunnel.	`vyatta@EAST# set tunnel 1 remote prefix 192.168.40.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Return to the top of the configuration tree.	`vyatta@EAST# top`
Now commit the configuration.	`vyatta@EAST# commit`

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`  local {`
	`  prefix 192.168.60.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.40.0/24`
	`  }`
	`}`

View data plane interface dp0p1p1 address configuration. **local-address** is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address`
	
	`address 192.0.2.33/27`

RSA digital signature authentication

This section presents the following topics:

In this set of examples, you modify the VPN connection configured in the previous set of examples between WEST and EAST (Basic site-to-site connection on page 21). The site-to-site connection created in that set of examples used pre-shared keys for authentication. This set of examples modifies the connection to use RSA digital signatures for authentication.

Generate a digital signature on WEST

In this example, you generate WEST's digital signature. This signature will have two parts: a public part (the public key) and a private part (the private key). The public key will be shared with EAST; the private key will remain secret.

To generate an RSA digital signature for system WEST, perform the following steps in operational mode.

TABLE 10 Generating a digital signature on WEST

Step	Command
Generate the key.	`vyatta@WEST> generate vpn rsa-key`
The system warns you that the existing RSA key file will be overwritten. You have the opportunity to exit the key generation process by pressing <Ctrl>+c.	`A local RSA key file already exists and will be overwritten` `<CTRL>C to exit: 8`
The system indicates the location of the file where the key will be written.	`Generating rsa-key to /config/ipsec.d/rsa-keys/localhost.key`

TABLE 10 Generating a digital signature on WEST (Continued)

Step	Command
The system displays the public portion of the generated key. By default, this key (including the private portion of the key) is stored in /config/ipsec.d/rsa-keys/localhost.key	Your new local RSA key has been generated The public portion of the key is: 0sAQPEOQvukvkv1ofuO8gEKp7IFFZz4lQqMZyVMInoQKUU/T0iKSK/ 0NSH9Ldrr8yQUFayzKag6wM7ASXWXKyt0LS1Gn8tJVsjKGaOkFgLREtVJD3pRzoc7DSUOBViCD6f/ TloTkPepRUtW1bmYev2H7tajSO0K0 +7nlocZI0ppMAyF6CS+Wd5W1JBpVGL+EkKfyEl9RagKxRW82XJbgY4LG77K2YDN90Wd2GgMY3kf+YJLIzFEt/x 5PedUutJCK5RMwl +IJGaxrKf1OmCQfzXlkM09ijZx8kzPIlBk 5hulZrbUWjzBJdFcwFAyPM3yCuv3+ndFX00t3ZLfKu+/wX595J vyatta@WEST>

Generate a digital signature on EAST

In this example, you generate EAST's digital signature. This signature will have two parts: a public part (the public key) and a private part (the private key). The public key will be shared with WEST; the private key will remain secret.

To generate an RSA digital signature for system EAST, perform the following steps in operational mode.

TABLE 11 Generating a digital signature on EAST

Step	Command
Generate the key.	vyatta@EAST> generate vpn rsa-key
The system warns you that the existing RSA key file will be overwritten. You have the opportunity to exit the key generation process by pressing <Ctrl>+c.	A local RSA key file already exists and will be overwritten <CTRL>C to exit: 5
The system indicates the location of the file where the key will be written.	Generating rsa-key to /config/ipsec.d/rsa-keys/localhost.key

TABLE 11 Generating a digital signature on EAST (Continued)

Step	Command
The system displays the public portion of the generated key. By default, this key (including the private portion of the key) is stored in /config/ipsec.d/rsa-keys/localhost.key	Your new local RSA key has been generated The public portion of the key is: 0sAQOVBIJL+rIkpTuwh8FPeceAF0bhgLr+ +W51bOAIjFbRDbR8gX3Vlz6wiUbMgGwQxWlYQiqsCeacicsfZx/ amlEn9PkSE4e7tqK/JQo40L5C7gcNM24mup1d +0WmN3zLb9Qhmq5q3pNJxEwnVbPPQeIdZMJxnb1+lA8DPC3SIxJM/3at1/ KrwqCAhX3QNFY/zNmOtFogELCeyl4+d54wQljA+3dwFAQ4bboJ7YIDs +rqORxWd3l3I7IajT/pLrwr5eZ8OA9NtAedbMiCwxyuyUbznxXZ8Z/ MAi3xjL1pjYyWjNNiOij82QJfMOrjoXVCfcPn96ZN+Jqk +KknoVeNDwzpoahFOseJREeXzkw3/lkMN9N1 vyatta@EAST>

Record EAST's public key on WEST

In this example, you record the public key you have obtained from EAST. The key is then saved under a name that you can refer to in site-to-site configuration.

A digital signature can be typed in manually, but digital signatures are lengthy and difficult to type. It is generally easier to copy the digital signature into the clipboard of your system and then paste it into the configuration. You do this in a number of ways; for example:

- Receive the public key from the operator of the VPN peer in an e-mail—perhaps an e-mail protected by a PGP signature. Copy the key text into your clipboard.
- From an X.509 certificate, provided by a Certificate Agency.
- Connect to the VPN peer directly through a Telnet or SSH control session. View the public portion of the key using a **show** command, select the text, and copy the key text into your clipboard.

Table 12 pastes EAST's public key into RSA configuration. The name “EAST-key” is used as the identifier of the key.

Before you begin, copy EAST's public key into your clipboard.

If you are in operational mode on WEST, enter configuration mode now and perform the following steps:

TABLE 12 Record EAST's public key on WEST

Step	Command
Specify a name for EAST's public key and paste EAST's public key into the configuration.	vyatta@WEST# set security vpn rsa-keys rsa-key-name EAST-key rsa-key 0sAQOVBIJL+rIkpTuwh8FPeceAF0bhgLr+ +W51bOAIjFbRDbR8gX3Vlz6wiUbMgGwQxWlYQiqsCeacicsfZx/ amlEn9PkSE4e7tqK/JQo40L5C7gcNM24mup1d +0WmN3zLb9Qhmq5q3pNJxEwnVbPPQeIdZMJxnb1+lA8DPC3SIxJM/3at1/ KrwqCAhX3QNFY/zNmOtFogELCeyl4+d54wQljA+3dwFAQ4bboJ7YIDs +rqORxWd3l3I7IajT/pLrwr5eZ8OA9NtAedbMiCwxyuyUbznxXZ8Z/ MAi3xjL1pjYyWjNNiOij82QJfMOrjoXVCfcPn96ZN+Jqk +KknoVeNDwzpoahFOseJREeXzkw3/lkMN9N1
Commit the configuration.	vyatta@WEST# commit

TABLE 12 Record EAST's public key on WEST (Continued)

Step	Command

View the configuration for RSA keys.
 	
Since you have not changed the configuration for the local host's key, it does not display.
	vyatta@WEST# show security vpn rsa-keys
	
	rsa-key-name EAST-key {
	
	rsa-key
	0sAQOVBIJL+rIkpTuwh8FPeceAF0bhgLr+
	+W51bOAIjFbRDbR8gX3Vlz6wiUbMgGwQxWlYQiqsCeacicsfZx/
	amlEn9PkSE4e7tqK/JQo40L5C7gcNM24mup1d
	+0WmN3zLb9Qhmq5q3pNJxEwnVbPPQeIdZMJxnb1+lA8DPC3SIxJM/3at1/
	KrwqCAhX3QNFY/zNmOtFogELCeyl4+d54wQljA+3dwFAQ4bboJ7YIDs
	+rqORxWd3l3I7IajT/pLrwr5eZ8OA9NtAedbMiCwxyuyUbznxXZ8Z/
	MAi3xjL1pjYyWjNNiOij82QJfMOrjoXVCfcPn96ZN+Jqk
	+KknoVeNDwzpoahFOseJREeXzkw3/lkMN9N1
	}
	
	vyatta@WEST#

Modify WEST's connection to EAST

Table 13 modifies the connection from WEST to EAST to use RSA digital signatures for authentication. In this example:

- The authentication mode is changed from pre-shared secret to RSA digital signatures.
- EAST's public key is specified as the remote key, under the identifier configured in the previous step (see Record EAST's public key on WEST on page 32).

To modify the site-to-site connection to use RSA configuration, perform the following steps:

TABLE 13 Configure WEST for RSA authentication

Step	Command
Remove the pre-shared key.	vyatta@WEST# delete security vpn ipsec site-to-site peer 192.0.2.33 authentication pre-shared-secret
Change the authentication mode.	vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication mode rsa
Provide the identifier for EAST's digital signature.	vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication rsa-key-name EAST-key
Commit the configuration.	vyatta@WEST# commit

View the modified configuration for the site-to-site connection.	vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33
	
	authentication {
	mode rsa
	rsa-key-name EAST-key
	}
	default-esp-group ESP-1W
	ike-group IKE-1W
	local-address 192.0.2.1
	tunnel 1 {
	local {
	prefix 192.168.40.0/24
	}
	remote {
	prefix 192.168.60.0/24
	}
	}

TABLE 13 Configure WEST for RSA authentication (Continued)

Step	Command
View data plane interface dp0p1p2 address configuration. local-address is set to this address.	`vyatta@WEST# show interfaces dataplane dp0p1p2 address address 192.0.2.1/27`

Record WEST's public key on EAST

Table 14 pastes WEST's public key into RSA configuration. The name “WEST-key” is used as the identifier of the key.

Before you begin, copy WEST's public key into your clipboard.

If you are in operational mode on EAST, enter configuration mode now and perform the following steps:

TABLE 14 Record WEST's public key on EAST

Step	Command
Specify a name for WEST's public key and paste WEST's public key into the configuration.	`vyatta@EAST# set security vpn rsa-keys rsa-key-name WEST-key rsa-key 0sAQPEOQvukvkvlofuO8gEKp7IFFZz4lQqMZyVMInoQKUU/T0iKSK/ 0NSH9Ldrr8yQUFayzKag6wM7ASXWXKyt0LS1Gn8tJVsjKGaOkFgLREtVJD3pRzoc7DSUOBViCD6f/ TloTkPepRUtW1bmYev2H7tajSO0K0 +7nlocZI0ppMAyF6CS+Wd5W1JBpVGL+EkKfyEl9RagKxRW82XJbgY4LG77K2YDN90Wd2GgMY3kf+YJLIzFEt/ 5PedUutJCK5RMwl +IJGaxrKf1OmCQfzXlkM09ijZx8kzPIlBk 5hulZrbUWjzBJdFcwFAyPM3yCuv3+ndFX00t3ZLfKu+/wX595J`
Commit the configuration.	`vyatta@EAST# commit`
View the configuration for RSA keys. Since you have not changed the configuration for the local host's key, it does not display.	`vyatta@EAST# show security vpn rsa-keys` `rsa-key-name WEST-key {` `rsa-key 0sAQPEOQvukvkvlofuO8gEKp7IFFZz4lQqMZyVMInoQKUU/T0iKSK/ 0NSH9Ldrr8yQUFayzKag6wM7ASXWXKyt0LS1Gn8tJVsjKGaOkFgLREtVJD3pRzoc7DSUOBViCD6f/ TloTkPepRUtW1bmYev2H7tajSO0K0 +7nlocZI0ppMAyF6CS+Wd5W1JBpVGL+EkKfyEl9RagKxRW82XJbgY4LG77K2YDN90Wd2GgMY3kf+YJLIzFEt/ 5PedUutJCK5RMwl +IJGaxrKf1OmCQfzXlkM09ijZx8kzPIlBk 5hulZrbUWjzBJdFcwFAyPM3yCuv3+ndFX00t3ZLfKu+/wX595J` `}`  `vyatta@EAST#`

Modify EAST's connection to WEST

Table 15 modifies the connection from EAST to WEST to use RSA digital signatures for authentication.

In this example:

- The authentication mode is changed from pre-shared secret to RSA digital signatures.
- WEST's public key is specified as the remote key, under the identifier configured in the previous step (see Record WEST's public key on EAST on page 34).

To modify the site-to-site connection to use RSA configuration, perform the following steps:

TABLE 15 Configure EAST for RSA authentication

Step	Command
Remove the pre-shared key.	`vyatta@EAST# delete security vpn ipsec site-to-site peer 192.0.2.1 authentication pre-shared-secret`
Change the authentication mode.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication mode rsa`
Provide the identifier for WEST's digital signature.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication rsa-key-name WEST-key`
Commit the configuration.	`vyatta@EAST# commit`

View the modified configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	`authentication {`
	`  mode rsa`
	`  rsa-key WEST-key`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`  local {`
	`  prefix 192.168.60.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.40.0/24`
	`  }`
	`}`

View data plane interface dp0p1p1 address configuration. local-address is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address` `address 192.0.2.33/27`

X.509 certificate authentication

In this set of examples, you modify the VPN connection configured in the basic set of examples between WEST and EAST (Basic site-to-site connection on page 21). The site-to-site connection created in that set of examples used pre-shared keys for authentication. This set of examples modifies the configuration to use X.509 certificates for authentication.

In general, the procedure for obtaining the files required to authenticate using X.509 certificates is as follows:

1. Generate the private key and a certificate signing request (CSR) (based on the public key). This can be accomplished using the **generate vpn x509 key-pair <name>** command (for example, **generate vpn x509 key-pair west**, where west.key is the private key and *west.csr* is the certificate signing request file—both created in /config/auth).
2. Send the CSR file (for example, *west.csr*) to the certificate authority (CA) and receive back a server certificate (for example, *west.crt*), the CA certificate (for example, *ca.crt*), and potentially, a certificate revocation list (CRL) file. This procedure varies according to the CA being used.

At this point, the configuration can be modified to use these files.

Modify WEST's connection to EAST

Table 16 modifies the connection from WEST to EAST to use X.509 certificates for authentication. In this example:

- The authentication mode is changed from pre-shared secret to X.509 certificates.
- The certificate for the peer is identified using its Distinguished Name information. This is the information prompted for when creating the certificate signing request (CSR) file on the peer.
- The locations of the CA certificate, the server certificate, and the private key file for the server are specified.

To modify the site-to-site connection to use X.509 certificate authentication, perform the following steps:

TABLE 16 Configure WEST for x.509 certificate authentication

Step	Command
Remove the pre-shared key.	`vyatta@WEST# delete security vpn ipsec site-to-site peer 192.0.2.33 authentication pre-shared-secret`
Change the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication mode x509`
Specify the 'distinguished name' of the certificate for the peer.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication remote-id "C=US, ST=CA, O=ABC Company, CN=east, E=root@abcco.com"`
Specify the location of the CA certificate.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication x509 ca-cert-file /config/auth/ca.crt`
Specify the location of the server certificate.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication x509 cert-file /config/auth/west.crt`
Specify the location of the server key file.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication x509 key file /config/auth/west.key`
Specify the password for the server key file.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication x509 key password testpwd-west`
Commit the configuration.	`vyatta@WEST# commit`

View the modified configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33`
	`authentication {`
	`mode x509`
	`remote-id "C=US, ST=CA, O=ABC Company, CN=east,`
	`E=root@abcco.com"`
	`x509 {`
	`ca-cert-file /config/auth/ca.crt`
	`cert-file /config/auth/west.crt`
	`key {`
	`file /config/auth/west.key`
	`password testpwd-west`
	`}`
	`}`
	`}`

TABLE 16 Configure WEST for x.509 certificate authentication (Continued)

Step	Command

	default-esp-group ESP-1W
	ike-group IKE-1W
	local-address 192.0.2.1
	tunnel 1 {
	local {
	prefix 192.168.40.0/24
	}
	remote {
	prefix 192.168.60.0/24
	}
	}

View data plane interface dp0p1p2 address configuration. **local-address** is set to this address.	vyatta@WEST# show interfaces dataplane dp0p1p2 address address 192.0.2.1/27

Modify EAST's connection to WEST

Table 17 modifies the connection from EAST to WEST to use X.509 certificates for authentication.

In this example:

- The authentication mode is changed from pre-shared secret to X.509 certificates.
- The certificate for the peer is identified using its 'distinguished name' information. This is the information prompted for when creating the certificate signing request (CSR) file.
- The locations of the CA certificate, the server certificate, and the private key file for the server are specified.

To modify the site-to-site connection to use X.509 certificate authentication, perform the following steps:

TABLE 17 Configure EAST for x.509 certificate authentication

Step	Command
Remove the pre-shared key.	vyatta@EAST# delete security vpn ipsec site-to-site peer 192.0.2.1 authentication pre-shared-secret
Change the authentication mode.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication mode x509
Specify the 'distinguished name' of the certificate for the peer.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication remote-id "C=US, ST=CA, O=ABC Company, CN=west, E=root@abcco.com"
Specify the location of the CA certificate.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication x509 ca-cert-file /config/auth/ca.crt
Specify the location of the server certificate.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication x509 cert-file /config/auth/east.crt
Specify the location of the server key file.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication x509 key file /config/auth/east.key
Specify the password for the server key file.	vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication x509 key password testpwd-east
Commit the configuration.	vyatta@EAST# commit

TABLE 17 Configure EAST for x.509 certificate authentication (Continued)

Step	Command

View the modified configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	`authentication {`
	`mode x509`
	`remote-id "C=US, ST=CA, O=ABC Company, CN=west,`
	`E=root@abcco.com"`
	`x509 {`
	`ca-cert-file /config/auth/ca.crt`
	`cert-file /config/auth/east.crt`
	`key {`
	`file /config/auth/east.key`
	`password testpwd-east`
	`}`
	`}`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.60.0/24`
	`}`
	`remote {`
	`prefix 192.168.40.0/24`
	`}`
	`}`

View data plane interface dp0p1p1 address configuration. local-address is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address address 192.0.2.33/27`

VPN connection to a peer with a dynamic IP address

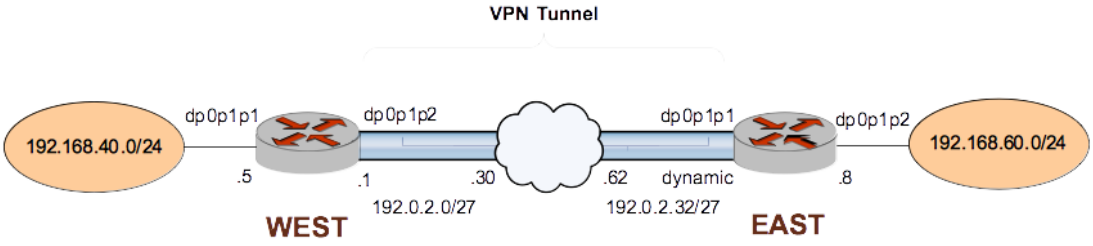
This section presents the following topics:

This section presents a sample configuration for a connection between WEST and EAST, where EAST has a dynamic IP address (it is configured as a DHCP client). In this example:

- EAST has a dynamic IP address from WEST's point of view.
- WEST retains its fixed IP address.

When you have finished, these systems will be configured as shown in Figure 2 .

FIGURE 2 IPsec VPN connection with dynamic IP address



Before you begin:

- This example assumes that you have already configured a basic site-to-site connection using a preshared key between WEST and EAST, as explained in the section Basic site-to-site connection on page 21. Only the relevant changes to that configuration are presented here.

Configure WEST

Table 18 defines configuration changes for a new site-to-site connection to EAST. The main change is the IP address specification of the peer. This is set to 0.0.0.0 to represent “any” IP address. Because the IP address of the peer is unknown, WEST will not initiate connections to the peer. It will only receive connections from the peer.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 18 Creating a site-to-site connection to a peer with a dynamic IP address

Step	Command
Delete the previous configuration.	`vyatta@WEST# delete security vpn ipsec site-to-site peer 192.0.2.33`
Create the node for EAST and set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 0.0.0.0 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 0.0.0.0` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Provide the string that will be used to generate encryption keys.	`vyatta@WEST# set authentication pre-shared-secret test_key_1` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Create a tunnel configuration, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 1 local prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Provide the remote subnet for the tunnel.	`vyatta@WEST# set tunnel 1 remote prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 0.0.0.0]`
Return to the top of the configuration tree.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

TABLE 18 Creating a site-to-site connection to a peer with a dynamic IP address (Continued)

Step	Command

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 0.0.0.0`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`

View data plane interface dp0p1p2 address configuration. local-address is set to this address.	`vyatta@WEST# show interfaces dataplane dp0p1p2 address`
	
	`address 192.0.2.1/27`

Configure EAST

The connection from EAST to WEST only requires a minor change from that configured in the section Basic site-to-site connection on page 21.

- WEST retains its fixed IP, so no modification is required to the remote peer IP address.
- EAST has a dynamic local IP, so that must change. The *dhcp-interface* option specifies the DHCP client interface.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 19 Specify that the local IP is dynamic

Step	Command

Remove the existing **local-address** configuration so that doesn't conflict with the **dhcp-interface** configuration that will be set.	`vyatta@EAST# delete security vpn ipsec site-to-site peer 192.0.2.1 local-address`
	
	`[edit]`

Specify the DHCP client interface to use for the connection.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 dhcp-interface dp0p1p1`
	
	`[edit]`

Commit the configuration.	`vyatta@EAST# commit`

TABLE 19 Specify that the local IP is dynamic (Continued)

Step	Command

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1E`
	`dhcp-interface dp0p1p1`
	`ike-group IKE-1E`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.60.0/24`
	`}`
	`remote {`
	`prefix 192.168.40.0/24`
	`}`
	`}`

View data plane interface dp0p1p1 address configuration. It is set to dhcp which configures it as a DHCP client. This is the setting required by dhcp-interface.	`vyatta@EAST# show interfaces dataplane dp0p1p1`
	
	`address dhcp`

VPN connection to a peer using dynamic DNS

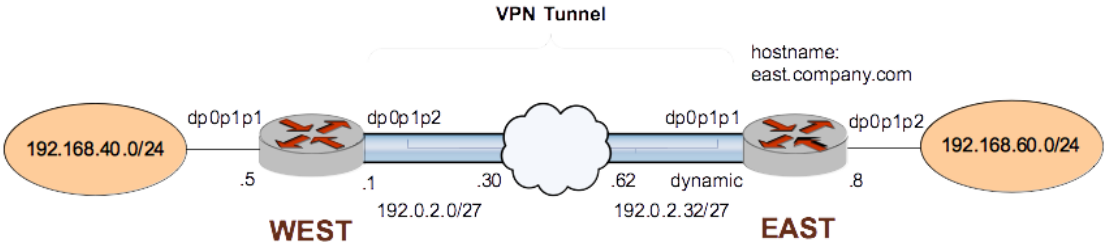
This section presents the following topics:

This section presents a sample configuration for a connection between WEST and EAST, where EAST has a dynamic IP address (it is configured as a DHCP client) and is configured for dynamic DNS. In this example:

- EAST has a dynamic IP address from WEST's point of view but WEST can initiate connections to EAST because EAST's hostname remains constant even though its IP address may change.
- WEST retains its fixed IP address.

When you have finished, these systems will be configured as shown in the following figure.

FIGURE 3 IPsec VPN connection with dynamic IP address and dynamic DNS



Before you begin:

- This example assumes that you have already configured a basic site-to-site connection using a preshared key between WEST and EAST, as explained in the section Basic site-to-site connection on page 21. Only the relevant changes to that configuration are presented here.

Configure WEST

Table 20 defines configuration changes for a new site-to-site connection to EAST.

- The main change is the IP address specification of the peer. This is set to the hostname for EAST: “east.company.com”. This is the hostname that is configured on EAST with the dynamic DNS provider. Because the IP address for EAST can be resolved, WEST can either initiate IPsec connections to, or receive IPsec connections from EAST.
- The other important change is to configure **auto-update** so that if EAST's IP address changes, the IPsec connection to EAST will be restarted automatically.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 20 Creating a site-to-site connection to a peer with a dynamic IP address and using dynamic DNS

Step	Command
Delete the previous configuration.	`vyatta@WEST# delete security vpn ipsec site-to-site peer 192.0.2.33`
Create the node for EAST and set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer east.company.com authentication mode pre-shared-secret`

Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer east.company.com`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Provide the string that will be used to generate encryption keys.	`vyatta@WEST# set authentication pre-shared-secret test_key_1`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Create a tunnel configuration, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 1 local prefix 192.168.40.0/24`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

TABLE 20 Creating a site-to-site connection to a peer with a dynamic IP address and using dynamic DNS (Continued)

Step	Command

Provide the remote subnet for the tunnel.	`vyatta@WEST# set tunnel 1 remote prefix 192.168.60.0/24`
	
	`[edit security vpn ipsec site-to-site peer east.company.com]`

Return to the top of the configuration tree.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer east.company.com`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`

View data plane interface dp0p1p2 address configuration. local-address is set to this address.	`vyatta@WEST# show interfaces dataplane dp0p1p2 address`
	
	`address 192.0.2.1/27`

Specify that the IPsec connection should be refreshed every 60 seconds - in case the peer's IP address changes. If this happens, the new IP address will be resolved via the dynamic DNS service provider.	`vyatta@WEST# set security vpn ipsec auto-update 60`
	
	`[edit]`

Commit the configuration.	`vyatta@WEST# commit`

View the configuration.	`vyatta@WEST# show security vpn ipsec auto-update`
	
	`auto-update 60`

Configure EAST

The connection from EAST to WEST only requires a minor change from that configured in the section Basic site-to-site connection on page 21.

- WEST retains its fixed IP, so no modification is required to the remote peer IP address.
- EAST has a dynamic local IP, so that must change. The **dhcp-interface** option specifies the DHCP client interface.
- EAST is also configured for dynamic DNS, in this case with service provider DynDNS. See the “Configuring Dynamic DNS” section in the *Brocade 5600 vRouter Services Reference Guide* for details on configuring a system for dynamic DNS.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 21 Specify that the local IP is dynamic

Step	Command

Remove the existing local-address configuration so that doesn't conflict with the dhcp-interface configuration that will be set.	`vyatta@EAST# delete security vpn ipsec site-to-site peer 192.0.2.1 local-address`
	
	`[edit]`

Specify the DHCP client interface to use for the connection.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 dhcp-interface dp0p1p1`
	
	`[edit]`

Commit the configuration.	`vyatta@EAST# commit`

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1E`
	`dhcp-interface dp0p1p1`
	`ike-group IKE-1E`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.60.0/24`
	`}`
	`remote {`
	`prefix 192.168.40.0/24`
	`}`
	`}`

View data plane interface dp0p1p1 address configuration. It is set to dhcp which configures it as a DHCP client. This is the setting required by dhcp-interface.	`vyatta@EAST# show interfaces dataplane dp0p1p1`
	
	`address dhcp`

Display the dynamic DNS configuration on EAST:

TABLE 22 Display the dynamic DNS configuration

Step	Command

View the dynamic DNS configuration.	`vyatta@EAST# show service dns dynamic`
	
	`interface dp0p1p1 {`
	`service dyndns {`
	`host-name east.company.com`
	`login test`
	`password testpassword`
	`}`
	`}`

VPN connection with NAT

This section presents the following topics:

- Configure WEST on page 46
- Configure EAST on page 48

Native IPsec packets are encapsulated using Encapsulated Security Payload (ESP). In these packets, the IP addresses are embedded within the encapsulated packet. This causes problems when IPsec packets must traverse a NAT gateway.

When performing Network Address Translation (NAT), the NAT gateway substitutes its own source IP address (and sometimes a port number), for the original source IP and port on outgoing packets. The NAT device listens for a reply, and when a response packet is received, the NAT device reverses the translation so that the incoming packet can arrive at the correct destination. This allows IP addresses within a private network to be “hidden” from external networks.

NAT does not work well with IPsec, because the IP addresses are embedded within the payload of the encapsulated packet. For a number of reasons, this means that the IPsec peer cannot be located behind the NAT device.

The IPsec NAT Traversal protocol (NAT-T, RFCs 3947 and 3948) allows each IPsec packet to be re-encapsulated within a UDP packet, which can be handled correctly by the NAT device. NAT-T runs on top of IPsec. To support NAT-T, the firewall must be set to allow all of the following:

- IKE through UDP port 500
- IPsec NAT-T through UDP port 4500
- ESP

Some gateway devices pre-allow all of these in a feature called IPsec Passthrough. However, IPsec Passthrough is incompatible with NAT traversal. IPsec Passthrough devices recognize the IPsec-in-UDP packets and incorrectly attempt passthrough-type operations on the packets. This corrupts the packets in such a way that NAT-T no longer works.

NOTE
If you enable NAT traversal support, make sure you DISABLE IPsec Passthrough on the NAT device.

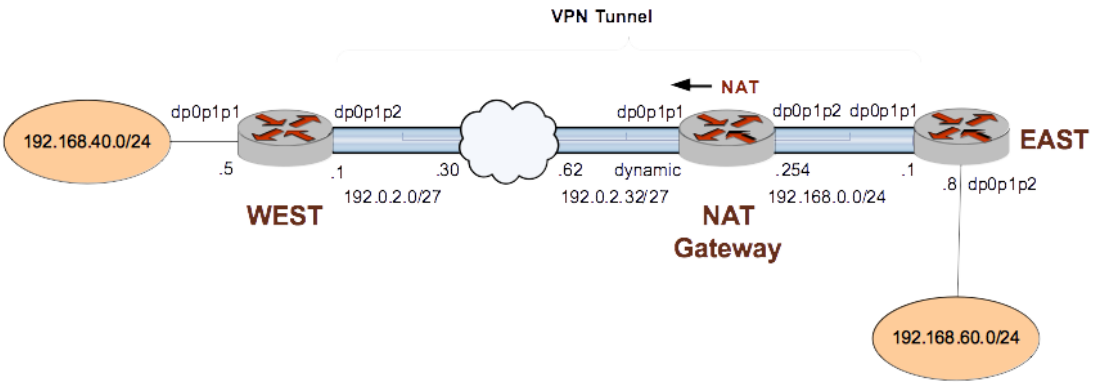
This section presents a sample configuration for a NATted connection between WEST and EAST. It is similar to the previous example except that in this case EAST resides behind a NAT device. In this example:

- EAST resides behind a NAT device, and has a dynamic IP address from WEST's point of view.
- WEST retains its fixed IP address.

This configuration is similar to something you might see for an IPsec endpoint that is behind a DSL connection, where the DSL peer's public IP address is dynamic and the DSL peer is performing NAT.

When you have finished, these systems will be configured as shown in the following figure.

FIGURE 4 IPsec VPN connection with dynamic IP address and NAT



Before you begin:

- This example assumes that you have already configured a basic site-to-site connection using a preshared key between WEST and EAST, as explained in the section Basic site-to-site connection on page 21. Only the relevant changes to that configuration are presented here.
- This example also assumes that a Masquerade NAT rule is configured on a Brocade vRouter called “NAT Gateway” in front of EAST as follows:

TABLE 23 NAT configuration on the NAT gateway

Step	Command

Show the configuration.	`vyatta@NATGwy# run show nat source rule`
	
	`outbound-interface dp0p1p1`
	`source {`
	`address 192.168.0.0/24`
	`}`
	`type masquerade`

Configure WEST

To allow for EAST's dynamic IP address via NAT, WEST must specify that the VPN will be traversing NAT, that addresses from certain private networks are allowed, that addresses from the same subnet as the local private subnet are not allowed, and that a new site-to-site connection is required to a peer that has a dynamic IP address.

Table 24 defines configuration changes for a new site-to-site connection to EAST via NAT.

- One important change is to add the NAT traversal related commands.
- Another important change is the IP address of the peer. This is set to 0.0.0.0 to represent “any” IP address. Because the IP address of the peer is unknown, WEST will not initiate connections to the peer. It will only receive connections from the peer.
- All other information is set to be the same as the connection created for the basic site-to-site tunnel.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 24 Creating a site-to-site connection to a peer with a dynamic IP address via NAT

Step	Command

Enable NAT traversal.	`vyatta@WEST# set security vpn ipsec nat-traversal enable`
	
	`[edit]`

Allow private network 10.0.0.0/8.	`vyatta@WEST# set security vpn ipsec nat-networks allowed-network 10.0.0.0/8`
	
	`[edit]`

Allow private network 172.16.0.0/12.	`vyatta@WEST# set security vpn ipsec nat-networks allowed-network 172.16.0.0/12`
	
	`[edit]`

Allow private network 192.168.0.0/16, but exclude the local subnet (192.168.40.0/24).	`vyatta@WEST# set security vpn ipsec nat-networks allowed-network 192.168.0.0/16 exclude 192.168.40.0/24`
	
	`[edit]`

Commit the configuration.	`vyatta@WEST# commit`

View the newly added configuration (only the relevant parts of the configuration are shown).	`vyatta@WEST# show security vpn ipsec`
	
	`(...)`
	`  nat-networks {`
	`  allowed-network 10.0.0.0/8 {`
	`  }`
	`  allowed-network 172.16.0.0/12 {`
	`  }`
	`  allowed-network 192.168.0.0/16 {`
	`  exclude 192.168.40.0/24`
	`  }`
	`  }`
	`  nat-traversal enable`
	`(...)`

Delete the previous configuration.	`vyatta@WEST# delete security vpn ipsec site-to-site peer 192.0.2.33`
Create the node for EAST, setting the IP address to “any”, and set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 0.0.0.0 authentication mode pre-shared-secret`

Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 0.0.0.0`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Provide the string that will be used to generate encryption keys.	`vyatta@WEST# set authentication pre-shared-secret test_key_1`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

TABLE 24 Creating a site-to-site connection to a peer with a dynamic IP address via NAT (Continued)

Step	Command

Create a tunnel configuration, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 1 local prefix 192.168.40.0/24`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Provide the remote subnet for the tunnel.	`vyatta@WEST# set tunnel 1 remote prefix 192.168.60.0/24`
	
	`[edit security vpn ipsec site-to-site peer 0.0.0.0]`

Return to the top of the configuration tree.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

View the configuration for the site-to-site connection.	`vyatta@WEST#`
	`show security vpn ipsec site-to-site peer 0.0.0.0`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`

View data plane interface dp0p1p2 address configuration. local-address is set to this address.	`vyatta@WEST#`
	`show interfaces dataplane dp0p1p2 address`
	`address 192.0.2.1/27`

Configure EAST

Similar to the WEST configuration, EAST must be configured for NAT traversal, but the connection from EAST to WEST requires only a minor change (*local-address*) from that configured in the section Basic site-to-site connection on page 21.

- The NAT device keeps track of EAST's fixed IP and correctly routes incoming packets to EAST, making any necessary changes to outgoing packets.
- WEST retains its fixed IP, so no modification is required to the remote peer IP address.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 25 Specify a new local-address and that NAT must be traversed

Step	Command

Enable NAT traversal.	`vyatta@EAST# set security vpn ipsec nat-traversal enable`
	
	`[edit]`

TABLE 25 Specify a new local-address and that NAT must be traversed (Continued)

Step	Command

Allow private network 10.0.0.0/8.	`vyatta@EAST# set security vpn ipsec nat-networks allowed-network 10.0.0.0/8`
	
	`[edit]`

Allow private network 172.16.0.0/12.	`vyatta@EAST# set security vpn ipsec nat-networks allowed-network 172.16.0.0/12`
	
	`[edit]`

Allow private network 192.168.0.0/16 but exclude the local subnet (192.168.60.0/24).	`vyatta@EAST# set security vpn ipsec nat-networks allowed-network 192.168.0.0/16 exclude 192.168.60.0/24`
	
	`[edit]`

Commit the configuration.	`vyatta@EAST# commit`

View the newly added configuration (only the relevant parts of the configuration are shown).	`vyatta@EAST# show security vpn ipsec`
	
	`(...)`
	`  nat-networks {`
	`  allowed-network 10.0.0.0/8 {`
	`  }`
	`  allowed-network 172.16.0.0/12 {`
	`  }`
	`  allowed-network 192.168.0.0/16 {`
	`  exclude 192.168.60.0/24`
	`  }`
	`  }`
	`  nat-traversal enable`
	`(...)`

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 local-address 192.168.0.1`
	
	`[edit]`

Commit the configuration.	`vyatta@EAST# commit`

View the modified configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`  authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`  }`
	`  default-esp-group ESP-1E`
	`  ike-group IKE-1E`
	`  local-address 192.168.0.1`
	`  tunnel 1 {`
	`  local {`
	`  prefix 192.168.60.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.40.0/24`
	`  }`
	`  }`

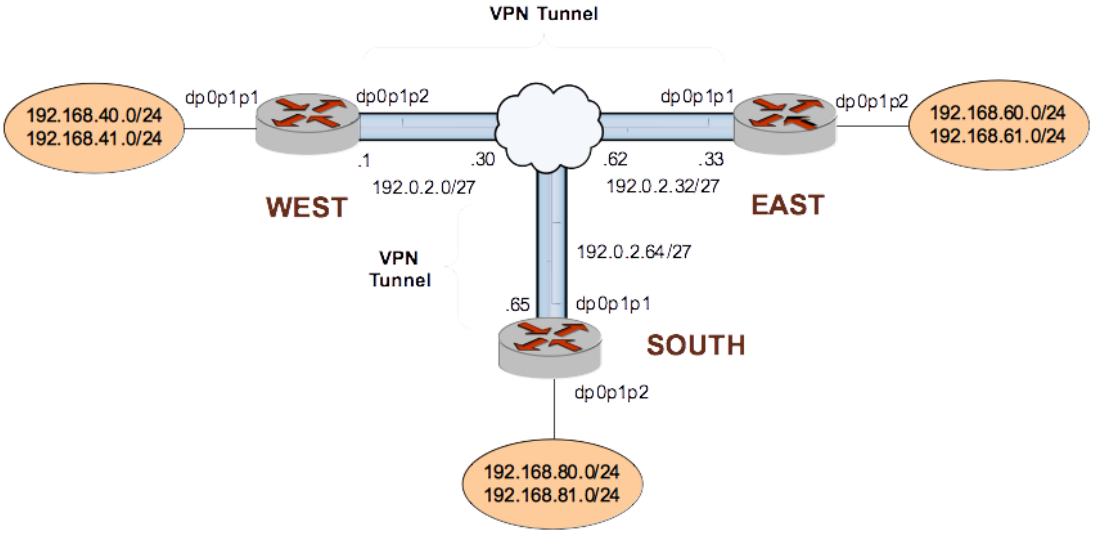
View data plane interface dp0p1p1 address configuration. local-address is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address address 192.168.0.1/24`

IPsec tunnels between three gateways

This section presents the following topics:

This section presents a sample configuration for multiple site-to-site tunnels between three gateways: WEST, EAST, and SOUTH. When you have finished, these peers will be configured as shown in the following figure.

FIGURE 5 Multiple site-to-site tunnels between three gateways



Configure WEST

This section presents the following topics:

This example assumes that WEST has already been configured for a basic connection to EAST, as described in “Configuring a Basic Site-to-Site Connection” on page 157. The additional configuration for WEST for this scenario consists of the following:

- An additional ESP group
- Three new tunnel configurations for the site-to-site connection to EAST
- A new site-to-site connection to SOUTH

This section presents the following examples:

Configuring the second ESP group on WEST

Table 26 creates a second ESP group ESP-2W on WEST. This ESP group contains just one proposal:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm.
- The lifetime of a proposal from this ESP group is set to 600 seconds.

To create this ESP group, perform the following steps on WEST in configuration mode.

TABLE 26 Configuring a second ESP group on WEST

Step	Command
Create the configuration node for proposal 1 of ESP group ESP-2W.	`vyatta@WEST# set security vpn ipsec esp-group ESP-2W proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@WEST# set security vpn ipsec esp-group ESP-2W proposal 1 encryption aes256`
Set the hash algorithm for proposal 1 of ESP-2W.	`vyatta@WEST# set security vpn ipsec esp-group ESP-2W proposal 1 hash sha1`
Set the lifetime for ESP-2W.	`vyatta@WEST# set security vpn ipsec esp-group ESP-2W lifetime 600`

View the configuration for the ESP group. Don't commit yet.	`vyatta@WEST# show security vpn ipsec esp-group ESP-2W`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  lifetime 600`

Adding tunnels to the connection to EAST

Table 27 adds three tunnels to the site-to-site connection from WEST to EAST.

- Tunnel 2 communicates between 192.168.40.0/24 on WEST and 192.168.61.0/24 on EAST, and uses the default ESP group ESP-1W.
- Tunnel 3 communicates between 192.168.41.0/24 on WEST and 192.168.60.0/24 on EAST, and uses ESP group ESP-2W.
- Tunnel 4 communicates between 192.168.41.0/24 on WEST and 192.168.61.0/24 on EAST, and uses ESP group ESP-2W.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 27 Adding tunnels to the connection to EAST

Step	Command

Navigate to the configuration node for EAST for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 192.0.2.33`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 2 local prefix 192.168.40.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

TABLE 27 Adding tunnels to the connection to EAST (Continued)

Step	Command

Provide the remote subnet for tunnel 2.	`vyatta@WEST# set tunnel 2 remote prefix 192.168.61.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 3 local prefix 192.168.41.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Provide the remote subnet for tunnel 3.	`vyatta@WEST# set tunnel 3 remote prefix 192.168.60.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify the ESP group for tunnel 3.	`vyatta@WEST# set tunnel 3 esp-group ESP-2W`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 4 local prefix 192.168.41.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Provide the remote subnet for tunnel 4.	`vyatta@WEST# set tunnel 4 remote prefix 192.168.61.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify the ESP group for tunnel 4.	`vyatta@WEST# set tunnel 4 esp-group ESP-2W`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Return to the top of the configuration tree.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

TABLE 27 Adding tunnels to the connection to EAST (Continued)

Step	Command

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33`
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`
	`tunnel 2 {`
	`local {`
	`prefix 192.168.40.0/24`
	`}`
	`remote {`
	`prefix 192.168.61.0/24`
	`}`
	`}`
	`tunnel 3 {`
	`esp-group ESP-2W`
	`local {`
	`prefix 192.168.41.0/24`
	`}`
	`remote {`
	`prefix 192.168.60.0/24`
	`}`
	`}`
	`tunnel 4 {`
	`esp-group ESP-2W`
	`local {`
	`prefix 192.168.41.0/24`
	`}`
	`remote {`
	`prefix 192.168.61.0/24`
	`}`
	`}`

View dat plane interface dp0p1p2 address configuration. **local-address** is set to this address.	`vyatta@WEST# show interfaces dataplane dp0p1p2 address address 192.0.2.1/27`

Create the connection to SOUTH

Table 28 defines a site-to-site connection from WEST to SOUTH.

- The connection has four tunnels:
  - Tunnel 1 communicates between 192.168.40.0/24 on WEST and 192.168.80.0/24 on SOUTH, and uses the default ESP group ESP-1W.
  - Tunnel 2 communicates between 192.168.40.0/24 on WEST and 192.168.81.0/24 on SOUTH, and uses the default ESP group ESP-1W.
  - Tunnel 3 communicates between 192.168.41.0/24 on WEST and 192.168.80.0/24 on SOUTH, and uses the default ESP group ESP-1W.
  - Tunnel 4 communicates between 192.168.41.0/24 on WEST and 192.168.81.0/24 on SOUTH, and uses the default ESP group ESP-1W.
- WEST uses IP address 192.0.2.1 on dp0p1p2.
- SOUTH uses IP address 192.0.2.65 on dp0p1p1.

- The IKE group is IKE-1W
- The preshared secret is “test_key_2”.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 28 Creating a site-to-site connection from WEST to SOUTH

Step	Command
Create the node for SOUTH and set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.65 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 192.0.2.65` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the string that will be used to generate encryption keys.	`vyatta@WEST# set authentication pre-shared-secret test_key_2` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 1, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 1 local prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 1.	`vyatta@WEST# set tunnel 1 remote prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 2 local prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 2.	`vyatta@WEST# set tunnel 2 remote prefix 192.168.81.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 3 local prefix 192.168.41.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 3.	`vyatta@WEST# set tunnel 3 remote prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`

TABLE 28 Creating a site-to-site connection from WEST to SOUTH (Continued)

Step	Command

Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@WEST# set tunnel 4 local prefix 192.168.41.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.65]`

Provide the remote subnet for tunnel 4.	`vyatta@WEST# set tunnel 4 remote prefix 192.168.81.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.65]`

Return to the top of the configuration tree.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.65`
	
	`authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_2`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`  local {`
	`  prefix 192.168.40.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.80.0/24`
	`  }`
	`}`
	`tunnel 2 {`
	`  local {`
	`  prefix 192.168.40.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.81.0/24`
	`  }`
	`}`
	`tunnel 3 {`
	`  local {`
	`  prefix 192.168.41.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.80.0/24`
	`  }`
	`}`
	`tunnel 4 {`
	`  local {`
	`  prefix 192.168.41.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.81.0/24`
	`  }`
	`}`

Configure EAST

This section presents the following topics:

This example assumes that EAST has already been configured for a basic connection to WEST, as described in Basic site-to-site connection on page 21. The additional configuration for EAST for this scenario consists of the following:

- An additional ESP group
- Three new tunnel configurations for the site-to-site connection to WEST
- A new site-to-site connection to SOUTH

This section presents the following examples:

Configuring the second ESP group on EAST

Table 29 creates a second ESP group ESP-2W on EAST. This ESP group contains just one proposal:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm

The lifetime of a proposal from this ESP group is set to 600 seconds.

To create this ESP group, perform the following steps on EAST in configuration mode.

TABLE 29 Configuring a second ESP group on EAST

Step	Command
Create the configuration node for proposal 1 of ESP group ESP-2E.	`vyatta@EAST# set security vpn ipsec esp-group ESP-2E proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@EAST# set security vpn ipsec esp-group ESP-2E proposal 1 encryption aes256`
Set the hash algorithm for proposal 1 of ESP-2E.	`vyatta@EAST# set security vpn ipsec esp-group ESP-2E proposal 1 hash sha1`
Set the lifetime for ESP-2E.	`vyatta@EAST# set security vpn ipsec esp-group ESP-2E lifetime 600`

View the configuration for the ESP group. Don't commit yet.	`vyatta@EAST# show security vpn ipsec esp-group ESP-2E`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  lifetime 600`

Adding tunnels to the connection to WEST

Table 30 adds three tunnels to the site-to-site connection from EAST to WEST.

- Tunnel 2 communicates between 192.168.60.0/24 on EAST and 192.168.41.0/24 on WEST, and uses the default ESP group ESP-1E.
- Tunnel 3 communicates between 192.168.61.0/24 on EAST and 192.168.40.0/24 on WEST, and uses ESP group ESP-2E.
- Tunnel 4 communicates between 192.168.61.0/24 on EAST and 192.168.41.0/24 on WEST, and uses ESP group ESP-2E.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 30 Adding tunnels to the connection to WEST

Step	Command
Navigate to the configuration node for WEST for easier editing.	`vyatta@EAST# edit security vpn ipsec site-to-site peer 192.0.2.1` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 2 local prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 2.	`vyatta@EAST# set tunnel 2 remote prefix 192.168.41.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 3 local prefix 192.168.61.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 3.	`vyatta@EAST# set tunnel 3 remote prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Specify the ESP group for tunnel 3.	`vyatta@EAST# set tunnel 3 esp-group ESP-2E` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 4 local prefix 192.168.61.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 4.	`vyatta@EAST# set tunnel 4 remote prefix 192.168.41.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Specify the ESP group for tunnel 4.	`vyatta@EAST# set tunnel 4 esp-group ESP-2E` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Return to the top of the configuration tree.	`vyatta@EAST# top`
Commit the configuration.	`vyatta@EAST# commit`

TABLE 30 Adding tunnels to the connection to WEST (Continued)

Step	Command

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	`authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`  local {`
	`  prefix 192.168.60.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.40.0/24`
	`  }`
	`}`
	`tunnel 2 {`
	`  local {`
	`  prefix 192.168.60.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.41.0/24`
	`  }`
	`}`
	`tunnel 3 {`
	`  esp-group ESP-2E`
	`  local {`
	`  prefix 192.168.61.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.40.0/24`
	`  }`
	`}`
	`tunnel 4 {`
	`  esp-group ESP-2E`
	`  local {`
	`  prefix 192.168.61.0/24`
	`  }`
	`  remote {`
	`  prefix 192.168.41.0/24`
	`  }`
	`}`

View data plane interface dp0p1p1 address configuration. **local-address** is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address address 192.0.2.33/27`

Creating the connection to SOUTH

Table 31 defines a site-to-site connection from EAST to SOUTH.

- The connection has four tunnels:
  - Tunnel 1 communicates between 192.168.60.0/24 on EAST and 192.168.80.0/24 on SOUTH, and uses the default ESP group ESP-1E.
  - Tunnel 2 communicates between 192.168.60.0/24 on EAST and 192.168.81.0/24 on SOUTH, and uses the default ESP group ESP-1E.
  - Tunnel 3 communicates between 192.168.61.0/24 on EAST and 192.168.80.0/24 on SOUTH, and uses the default ESP group ESP-1E.
  - Tunnel 4 communicates between 192.168.61.0/24 on EAST and 192.168.81.0/24 on SOUTH, and uses the default ESP group ESP-1E.
- EAST uses IP address 192.0.2.33 on dp0p1p2.
- SOUTH uses IP address 192.0.2.65 on dp0p1p1.

- The IKE group is IKE-1E.
- The preshared secret is “test_key_2”.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 31 Creating a site-to-site connection from EAST to SOUTH

Step	Command
Create the node for SOUTH and set the authentication mode.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.65 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@EAST# edit security vpn ipsec site-to-site peer 192.0.2.65` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the string that will be used to generate encryption keys.	`vyatta@EAST# set authentication pre-shared-secret test_key_2` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Specify the default ESP group.	`vyatta@EAST# set default-esp-group ESP-1E` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Specify the IKE group.	`vyatta@EAST# set ike-group IKE-1E` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@EAST# set local-address 192.0.2.33` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 1, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 1 local prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 1.	`vyatta@EAST# set tunnel 1 remote prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 2 local prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 2.	`vyatta@EAST# set tunnel 2 remote prefix 192.168.81.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 3 local prefix 192.168.61.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`
Provide the remote subnet for tunnel 3.	`vyatta@EAST# set tunnel 3 remote prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.65]`

TABLE 31 Creating a site-to-site connection from EAST to SOUTH (Continued)

Step	Command

Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@EAST# set tunnel 4 local prefix 192.168.61.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.65]`

Provide the remote subnet for tunnel 4.	`vyatta@EAST# set tunnel 4 remote prefix 192.168.81.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.65]`

Return to the top of the configuration tree.	`vyatta@EAST# top`
Commit the configuration.	`vyatta@EAST# commit`

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.65`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_2`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.60.0/24`
	`}`
	`remote {`
	`prefix 192.168.80.0/24`
	`}`
	`}`
	`tunnel 2 {`
	`local {`
	`prefix 192.168.60.0/24`
	`}`
	`remote {`
	`prefix 192.168.81.0/24`
	`}`
	`}`
	`tunnel 3 {`
	`local {`
	`prefix 192.168.61.0/24`
	`}`
	`remote {`
	`prefix 192.168.80.0/24`
	`}`
	`}`
	`tunnel 4 {`
	`local {`
	`prefix 192.168.61.0/24`
	`}`
	`remote {`
	`prefix 192.168.81.0/24`
	`}`
	`}`

Configure SOUTH

This section presents the following topics:

This section presents the following examples:

Configuring an IKE group on SOUTH

Table 32 creates IKE group IKE-1S on SOUTH. This IKE group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm.
- Proposal 2 uses AES-128 as the encryption cipher and SHA-1 as the hash algorithm.

The lifetime of a proposal from this IKE group is set to 3600.

Note that these parameters correspond to those set in IKE-1W on WEST and IKE-1E on EAST. You must ensure, in defining proposals, that the encryption ciphers and hash algorithms are such that the two peers will be able to agree on a combination.

To create this IKE group, perform the following steps on SOUTH in configuration mode.

TABLE 32 Configuring an IKE group on SOUTH

Step	Command
Creates the configuration node for proposal 1 of IKE group IKE-1S.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S proposal 1 hash sha1`
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of IKE group IKE-1S.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S proposal 2 encryption aes128`
Set the hash algorithm for proposal 2.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S proposal 2 hash sha1`
Set the lifetime for the whole IKE group.	`vyatta@SOUTH# set security vpn ipsec ike-group IKE-1S lifetime 3600`

View the configuration for the IKE group. Don't commit yet.	`vyatta@SOUTH# show security vpn ipsec ike-group IKE-1S`
	`>    proposal 1 {`
	`>        encryption aes256`
	`>        hash sha1`
	`>    }`
	`>    proposal 2 {`
	`>        encryption aes128`
	`>        hash sha1`
	`>    }`
	`>    lifetime 3600`

Configuring an ESP group on SOUTH

Table 33 creates ESP group ESP-1S on SOUTH. This ESP group contains two proposals:

- Proposal 1 uses AES-256 as the encryption cipher and SHA-1 as the hash algorithm.
- Proposal 2 uses Triple-DES as the encryption cipher and MD5 as the hash algorithm.

The lifetime of a proposal from this ESP group is set to 1800 seconds.

To create this ESP group, perform the following steps on SOUTH in configuration mode.

TABLE 33 Configuring an ESP group on SOUTH

Step	Command
Create the configuration node for proposal 1 of ESP group ESP-1S.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S proposal 1`
Set the encryption cipher for proposal 1.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S proposal 1 encryption aes256`
Set the hash algorithm for proposal 1.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S proposal 1 hash sha1`
Set the encryption cipher for proposal 2. This also creates the configuration node for proposal 2 of ESP group ESP-1S.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S proposal 2 encryption 3des`
Set the hash algorithm for proposal 2.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S proposal 2 hash md5`
Set the lifetime for the whole ESP group.	`vyatta@SOUTH# set security vpn ipsec esp-group ESP-1S lifetime 1800`

View the configuration for the ESP group. Don't commit yet.	`vyatta@SOUTH# show security vpn ipsec esp-group ESP-1S`
	
	`>  proposal 1 {`
	`>  encryption aes256`
	`>  hash sha1`
	`>  }`
	`>  proposal 2 {`
	`>  encryption 3de`
	`>  hash md5`
	`>  }`
	`>  lifetime 1800`

Creating the connection to WEST

Table 34 defines a site-to-site connection to WEST.

- This connection is configured with four tunnels:
  - Tunnel 1 communicates between 192.168.80.0/24 on SOUTH and 192.168.40.0/24 on WEST, and uses the default ESP group ESP-1S.
  - Tunnel 2 communicates between 192.168.80.0/24 on SOUTH and 192.168.41.0/24 on WEST, and uses the default ESP group ESP-1S.
  - Tunnel 3 communicates between 192.168.81.0/24 on SOUTH and 192.168.40.0/24 on WEST, and uses the default ESP group ESP-1S.
  - Tunnel 4 communicates between 192.168.81.0/24 on SOUTH and 192.168.41.0/24 on WEST, and uses the default ESP group ESP-1S.
- SOUTH uses IP address 192.0.2.65 on dp0p1p1.
- WEST uses IP address 192.0.2.1 on dp0p1p2.
- The IKE group is IKE-1S.
- The preshared secret is “test_key_2”.

To configure this connection, perform the following steps on SOUTH in configuration mode.

TABLE 34 Creating a site-to-site connection from SOUTH to WEST

Step	Command
Create the node for WEST and set the authentication mode.	`vyatta@SOUTH# set security vpn ipsec site-to-site peer 192.0.2.1 authentication mode pre-shared-secret`
Navigate to the node for the peer for easier editing.	`vyatta@SOUTH# edit security vpn ipsec site-to-site peer 192.0.2.1` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the string that will be used to generate encryption keys.	`vyatta@SOUTH# set authentication pre-shared-secret test_key_2` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Specify the default ESP group.	`vyatta@SOUTH# set default-esp-group ESP-1S` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Specify the IKE group.	`vyatta@SOUTH#set ike-group IKE-1S` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@SOUTH# set local-address 192.0.2.65` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 1, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 1 local prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 1.	`vyatta@SOUTH# set tunnel 1 remote prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 2 local prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 2.	`vyatta@SOUTH# set tunnel 2 remote prefix 192.168.41.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 3 local prefix 192.168.81.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`
Provide the remote subnet for tunnel 3.	`vyatta@SOUTH# set tunnel 3 remote prefix 192.168.40.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.1]`

TABLE 34 Creating a site-to-site connection from SOUTH to WEST (Continued)

Step	Command

Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 4 local prefix 192.168.81.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Provide the remote subnet for tunnel 4.	`vyatta@SOUTH# set tunnel 4 remote prefix 192.168.41.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Return to the top of the configuration tree.	`vyatta@SOUTH# top`
Now commit the configuration.	`vyatta@SOUTH# commit`

View the configuration for the site-to-site connection.	`vyatta@SOUTH# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_2`
	`}`
	`default-esp-group ESP-1S`
	`ike-group IKE-1S`
	`local-address 192.0.2.65`
	`tunnel 1 {`
	`local {`
	`prefix 192.168.80.0/24`
	`}`
	`remote {`
	`prefix 192.168.40.0/24`
	`}`
	`}`
	`tunnel 2 {`
	`local {`
	`prefix 192.168.80.0/24`
	`}`
	`remote {`
	`prefix 192.168.41.0/24`
	`}`
	`}`
	`tunnel 3 {`
	`local {`
	`prefix 192.168.81.0/24`
	`}`
	`remote {`
	`prefix 192.168.40.0/24`
	`}`
	`}`
	`tunnel 4 {`
	`local {`
	`prefix 192.168.81.0/24`
	`}`
	`remote {`
	`prefix 192.168.41.0/24`
	`}`
	`}`

View data plane interface dp0p1p1 address configuration. **local-address** is set to this address.	`vyatta@SOUTH# show interfaces dataplane dp0p1p1 address address 192.0.2.65/27`

Creating the connection to EAST

Table 35 defines a site-to-site connection to EAST.

- This connection is configured with four tunnels:
  - Tunnel 1 communicates between 192.168.80.0/24 on SOUTH and 192.168.60.0/24 on EAST, and uses the default ESP group ESP-1S.
  - Tunnel 2 communicates between 192.168.80.0/24 on SOUTH and 192.168.61.0/24 on EAST, and uses the default ESP group ESP-1S.
  - Tunnel 3 communicates between 192.168.81.0/24 on SOUTH and 192.168.60.0/24 on EAST, and uses the default ESP group ESP-1S.
  - Tunnel 4 communicates between 192.168.81.0/24 on SOUTH and 192.168.61.0/24 on EAST, and uses the default ESP group ESP-1S.
- SOUTH uses IP address 192.0.2.65 on dp0p1p1.
- EAST uses IP address 192.0.2.33 on dp0p1p2.
- The IKE group is IKE-1S.
- The preshared secret is “test_key_2”.

To configure this connection, perform the following steps on SOUTH in configuration mode.

TABLE 35 Creating a site-to-site connection from SOUTH to EAST

Step	Command
Create the node for EAST and set the authentication mode.	`vyatta@SOUTH# set security vpn ipsec site-to-site peer 192.0.2.33`
Navigate to the node for the peer for easier editing.	`vyatta@SOUTH# edit security vpn ipsec site-to-site peer 192.0.2.33` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Provide the string that will be used to generate encryption keys.	`vyatta@SOUTH# set authentication pre-shared-secret test_key_2` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Specify the default ESP group.	`vyatta@SOUTH# set default-esp-group ESP-1S` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Specify the IKE group.	`vyatta@SOUTH# set ike-group IKE-1S` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@SOUTH# set local-address 192.0.2.65` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Create the configuration node for tunnel 1, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 1 local prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Provide the remote subnet for tunnel 1.	`vyatta@SOUTH# set tunnel 1 remote prefix 192.168.60.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`
Create the configuration node for tunnel 2, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 2 local prefix 192.168.80.0/24` `[edit security vpn ipsec site-to-site peer 192.0.2.33]`

TABLE 35 Creating a site-to-site connection from SOUTH to EAST (Continued)

Step	Command

Provide the remote subnet for tunnel 2.	`vyatta@SOUTH# set tunnel 2 remote prefix 192.168.61.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Create the configuration node for tunnel 3, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 3 local prefix 192.168.81.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Provide the remote subnet for tunnel 3.	`vyatta@SOUTH# set tunnel 3 remote prefix 192.168.60.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Create the configuration node for tunnel 4, and provide the local subnet for this tunnel.	`vyatta@SOUTH# set tunnel 4 local prefix 192.168.81.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Provide the remote subnet for tunnel 4.	`vyatta@SOUTH# set tunnel 4 remote prefix 192.168.61.0/24`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Return to the top of the configuration tree.	`vyatta@SOUTH#` `top`
Now commit the configuration.	`vyatta@SOUTH# commit`

View the configuration for the site-to-site connection.	`vyatta@SOUTH# show security vpn ipsec site-to-site peer 192.0.2.33`
	
	`authentication`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_2`
	`}`
	`default-esp-group ESP-1S`
	`ike-group IKE-1S`
	`local-address 192.0.2.65`

TABLE 35 Creating a site-to-site connection from SOUTH to EAST (Continued)

Step	Command

	tunnel 1 {
	local {
	prefix 192.168.80.0/24
	}
	remote {
	prefix 192.168.60.0/24
	}
	}
	tunnel 2 {
	local {
	prefix 192.168.80.0/24
	}
	remote {
	prefix 192.168.61.0/24
	}
	}
	tunnel 3 {
	local {
	prefix 192.168.81.0/24
	}
	remote {
	prefix 192.168.60.0/24
	}
	}
	tunnel 4 {
	local {
	prefix 192.168.81.0/24
	}
	remote {
	prefix 192.168.61.0/24
	}
	}

GRE tunnel protected with IPsec

GRE, IP-in-IP, and SIT tunnels are not encrypted, and provide no security outside of a simple password-like key that is exchanged in clear text in each packet. This means that GRE, IP-in-IP, and SIT tunnels, on their own, do not provide adequate security for production environments.

At the same time, IPsec policy-based tunnels cannot directly route non-IP or multicast protocols, and IPsec also has limitations from an operations point of view. Using tunnel interfaces in conjunction with IPsec VPN provides secure, routable tunnel connections between gateways, that have some advantages over traditional IPsec policy-based tunnel mode connections:

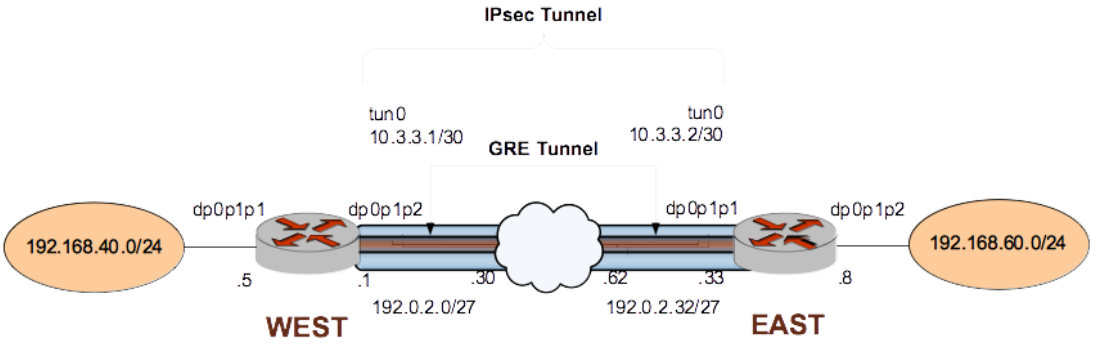
- Support for standard operational commands such as **show interfaces** and **show route**
- Support for operational tools such as **traceroute** and SNMP
- Dynamic tunnel failover using routing protocols
- Simplified IPsec policies and troubleshooting

For secure routable tunnels, GRE, IP-in-IP, and SIT tunnel interfaces should be used in conjunction with an IPsec connection, so that the IP tunnel can be protected by the IPsec tunnel.

This set of examples configures a GRE tunnel between EAST to WEST and protects it within an IPsec tunnel between the same endpoints.

When you have finished, WEST and EAST will be configured as shown in the following figure.

FIGURE 6 GRE tunnel protected by an IPsec tunnel



Configure WEST

This section presents the following examples:

- Defining the GRE tunnel on WEST on page 68
- Defining the IPsec tunnel on WEST on page 69
- Defining a static route on WEST on page 70

Defining the GRE tunnel on WEST

For details on GRE tunnels, refer to *Brocade 5600 vRouter Tunnels Reference Guide*.

Table 36 defines WEST's end of the GRE tunnel. In this example:

- The tunnel interface tun0 on router WEST is assigned the IP address 10.3.3.1/30.
- The encapsulation type is set to GRE.
- The IP address on the local side of the GRE tunnel (**local-ip**) is set to that of the local data plane interface (192.0.2.1).
- The IP address of the other end of the GRE tunnel (**remote-ip**) is set to the address of the remote system (192.0.2.33).
- Multicast is enabled in order to allow routing protocols to be carried on the GRE tunnel.

To create the tunnel interface and the tunnel endpoint on WEST, perform the following steps in configuration mode.

TABLE 36 Defining the GRE tunnel from WEST to EAST

Step	Command
Create the GRE tunnel interface, and specify the IP address to be associated with it.	`vyatta@WEST# set interfaces tunnel tun0 address 10.3.3.1/30`
Assign a brief description for the GRE tunnel interface.	`vyatta@WEST# set interfaces tunnel tun0 description "GRE tunnel to router EAST"`
Specify the encapsulation mode for the tunnel.	`vyatta@WEST# set interfaces tunnel tun0 encapsulation gre`
Allow multicast protocols (e.g., routing protocols) to be carried over the tunnel.	`vyatta@WEST# set interfaces tunnel tun0 multicast enable`
Specify the local IP address for the GRE tunnel.	`vyatta@WEST# set interfaces tunnel tun0 local-ip 192.0.2.1`

TABLE 36 Defining the GRE tunnel from WEST to EAST (Continued)

Step	Command
Specify the remote IP address for the GRE tunnel.	`vyatta@WEST# set interfaces tunnel tun0 remote-ip 192.0.2.33`
Commit the configuration.	`vyatta@WEST# commit`

View the modified configuration.	`vyatta@WEST# show interfaces tunnel tun0`
	
	`address 10.3.3.1/30`
	`description "GRE tunnel to router EAST"`
	`encapsulation gre`
	`local-ip 192.0.2.1`
	`multicast enable`
	`remote-ip 192.0.2.33`

Defining the IPsec tunnel on WEST

Table 37 creates the IPsec tunnel from WEST to EAST.

- WEST uses IP address 192.0.2.1 on dp0p1p2.
- EAST uses IP address 192.0.2.33 on dp0p1p1.
- The IKE group is IKE-1W.
- The preshared secret is "test_key_1".
- All GRE traffic will be passed through the tunnel.

This examples assumes that you have already configured the following:

- IKE group IKE-1W (see Configure an IKE group on WEST on page 22)
- ESP group ESP-1W (see Configure an ESP group on WEST on page 23)

To create the IPsec tunnel from WEST to EAST, perform the following steps on WEST in configuration mode.

TABLE 37 Defining the IPsec tunnel from WEST to EAST

Step	Command
Define the site-to-site connection to EAST. Set the authentication mode.	`vyatta@WEST# set security vpn ipsec site-to-site peer 192.0.2.33 authentication mode pre-shared-secret`

Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 192.0.2.33`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Provide the string that will be used to authenticate the peers.	`vyatta@WEST# set authentication pre-shared-secret test_key_1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify the default ESP group for all tunnels.	`vyatta@WEST# set default-esp-group ESP-1W`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify the IKE group.	`vyatta@WEST# set ike-group IKE-1W`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

TABLE 37 Defining the IPsec tunnel from WEST to EAST (Continued)

Step	Command

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@WEST# set local-address 192.0.2.1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify that only GRE traffic will pass through the tunnel.	`vyatta@WEST# set tunnel 1 protocol gre`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Return to the top of the configuration hierarchy.	`vyatta@WEST# top`
Commit the configuration.	`vyatta@WEST# commit`

View the modified configuration.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33`
	
	`authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1W`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`tunnel 1 {`
	`  protocol gre`
	`}`

Defining a static route on WEST

Table 38 creates the static route for traffic destined for the far end of the GRE tunnel.

NOTE
Routing protocols can be used to specify how to get to the remote network. This method simply provides the minimal requirement to achieve this.

- Send traffic destined for 192.168.60.0/24 to the far end of the GRE tunnel at 10.3.3.2.

To create the static route, perform the following steps on WEST in configuration mode.

TABLE 38 Defining a static route on WEST

Step	Command
Create the static route.	`vyatta@WEST# set protocols static route 192.168.60.0/24 next-hop 10.3.3.2`
Commit the configuration.	`vyatta@WEST# commit`

View the modified configuration.	`vyatta@WEST# show protocols static route`
	
	`192.168.60.0/24 {`
	`  next-hop 10.3.3.2`
	`}`

Configure EAST

This section presents the following examples:

Defining the GRE tunnel on EAST

For details on GRE tunnels, refer to *Brocade 5600 vRouter Tunnels Reference Guide*.

Table 39 defines EAST's end of the GRE tunnel. In this example:

- The tunnel interface tun0 on router EAST is assigned the IP address 10.3.3.2/30.
- The encapsulation type is set to GRE.
- The IP address on the local side of the GRE tunnel (**local-ip**) is set to that of the local data plane interface (192.0.2.33).
- The IP address of the other end of the GRE tunnel (**remote-ip**) is set to the address of the remote system (192.0.2.1).

To create the tunnel interface and the tunnel endpoint on EAST, perform the following steps in configuration mode.

TABLE 39 Defining the GRE tunnel from EAST to WEST

Step	Command
Create the GRE tunnel interface, and specify the IP address to be associated with it.	`vyatta@EAST# set interfaces tunnel tun0 address 10.3.3.2/30`
Assign a brief description for the GRE tunnel interface.	`vyatta@EAST# set interfaces tunnel tun0 description "GRE tunnel to router WEST"`
Specify the encapsulation mode for the tunnel.	`vyatta@EAST# set interfaces tunnel tun0 encapsulation gre`
Allow multicast protocols (e.g., routing protocols) to be carried over the tunnel.	`vyatta@EAST# set interfaces tunnel tun0 multicast enable`
Specify the local IP address for the GRE tunnel.	`vyatta@EAST# set interfaces tunnel tun0 local-ip 192.0.2.33`
Specify the remote IP address for the GRE tunnel.	`vyatta@EAST# set interfaces tunnel tun0 remote-ip 192.0.2.1`
Commit the configuration.	`vyatta@EAST# commit`

View the modified configuration.	`vyatta@EAST# show interfaces tunnel tun0`
	
	`address 10.3.3.2/30`
	`description "GRE tunnel to router WEST"`
	`encapsulation gre`
	`local-ip 192.0.2.33`
	`multicast enable`
	`remote-ip 192.0.2.1`

Defining the IPsec tunnel on EAST

Table 40 creates the IPsec tunnel from EAST to WEST.

- EAST uses IP address 192.0.2.33 on dp0p1p1.
- WEST uses IP address 192.0.2.1 on dp0p1p2.
- The IKE group is IKE-1E.
- The preshared secret is “test_key_1”.
- All GRE traffic will be passed through the tunnel.

This examples assumes that you have already configured the following:

- IKE group IKE-1E (see Configure an IKE group on EAST on page 27)
- ESP group ESP-1E (see Configure an ESP group on EAST on page 27)

To create the IPsec tunnel from EAST to WEST, perform the following steps on EAST in configuration mode.

TABLE 40 Defining the IPsec tunnel from EAST to WEST

Step	Command
Define the site-to-site connection to WEST. Set the authentication mode.	`vyatta@EAST# set security vpn ipsec site-to-site peer 192.0.2.1 authentication mode pre-shared-secret`

Navigate to the node for the peer for easier editing.	`vyatta@EAST# edit security vpn ipsec site-to-site peer 192.0.2.1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Provide the string that will be used to authenticate the peers.	`vyatta@EAST# set authentication pre-shared-secret test_key_1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify the default ESP group for all tunnels.	`vyatta@EAST# set default-esp-group ESP-1E`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify the IKE group.	`vyatta@EAST# set ike-group IKE-1E`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Identify the IP address on this Brocade vRouter to be used for this connection.	`vyatta@EAST# set local-address 192.0.2.33`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify that only GRE traffic will pass through the tunnel.	`vyatta@EAST# set tunnel 1 protocol gre`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Return to the top of the configuration hierarchy.	`vyatta@EAST# top`
Commit the configuration.	`vyatta@EAST# commit`

View the modified configuration.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`}`
	`default-esp-group ESP-1E`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`tunnel 1 {`
	`  protocol gre`
	`}`

View data plane interface dp0p1p1 address configuration. **local-address** is set to this address.	`vyatta@EAST# show interfaces dataplane dp0p1p1 address address 192.0.2.33/27`

Defining a static route on EAST

Table 41 creates the static route for traffic destined for the far end of the GRE tunnel.

NOTE
Routing protocols can be used to specify how to get to the remote network. This method simply provides the minimal requirement to achieve this.

- Send traffic destined for 192.168.40.0/24 to the far end of the GRE tunnel at 10.3.3.1.

To create the static route, perform the following steps on EAST in configuration mode.

TABLE 41 Defining a static route on EAST

Step	Command
Create the static route.	`vyatta@EAST# set protocols static route 192.168.40.0/24 next-hop 10.3.3.1`
Commit the configuration.	`vyatta@EAST# commit`

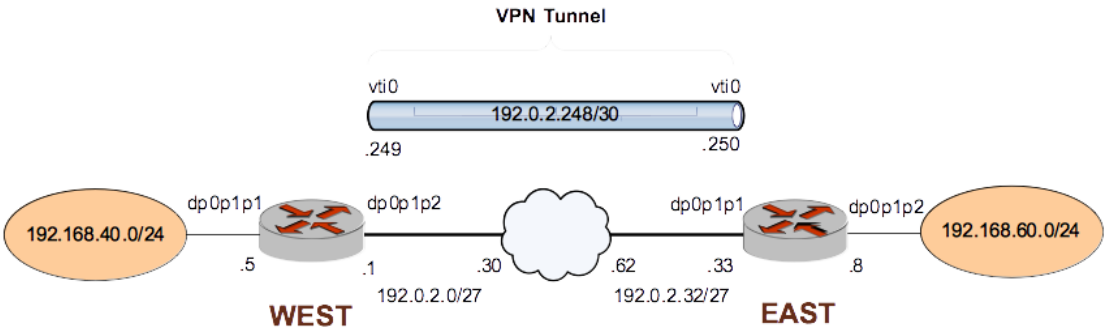
View the modified configuration.	`vyatta@EAST# show protocols static route`
	`192.168.40.0/24 {`
	`  next-hop 10.3.3.1`
	`}`

Basic site-to-site connection using a virtual tunnel interface

This section presents a sample configuration for a connection between WEST and EAST, where a virtual tunnel interface is bound to each end of an IPsec VPN connection. The advantage of this is that, when bound to a virtual tunnel interface, the VPN can be treated like any other routable interface.

In this example, you modify the VPN connection configured in the basic site-to-site IPsec VPN connection created in a previous example (see Basic site-to-site connection on page 21). The resulting configuration provides a virtual tunnel interface on both ends of the VPN tunnel. When you have finished, these systems will be configured as shown in Basic site-to-site connection over IPv6 on page 76.

FIGURE 7 IPsec VPN connection with virtual tunnel interfaces



This example assumes that you have already configured a basic site-to-site connection using a preshared key between WEST and EAST, as explained in the section Basic site-to-site connection on page 21. Only the relevant changes to that configuration are presented here.

Configure WEST

Table 42 defines configuration required to create a virtual tunnel interface on WEST.

To configure this interface, perform the following steps on WEST in configuration mode.

TABLE 42 Creating a virtual tunnel interface on WEST

Step	Command

Create the vti interface and assign it an IP address.	`vyatta@WEST# set interfaces vti vti0 address 192.0.2.249/30`
	
	`[edit]`

Commit the configuration.	`vyatta@WEST# commit`

View the configuration.	`vyatta@WEST# show interfaces vti`
	
	`vti0 {`
	`address 192.0.2.249/30`
	`}`

Table 43 defines configuration changes for a new site-to-site connection to EAST.

The main changes from the basic site-to-site configuration are that the tunnel specification and default-esp-group specification are removed, and that the VPN is bound to the virtual tunnel interface created above.

To configure this connection, perform the following steps on WEST in configuration mode.

TABLE 43 Binding the VPN connection to the virtual tunnel interface

Step	Command

Navigate to the node for the peer for easier editing.	`vyatta@WEST# edit security vpn ipsec site-to-site peer 192.0.2.33`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Delete the default-esp-group specification from the previous configuration.	`vyatta@WEST# delete default-esp-group`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Delete the tunnel specification from the previous configuration.	`vyatta@WEST# delete tunnel`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Bind the VPN tunnel to the vti0 interface.	`vyatta@WEST# set vti bind vti0`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Specify the ESP group for the tunnel.	`vyatta@WEST# set vti esp-group ESP-1W`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.33]`

Return to the top of the configuration tree.	`vyatta@WEST# top`

TABLE 43 Binding the VPN connection to the virtual tunnel interface (Continued)

Step	Command
Commit the configuration.	`vyatta@WEST# commit`

View the configuration for the site-to-site connection.	`vyatta@WEST# show security vpn ipsec site-to-site peer 192.0.2.33`
	
	`authentication {`
	`mode pre-shared-secret`
	`pre-shared-secret test_key_1`
	`}`
	`ike-group IKE-1W`
	`local-address 192.0.2.1`
	`vti {`
	`bind vti0`
	`esp-group ESP-1W`
	`}`

Configure EAST

Table 44 defines configuration required to create a virtual tunnel interface on EAST.

To configure this interface, perform the following steps on EAST in configuration mode.

TABLE 44 Creating a virtual tunnel interface on EAST

Step	Command

Create the vti interface and assign it an IP address.	`vyatta@EAST# set interfaces vti vti0 address 192.0.2.250/30`
	
	`[edit]`

Commit the configuration.	`vyatta@EAST# commit`

View the configuration.	`vyatta@EAST# show interfaces vti`
	
	`vti0 {`
	`address 192.0.2.250/30`
	`}`

Table 45 defines configuration changes for a new site-to-site connection to WEST.

- The main changes from the basic site-to-site configuration are that the tunnel specification and default-esp-group specification are removed, and that the VPN is bound to the virtual tunnel interface created above.

To configure this connection, perform the following steps on EAST in configuration mode.

TABLE 45 Binding the VPN connection to the virtual tunnel interface

Step	Command

Navigate to the node for the peer for easier editing.	`vyatta@EAST# edit security vpn ipsec site-to-site peer 192.0.2.1`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.`
	`1]`

TABLE 45 Binding the VPN connection to the virtual tunnel interface (Continued)

Step	Command

Delete the default-esp-group specification from the previous configuration.	`vyatta@EAST# delete default-esp-group`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Delete the tunnel specification from the previous configuration.	`vyatta@EAST# delete tunnel`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Bind the VPN tunnel to the vti0 interface.	`vyatta@EAST# set vti bind vti0`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Specify the ESP group for the tunnel.	`vyatta@EAST# set vti esp-group ESP-1E`
	
	`[edit security vpn ipsec site-to-site peer 192.0.2.1]`

Return to the top of the configuration tree.	`vyatta@EAST# top`
Commit the configuration.	`vyatta@EAST# commit`

View the configuration for the site-to-site connection.	`vyatta@EAST# show security vpn ipsec site-to-site peer 192.0.2.1`
	
	`authentication {`
	`  mode pre-shared-secret`
	`  pre-shared-secret test_key_1`
	`}`
	`ike-group IKE-1E`
	`local-address 192.0.2.33`
	`vti {`
	`  bind vti0`
	`  esp-group ESP-1E`
	`}`

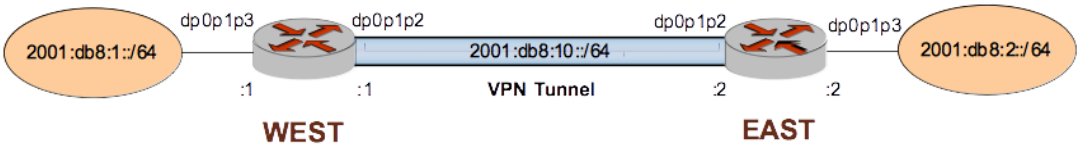
Basic site-to-site connection over IPv6

For the most part, configuring IPsec over IPv6 is the same as configuring IPsec over IPv4. There are three differences to note when configuring IPsec over IPv6.

1. IPv6 addresses are used instead of IPv4 addresses for all arguments that require IP addresses.
2. Hostnames cannot be used. They can only be used when configuring IPsec over IPv4.
3. The **any** keyword cannot be used as the **local-address**. It can only be used when configuring IPsec over IPv4.

   The following example creates a basic site-to-site IPsec connection from WEST to EAST over IPv6.

FIGURE 8 Basic site-to-site IPsec VPN connection over IPv6



Configure WEST

The following example configuration is for the WEST system.

TABLE 46 Basic site-to-site IPsec VPN connection over IPv6 - WEST

Step	Command

View the data plane interface configuration on WEST.	`vyatta@WEST# show interfaces dataplane`
	
	`dataplane dp0p1p2 {`
	`  address 2001:db8:10::1/64`
	`  duplex auto`
	`  hw-id 00:15:5d:00:d5:33`
	`  speed auto`
	`}`
	`dataplane dp0p1p3 {`
	`  address 2001:db8:1::1/64`
	`  duplex auto`
	`  hw-id 00:15:5d:00:d5:34`
	`  speed auto }`
	`}`
	`[edit]`

View the IPv6 IPsec configuration on WEST.	`vyatta@WEST# show security vpn`
	
	`vpn {`
	`  ipsec {`
	`  esp-group ESP-1W {`
	`  compression disable`
	`  lifetime 3600`
	`  mode tunnel`
	`  pfs enable`
	`  proposal 1 {`
	`  encryption aes128`
	`  hash sha1`
	`  }`
	`  }`
	`  ike-group IKE-1W {`
	`  lifetime 28800`
	`  proposal 1 {`
	`  encryption aes128`
	`  hash sha1`
	`  }`
	`  }`

TABLE 46 Basic site-to-site IPsec VPN connection over IPv6 - WEST (Continued)

Step	Command

View the IPv6 IPsec configuration on WEST.	logging {
	log-modes all
	}
	nat-traversal disable
	site-to-site {
	peer 2001:db8:10::2 {
	authentication {
	mode pre-shared-secret
	pre-shared-secret test123
	}
	connection-type initiate
	default-esp-group ESP-1W
	ike-group IKE-1W
	local-address 2001:db8:10::1
	tunnel 1 {
	allow-nat-networks disable
	allow-public-networks disable
	local {
	prefix 2001:db8:1::/64
	}
	remote {
	prefix 2001:db8:2::/64
	}
	}
	}
	}
	}
	}
	[edit]

Configure EAST

The following example configuration is for the EAST system.

TABLE 47 Basic site-to-site IPsec VPN connection over IPv6 - EAST

Step	Command

View the data plane interface configuration on EAST.	vyatta@EAST# show interfaces dataplane
	dataplane dp0p1p2 {
	address 2001:db8:10::2/64
	duplex auto
	hw-id 00:15:5d:00:d5:35
	speed auto
	}
	dataplane dp0p1p3 {
	address 2001:db8:2::2/64
	duplex auto
	hw-id 00:15:5d:00:d5:36
	speed auto }
	}
	[edit]

TABLE 47 Basic site-to-site IPsec VPN connection over IPv6 - EAST (Continued)

Step	Command

View the IPv6 IPsec configuration on EAST.	vyatta@EAST# show security vpn
	vpn {
	ipsec {
	esp-group ESP-1E {
	compression disable
	lifetime 3600
	mode tunnel
	pfs enable
	proposal 1 {
	encryption aes128
	hash sha1
	}
	}
	ike-group IKE-1E {
	lifetime 28800
	proposal 1 {
	encryption aes128
	hash sha1
	}
	}

	logging {
	log-modes all
	}
	nat-traversal disable
	site-to-site {
	peer 2001:db8:10::1 {
	authentication {
	mode pre-shared-secret
	pre-shared-secret test123
	}
	connection-type initiate
	default-esp-group ESP-1E
	ike-group IKE-1E
	local-address 2001:db8:10::2
	tunnel 1 {
	allow-nat-networks disable
	allow-public-networks disable
	local {
	prefix 2001:db8:2::/64
	}
	remote {
	prefix 2001:db8:1::/64
	}
	}
	}
	}
	}
	}
	[edit]

IPsec Site-to-Site VPN Commands

generate vpn rsa-key

Generates a pair of RSA public and private keys.

Syntax **generate vpn rsa-key** [**F4**] [**bits** *bits*]

Parameters *bits*

Bit-length of the generated key, in 16-bit increments. The length ranges from 1024 through 4096. The default length is 2192.

F4

When specified, sets the public exponent to 65537. When absent, sets the public exponent to 3.

Modes Operational mode

Usage Guidelines Use this command to generate a pair of RSA public and private keys. This command is available only to users with administrative privileges.

NOTE
A larger exponent makes brute-force attacks on public keys more difficult, so Brocade recommends using the F4 option.

RSA key pairs authenticate identities of hosts or users and securely exchange a random one-time key, which is then used for a session as the symmetrical encryption key. The public key or keys (more than one public key can be derived from the private key component) are shared with the peer that requests communication with the holder of the private key. Due to this potential one-to-many relationship, the private key is typically generated by and stored on the server, and the public key or keys are distributed to one or more clients.

The RSA key pair for the local host is generated by using this command in operational mode. After the key pair is generated, it is stored at the location that is specified by the **local-key rsa-key-name** option. By default, this location is the **localhost.key** file in the /config/ipsec.d/rsa-keys/ directory.

You can change the name and location of the key file by using security vpn rsa-keys on page 158.

Examples The following example shows how to extract the public key in an exportable form. The public key can be extracted in the format that is used in RFC-2537, RSA/MD5 KEYs and SIGs in the Domain Name System (DNS), as the credentials of a peer by extracting it from the **localhost.key** file. You can then paste it into the appropriate configuration parameter on the peer.

```
vyatta@WEST:~$ sed -n -e 's/^.*#pubkey=//p' /config/ipsec.d/rsa-keys/localhost.key
0sAQPEm9WaOOMSSxRYprYinUcalng5qiDaYdGUrHRgVWqLpi4jplpkgGdPJWHjsgzLtoIhMMvtvJ4QCzXdom29
0m8EyHcuaXfST+muZvsLyf06sRR0iM6xdqcNvMc4E4MY+NCHky+Y0MEg8SjKAldQs
+A2Nun2DQmzTNM6Slwe4VYXnQ==
#
```

The following example shows how to generate a pair of RSA public and private keys.

```
vyatta@WEST:~$ generate vpn rsa-key bits 1024
Generating rsa-key to /config/ipsec.d/rsa-keys/localhost.key
Your new local RSA key has been generated.
RSA key fingerprint: 78:af:08:60:92:34:c6:02:94:a2:52:53:69:91:a0:91
```

generate vpn x509 key-pair <name>

Generates an X.509 private key file and a certificate signing request file.

Syntax **generate vpn x509 key-pair** *name*

Parameters *name*

The name to be used for the X.509 private key file and certificate signing request file. The private key file will be called /config/auth/name.key and the certificate signing request file will be called /config/auth/name.csr.

Modes Operational mode

Usage Guidelines Use this command to generate an X.509 private key file and a certificate signing request file. The private key file is required for configuring a VPN for X.509 authentication (see security vpn ipsec site-to-site peer <peer> authentication x509 key file <file-name> on page 137). The certificate signing request file must be sent to a certificate authority (CA). In return, the CA will provide a server certificate (e.g. **name.crt**), a CA certificate (e.g. **ca.crt**), and potentially, a certificate revocation list (.crl) file. This procedure varies according to the CA being used. The files returned are also used to configure a VPN for X.509 authentication (see security vpn ipsec site-to-site peer <peer> authentication x509 cert-file <file-name> on page 135 for specifying the server certificate, security vpn ipsec site-to-site peer <peer> authentication x509 ca-cert-file <file-name> on page 134 for specifying the CA certificate, and security vpn ipsec site-to-site peer <peer> authentication x509 crl-file <file-name> on page 136 for specifying the certificate revocation list).

reset vpn ipsec-peer <peer>

Resets tunnels associated with the IPsec peer.

Syntax **reset vpn ipsec-peer** *peer* [**tunnel** *tunnel* | **vti**]

Parameters *peer*

The IPv4 or IPv6 address of the VPN peer.

tunnel

The tunnel to be reset. The numbers range from 0 through 4294967295.

vti

Reset the virtual tunnel interface associated with the peer.

Modes Operational mode

Usage Guidelines Use this command to reset IPsec tunnels associated with the specified peer. Resetting IPsec tunnels will cause the tunnels to be torn down and re-established.

If the peer is 0.0.0.0, “any”, or @id, then the tunnel is torn down and re-loaded but a new connection is not initiated because the remote end could be multiple end-points.

If tunnel or vti is not specified then all IPsec connections associated with the peer will be restarted.

restart vpn

Restarts the IPsec process.

Syntax **restart vpn**

Modes Operational mode

Usage Guidelines Use this command to restart the IPsec process.

Restarting IPsec will cause all tunnels to be torn down and re-established.

Examples The following example shows the output resulting from the **restart vpn** command.

```
vyatta@WEST> restart vpn
Stopping Openswan IPsec...
Starting Openswan IPsec 2.4.6...
vyatta@WEST>
```

show vpn debug

Provides trace-level information about IPsec VPN.

Syntax **show vpn debug** [**detail** | **peer** *peer* [**tunnel** *tunnel*]]

Parameters *detail*

Provides extra verbose output at the trace level.

peer

Shows trace-level information for the specified VPN peer. The format is the IPv4 or IPv6 address of the peer.

tunnel

Shows trace-level information for the specified tunnel to the specified peer. The *tunnel* argument is an integer that uniquely identifies the tunnel to the specified peer. The numbers range from 0 through 4294967295.

Modes Operational mode

Usage Guidelines Use this command to view trace-level messages for IPsec VPN.

This command is useful for troubleshooting and diagnostic situations.

Examples The following example shows the output of the **show vpn debug** command.

```
vyatta@WEST> show vpn debug
000 Status of IKEv1 pluto daemon (strongSwan 4.3.2):
000 interface lo/lo ::1:500
000 interface lo/lo 127.0.0.1:500
000 interface dp0p1p1/dp0p1p1 172.16.117.128:500
000 interface dp0p1p3/dp0p1p3 172.16.139.128:500
000 %myid = (none)
000 loaded plugins: curl ldap random pubkey openssl hmac gmp
000 debug options: none
000
000 "peer-172.16.139.160-tunnel-1": 172.16.139.128...172.16.139.160; erouted; eroute
owner: #5
000 "peer-172.16.139.160-tunnel-1":   ike_life: 28800s; ipsec_life: 3600s;
rekey_margin: 540s; rekey_fuzz: 100%; keyingtries: 3
000 "peer-172.16.139.160-tunnel-1":   policy: PSK+ENCRYPT+TUNNEL+PFS+UP; prio: 32,32;
interface: dp0p1p3;
000 "peer-172.16.139.160-tunnel-1":   newest ISAKMP SA: #4; newest IPsec SA: #5;
000 "peer-172.16.139.160-tunnel-1":   IKE proposal: AES_CBC_128/HMAC_SHA1/MODP_1536
000 "peer-172.16.139.160-tunnel-1":   ESP proposal: AES_CBC_128/HMAC_SHA1/<Phase1>
000
000 #5: "peer-172.16.139.160-tunnel-1" STATE_QUICK_R2 (IPsec SA established);
EVENT_SA_REPLACE in 3292s; newest IPSEC; eroute owner
000 #5: "peer-172.16.139.160-tunnel-1" esp.c75a2bd9@172.16.139.160 (0 bytes)
esp.d1c08d06@172.16.139.128 (0 bytes); tunnel
000 #4: "peer-172.16.139.160-tunnel-1" STATE_MAIN_R3 (sent MR3, ISAKMP SA
established); EVENT_SA_REPLACE in 28491s; newest ISAKMP
--More--
```

The following example shows the output of the **show vpn debug detail** command.

```
vyatta@WEST> show vpn debug detail
Unable to find IKEv2 messages. Strongswan might be running with IKEv2 turned off or
alternatively, your log files have been emptied (ie, logwatch)
vDUT-1
Wed Jan 20 23:22:27 GMT 2010
+ _____________________________ version
+ ipsec --version
Linux strongSwan U4.3.2/K2.6.31-1-586-vyatta
Institute for Internet Technologies and Applications
University of Applied Sciences Rapperswil, Switzerland
See 'ipsec --copyright' for copyright information.
+ _____________________________ /proc/net/pfkey
+ test -r /proc/net/pfkey
+ cat /proc/net/pfkey
sk       RefCnt Rmem   Wmem   User   Inode
+ _____________________________ ip-xfrm-state
+ ip -s xfrm state
src 172.16.139.128 dst 172.16.139.160
   proto esp spi 0xc75a2bd9(3344575449) reqid 16385(0x00004001) mode tunnel
   replay-window 32 seq 0x00000000 flag  (0x00000000)
   auth hmac(sha1) 0x7cd0c727850b972ef14ad983e4067833ac9e9b74 (160 bits)
   enc cbc(aes) 0x492215c8e674a858e887d23b05ec8fb1 (128 bits)
   sel src 0.0.0.0/0 dst 0.0.0.0/0 uid 0
   lifetime config:
     limit: soft (INF)(bytes), hard (INF)(bytes)
     limit: soft (INF)(packets), hard (INF)(packets)
     expire add: soft 0(sec), hard 0(sec)
     expire use: soft 0(sec), hard 0(sec)
   lifetime current:
     0(bytes), 0(packets)
     add 2010-01-20 22:44:56 use -
   stats:
     replay-window 0 replay 0 failed 0
--More--
```

show vpn ike rsa-keys

Displays RSA public keys recorded in the system.

Syntax **show vpn ike rsa-keys**

Modes Operational mode

Usage Guidelines Use this command to display the public portion of all RSA digital signatures recorded on the system.

This will include the public portion of the RSA digital signature of the local host (the private portion will not be displayed), plus the public key configured for any VPN peer.

Examples The following example shows output of the **show vpn ike rsa-keys** command, which displays the RSA digital signatures stored on router WEST. In this example:

- The public portion of the key for the local host is shown, but the private portion of the local key remains hidden in the RSA keys file.
- The RSA public key recorded for the VPN peer EAST is also shown.

```
vyatta@WEST> show vpn ike rsa-keys

Local public key
0sAQNfpZicOXWl1rMvNWLIfFppq1uWtUvj8esyjBl/zBfrK4ecZbt7WzMdMLiLugYtVgo+zJQV5dmQnN
+n3qkU9ZLM5QWBxG4iLFtYcwC5fCMx0hBJfnIEd68d11h7Ea6J4IAm3ZWXcBeOV4S8mC4HV
+mqZfv3xyh1ELjfmLM3fWkp8g5mX7ymgcTpneHiSYX1T9NU3i2CHjYfeKPFb4zJIopu2R654kODGOa
+4r241Zx3cDIJgHBYSYOiSFYbcdQhKQS3cclFPGVMHYGXjjoiUSA7d2eMabDtIU4FwnqH3qVN/
kdedK34sEJiMUgieT6pJQ6W8y+5PgESvouykx8cyTiOobnx0G9oqFcxYLknQ3GbrPej

================================================================
Peer IP: 10.1.0.55  (EAST)

0sAQOVBIJL+rIkpTuwh8FPeceAF0bhgLr++W51bOAIjFbRDbR8gX3Vlz6wiUbMgGwQxWlYQiqsCeacicsfZx/
amlEn9PkSE4e7tqK/JQo40L5C7gcNM24mup1d
+0WmN3zLb9Qhmq5q3pNJxEwnVbPPQeIdZMJxnb1+lA8DPC3SIxJM/3at1/KrwqCAhX3QNFY/
zNmOtFogELCeyl4+d54wQljA+3dwFAQ4bboJ7YIDs+rqORxWd3l3I7IajT/
pLrwr5eZ8OA9NtAedbMiCwxyuyUbznxXZ8Z/MAi3xjL1pjYyWjNNiOij82QJfMOrjoXVCfcPn96ZN+Jqk
+KknoVeNDwzpoahFOseJREeXzkw3/lkMN9N1
vyatta@WEST>
```

show vpn ike sa

Provides information about all currently active IKE (ISAKMP) security associations.

Syntax **show vpn ike sa** [**nat-traversal** | **peer** *peer*]

Parameters **nat-traversal**

Displays all the IKE SAs that are using RFC 3947 NAT Traversal.

peer

Shows IKE SA information for the specified VPN peer. The format is the IPv4 or IPv6 address of the peer.

There will be at most one IKE SA per peer (except possibly during re-key negotiation).

Modes Operational mode

Usage Guidelines Use this command to display information about IKE security associations (SAs).

Examples The following example shows the output of the **show vpn ike sa** command.

```
vyatta@WEST> show vpn ike sa

Peer ID / IP                            Local ID / IP
------------                            -------------
192.168.1.1                             192.168.1.2

    Description: site-to-site x509 tunnel

    State  Encrypt  Hash  D-H Grp  NAT-T  A-Time  L-Time
    -----  -------  ----  -------  -----  ------  ------
    up     aes128   sha1  5        no     2162    28800

vyatta@WEST>
```

show vpn ike secrets

Displays configured pre-shared secrets.

Syntax **show vpn ike secrets**

Modes Operational mode

Usage Guidelines Use this command to display information about pre-shared secrets recorded in the system.

Examples The following example shows the output of the **show vpn ike secrets** command.

```
vyatta@WEST> show vpn ike secrets
Local IP/ID                             Peer IP/ID
-----------                             -----------
192.168.1.2                             1.1.1.2
N/A                                     192.168.2.2
    Secret: "secret"
Local IP/ID                             Peer IP/ID
-----------                             -----------
192.168.1.2                             192.168.2.2
N/A                                     192.168.2.2
    Secret: "secret"
```

show vpn ike status

Displays summary information about the IKE process.

Syntax **show vpn ike status**

Modes Operational mode

Usage Guidelines Use this command to see the status of the IKE process.

Examples The following example shows the output of the **show vpn ike status** command.

```
vyatta@west> show vpn ike status
IKE Process Running

PID: 5832

vyatta@west>
```

show vpn ipsec sa

Provides information about active IPsec security associations.

Syntax **show vpn ipsec sa** [**peer** *peer* [**tunnel** *tunnel*]]

Parameters *peer*

Shows active IPsec security associations for the specified VPN peer. The format is the IPv4 or IPv6 address of the peer.

tunnel

Shows active IPsec security associations for the specified tunnel to the specified peer. The *tunnel* argument is an integer that uniquely identifies the tunnel to the specified peer. The numbers range from 0 through 4294967295.

Modes Operational mode

Usage Guidelines Use this command to display information about remote VPN peers and IPsec security associations (SAs) currently in effect.

Examples The following example shows the output of the **show vpn ipsec sa** command.

```
vyatta@WEST> show vpn ipsec sa
Peer ID / IP                            Local ID / IP
------------                            -------------
1.1.1.2                                 192.168.1.2

    Tunnel  State  Bytes Out/In   Encrypt  Hash  NAT-T  A-Time  L-Time  Proto
    ------  -----  ------------   -------  ----  -----  ------  ------  -----
    1       up     0.0/0.0        aes128   sha1  yes    3415    3600    GRE
    2       down   n/a            n/a      n/a   yes    0       3600    all


Peer ID / IP                            Local ID / IP
------------                            -------------
192.168.2.2                             192.168.1.2

    Tunnel  State  Bytes Out/In   Encrypt  Hash  NAT-T  A-Time  L-Time  Proto
    ------  -----  ------------   -------  ----  -----  ------  ------  -----
    1       down   n/a            n/a      n/a   no     0       3600    GRE
vyatta@WEST>
```

The following example shows the output of the **show vpn ipsec sa peer** command.

```
vyatta@WEST> show vpn ipsec sa peer 1.1.1.2
Peer ID / IP                            Local ID / IP
------------                            -------------
1.1.1.2                                 192.168.1.2
    Tunnel  State  Bytes Out/In   Encrypt  Hash  NAT-T  A-Time  L-Time  Proto
    ------  -----  ------------   -------  ----  -----  ------  ------  -----
    1       up     0.0/0.0        aes128   sha1  yes    3415    3600    GRE
    2       down   n/a            n/a      n/a   yes    0       3600    all
vyatta@WEST>
```

The following example shows the output of the **show vpn ipsec sa peer tunnel** command.

```
vyatta@WEST> show vpn ipsec sa peer 1.1.1.2 tunnel 1
Peer ID / IP                            Local ID / IP
------------                            -------------
1.1.1.2                                 192.168.1.2
    Tunnel  State  Bytes Out/In   Encrypt  Hash  NAT-T  A-Time  L-Time  Proto
    ------  -----  ------------   -------  ----  -----  ------  ------  -----
    1       up     0.0/0.0        aes128   sha1  yes    3415    3600    GRE
vyatta@WEST>
```

show vpn ipsec sa detail

Provides detailed information about active IPsec security associations.

Syntax **show vpn ipsec sa detail** [**peer** *peer* [**tunnel** *tunnel*]]

Parameters *peer*

The peer to display information about.

tunnel

The tunnel to display information about. The number ranges from 0 through 4294967295.

Modes Operational mode

Usage Guidelines Use this command to display detailed information about remote VPN peers and IPsec security associations (SAs) currently in effect.

Examples The following example shows the output of the **show vpn ipsec sa detail** command.

```
vyatta@WEST> show vpn ipsec sa detail
----------------------------------------------------------------------
Peer IP:                192.168.1.1
Peer ID:                r2
Local IP:               192.168.1.2
Local ID:               r1
NAT Traversal:          no
NAT Source Port:        n/a
NAT Dest Port:          n/a
Description: site-to-site x509 tunnel
    Tunnel 1:
        State:                  up
        Inbound SPI:            714f7f33
        Outbound SPI:           8a84d58
        Encryption:             aes128
        Hash:                   sha1
        PFS Group:              5
        CA:
            C=US
            ST=CA
            L=BELMONT
            O=Organization
            CN=CertAuth
            E=test@example.com
        Local Net:              172.16.0.0/24
        Local Protocol:         all
        Local Port:             all
        Remote Net:             172.16.1.0/24
        Remote Protocol:        all
        Remote Port:            all
        Inbound Bytes:          0.0
        Outbound Bytes:         0.0
        Active Time (s):        1876
        Lifetime (s):           3600
vyatta@WEST>
```

The following example shows the output of the **show vpn ipsec sa detail peer** *peer* command for an x509 tunnel (note the “CA” information).

```
vyatta@WEST> show vpn ipsec sa detail peer 192.168.1.1
----------------------------------------------------------------------
Peer IP:                192.168.1.1
Peer ID:                r2
Local IP:               192.168.1.2
Local ID:               r1
NAT Traversal:          no
NAT Source Port:        n/a
NAT Dest Port:          n/a
Description: site-to-site x509 tunnel
    Tunnel 1:
        State:                  up
        Inbound SPI:            714f7f33
        Outbound SPI:           8a84d58
        Encryption:             aes128
        Hash:                   sha1
        PFS Group:              5
        CA:
            C=US
            ST=CA
            L=BELMONT
            O=Organization
            CN=CertAuth
            E=test@example.com
        Local Net:              172.16.0.0/24
        Local Protocol:         all
        Local Port:             all
        Remote Net:             172.16.1.0/24
        Remote Protocol:        all
        Remote Port:            all
        Inbound Bytes:          0.0
        Outbound Bytes:         0.0
        Active Time (s):        1876
        Lifetime (s):           3600
vyatta@WEST>
```

show vpn ipsec sa nat-traversal

Provides information about all active IPsec security associations that are using NAT Traversal.

Syntax **show vpn ipsec sa nat-traversal**

Modes Operational mode

Usage Guidelines Use this command to display information about all active IPsec security associations that are using RFC 3947 NAT Traversal.

show vpn ipsec sa statistics

Display statistics information about active IPsec security associations.

Syntax **show vpn ipsec sa statistics** [**peer** *peer* [**tunnel** *tunnel*]]

Parameters *peer*

The peer to display information about.

tunnel

The tunnel to display information about. The number ranges from 0 through 4294967295.

Modes Operational mode

Usage Guidelines Use this command to see statistics for active IPsec security associations.

Examples The following example shows the output of the **show vpn ipsec sa statistics** command.

```
vyatta@WEST> show vpn ipsec sa statistics

Peer ID / IP                            Local ID / IP
------------                            -------------
1.1.1.2                                 192.168.1.2


  Tun# Dir Source Network               Destination Network          Bytes
  ---- --- --------------               -------------------          -----
  1    in  192.168.2.2/32               192.168.1.2/32               0.0
  1    out 192.168.1.2/32               192.168.2.2/32               0.0
  2    in  n/a                          n/a                          0.0
  2    out n/a                          n/a                          0.0


Peer ID / IP                            Local ID / IP
------------                            -------------
192.168.2.2                             192.168.1.2

  Tun# Dir Source Network               Destination Network          Bytes
  ---- --- --------------               -------------------          -----
  1    in  n/a                          n/a                          0.0
  1    out n/a                          n/a                          0.0

vyatta@WEST>
```

show vpn ipsec status

Displays information about the status of IPsec processes.

Syntax **show vpn ipsec status**

Modes Operational mode

Usage Guidelines Use this command to display information about the status about running IPsec processes.

The information shown includes:

- The process ID
- The number of active tunnels
- The interfaces configured for IPsec
- The IP addresses of interfaces configured for IPsec

Examples The following example shows the output of the **show vpn ipsec status** command.

```
vyatta@WEST> show vpn ipsec status
IPSec Process Running  PID: 5832

4 Active IPsec Tunnels

IPsec Interfaces:
  dp0p1p2 (10.6.0.55)


vyatta@WEST>
```

security vpn ipsec

Enables IPsec VPN functionality on the system.

Syntax **set security vpn ipsec**

delete security vpn ipsec

show security vpn ipsec

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec
    }
}
```

Usage Guidelines Use this command to enable IPsec VPN functionality on the Brocade vRouter.

NOTE
The sending and receiving of ICMP redirects is disabled when IPsec VPN is configured.

Use the **set** form of this command to enable IPsec VPN.

Use the **delete** form of this command to remove all IPsec VPN configuration and disable IPsec VPN functionality.

Use the **show** form of this command to view the IPsec VPN configuration.

security vpn ipsec auto-update <interval>

Specifies the interval to automatically refresh IPsec connections.

Syntax **set security vpn ipsec auto-update** *interval*

delete security vpn ipsec auto-update

show security vpn ipsec auto-update

Command Default IPsec connections are not refreshed periodically.

Parameters *interval*

The interval (seconds) in which to review IPsec connections for changes (for example, the IP address of a dynamic DNS peer changes) and restart them if changes are found. The number ranges from 30 through 65535.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                auto-update interval
        }
    }
}
```

Usage Guidelines Use this command to specify the interval to automatically refresh IPsec connections. This is most useful for connections where the remote peer uses dynamic DNS to keep track of its address. Auto-update will review information pertaining to the connection at the specified interval and, if it is changed (for example, if the dynamic DNS peer's IP address has changed), will restart the connection.

Use the **set** form of this command to specify the interval at which to automatically refresh IPsec connections.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec esp-group <name>

Defines a named ESP configuration for IKE Phase 2 negotiations.

Syntax **set security vpn ipsec esp-group** *name*

delete security vpn ipsec esp-group

show security vpn ipsec esp-group

Parameters *name*

Multi-node. The name to be used to refer to the ESP configuration.

You can create multiple ESP configurations by creating multiple **esp-group** configuration nodes. At least one ESP configuration must be defined, for use in tunnel configuration.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name
            }
    }
}
```

Usage Guidelines Use this command to define an ESP group.

An ESP group lets you set the Encapsulating Security Payload (ESP) parameters required for IKE Phase 2 and the lifetime of the resulting IPsec security association.

Use the **set** form of this command to create and modify an ESP group.

Use the **delete** form of this command to remove ESP group configuration.

Use the **show** form of this command to view ESP group configuration.

security vpn ipsec esp-group <name> compression <state>

Specifies whether this VPN gateway should propose the use of compression.

Syntax

set security vpn ipsec esp-group *name* **compression** *state*

delete security vpn ipsec esp-group *name* **compression**

show security vpn ipsec esp-group *name* **compression**

Command Default

ESP compression is disabled.

Parameters

name

The name to be used to refer to the ESP configuration.

state

Enables or disables proposal of ESP compression. Supported values are as follows:

enable—Enables proposal of ESP compression.

disable—Disables proposal ESP compression.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                    compression state
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify whether or not to propose ESP compression during IKE Phase 2 negotiation.

NOTE
Regardless of this setting, if the other gateway proposes compression, this gateway will comply.

Use the **set** form of this command to specify whether or not to enable ESP compression.

Use the **delete** form of this command to restore the default behavior.

Use the **show** form of this command to view ESP compression configuration.

security vpn ipsec esp-group <name> lifetime <lifetime>

Specifies how long an ESP encryption key can stay in effect.

Syntax

set security vpn ipsec esp-group *name* **lifetime** *lifetime*

delete security vpn ipsec esp-group *name* **lifetime**

show security vpn ipsec esp-group *name* **lifetime**

Command Default

Keys stay in effect for 3,600 seconds (1 hour).

Parameters

name

The name to be used to refer to the ESP configuration.

lifetime

The time, in seconds, that any key created during IKE Phase 2 negotiation can persist before the next negotiation is triggered. The numbers range from 30 through 86400 (that is, 24 hours). The default is 3600 (1 hour).

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                    lifetime lifetime
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify the lifetime of a key.

Use the **set** form of this command to specify the lifetime of a key.

Use the **delete** form of this command to remove the lifetime configuration.

Use the **show** form of this command to view the lifetime configuration.

security vpn ipsec esp-group <name> mode <mode>

Specifies the IPsec connection mode to be used.

Syntax

set security vpn ipsec esp-group *name* **mode** *mode*

delete security vpn ipsec esp-group *name* **mode**

show security vpn ipsec esp-group *name* **mode**

Command Default

IPsec connections use tunnel mode.

Parameters

name

The name to be used to refer to the ESP configuration.

mode

The IPsec connection mode. Supported values are as follows:

tunnel—Tunnel mode.

transport—Transport mode.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                        mode mode
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify the IPsec connection mode to be used.

Use the **set** form of this command to specify the IPsec connection mode to be used.

Use the **delete** form of this command to restore the default IPsec connection mode.

Use the **show** form of this command to view IPsec connection mode configuration.

security vpn ipsec esp-group <name> pfs <pfs>

Specifies whether or not PFS is used.

Syntax **set security vpn ipsec esp-group** *name* **pfs** *pfs*

delete security vpn ipsec esp-group *name* **pfs**

show security vpn ipsec esp-group *name* **pfs**

Command Default Perfect Forward Secrecy is enabled and uses the Diffie-Hellman group defined in the ike-group.

Parameters *name*

The name to be used to refer to the ESP configuration.

pfs

Enables or disables Perfect Forward Secrecy. Supported values are as follows:

enable—Enables Perfect Forward Secrecy using Diffie-Hellman group defined in the ike-group.

dh-group2—Enables Perfect Forward Secrecy using Diffie-Hellman group 2.

dh-group5—Enables Perfect Forward Secrecy using Diffie-Hellman group 5.

disable—Disables Perfect Forward Secrecy.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                            pfs pfs
                }
            }
    }
}
```

Usage Guidelines Use this command to specify whether or not Perfect Forward Secrecy (PFS) will be used and, if used, which Diffie-Hellman group is to be used.

NOTE
Regardless of the setting of this parameter, if the far-end VPN peer requests PFS, the Brocade vRouter will use PFS.

Use the **set** form of this command to specify whether or not Perfect Forward Secrecy (PFS) will be used.

Use the **delete** form of this command to restore default PFS configuration.

Use the **show** form of this command to view PFS configuration.

security vpn ipsec esp-group <name> proposal <num>

Defines an ESP group proposal for IKE Phase 2 negotiation.

Syntax

set security vpn ipsec esp-group *name* **proposal** *num*

delete security vpn ipsec esp-group proposal

show security vpn ipsec esp-group proposal

Parameters

name

The name to be used to refer to the ESP configuration.

num

Multi-node. An integer uniquely identifying a proposal to be used in IKE Phase 2 negotiation.

You can define multiple proposals within a single ESP configuration by creating multiple **proposal** configuration nodes. Each must have a unique identifier.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                            proposal num
                }
            }
    }
}
```

Usage Guidelines

Use this command to define an ESP proposal for IKE Phase 2 negotiation.

Use the **set** form of this command to create an ESP proposal.

Use the **delete** form of this command to remove an ESP proposal and all its configuration.

Use the **show** form of this command to view ESP proposal configuration.

security vpn ipsec esp-group <name> proposal <num> encryption <cipher>

Specifies the encryption cipher for an ESP proposal.

Syntax **set security vpn ipsec esp-group** *name* **proposal** *num* **encryption** *cipher*

delete security vpn ipsec esp-group proposal *num* **encryption**

show security vpn ipsec esp-group proposal *num* **encryption**

Command Default The default is **aes128**.

Parameters *name*

The name to be used to refer to the ESP configuration.

proposal

An integer uniquely identifying a proposal to be used in IKE Phase 2 negotiation.

cipher

The encryption cipher to be proposed. Supported values are as follows:

aes128—Advanced Encryption Standard with a 128-bit key.

aes256—Advanced Encryption Standard with a 256-bit key.

3des—Triple-DES (Data Encryption Standard).

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                        proposal num {
                        encryption cipher
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify the encryption cipher to be proposed in an ESP proposal during IKE Phase 2 negotiation.

Use the **set** form of this command to specify the encryption cipher.

Use the **delete** form of this command to restore default encryption configuration.

Use the **show** form of this command to view ESP proposal encryption configuration.

security vpn ipsec esp-group <name> proposal <num> hash <hash>

Specifies the hash algorithm for an ESP proposal.

Syntax

set security vpn ipsec esp-group *name* **proposal** *num* **hash** *hash*

delete security vpn ipsec esp-group proposal *num* **hash**

show security vpn ipsec esp-group proposal *num* **hash**

Command Default

The default is *sha1*.

Parameters

name

The name to be used to refer to the ESP configuration.

proposal

An integer uniquely identifying a proposal to be used in IKE Phase 2 negotiation.

hash

The hash algorithm to be used. Supported values are as follows:

sha1—The SHA-1 variant of the Secure Hash Algorithm.

md5—Version 5 of the message digest algorithm.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                esp-group name {
                        proposal num {
                            hash hash
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the hash algorithm to be proposed in an ESP proposal.

Use the **set** form of this command to specify the hash algorithm to be proposed.

Use the **delete** form of this command to restore default hash algorithm configuration.

Use the **show** form of this command to view ESP proposal hash algorithm configuration.

security vpn ipsec ike-group <name>

Defines a named IKE configuration for IKE Phase 1 negotiations.

Syntax

set security vpn ipsec ike-group *name*

delete security vpn ipsec ike-group

show security vpn ipsec ike-group

Parameters

name

Mandatory. Multi-node. The name to be used to refer to this IKE configuration.

You can create multiple IKE configurations by creating multiple **ike-group** configuration nodes.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name
            }
    }
}
```

Usage Guidelines

Use this command to configure a set of values for IKE configuration.

This configuration can be referred to as part of configuring a site-to-site configuration with a VPN peer, using security vpn ipsec profile <profile-name> authentication mode <mode> on page 123.

Use the **set** form of this command to create an IKE group.

Use the **delete** form of this command to remove an IKE group and all its configuration.

Use the **show** form of this command to view IKE group configuration.

security vpn ipsec ike-group <name> dead-peer-detection

Defines the behavior if the VPN peer becomes unreachable.

Syntax

set security vpn ipsec ike-group *name* **dead-peer-detection** [**action** *action* | **interval** *interval* | **timeout** *timeout*]

delete security vpn ipsec ike-group *name* **dead-peer-detection**

show security vpn ipsec ike-group *name* **dead-peer-detection**

Command Default

Dead peers are not detected.

Parameters

name

The name to be used to refer to this IKE configuration.

action

Specifies the action to be taken if the timeout interval expires. Supported values are as follows:

hold—Queue packets until the tunnel comes back up.

clear—Delete the connection information.

restart—Attempt to restart the tunnel.

interval

The interval, in seconds, at which IKE keep-alive messages will be sent to VPN peers. The numbers range from 15 through 86400. The default is 30.

timeout

The interval, in seconds, after which if the peer has not responded the defined action will be taken. The numbers range from 30 through 86400. The default is 120.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                    dead-peer-detection {
                        action action
                        interval interval
                        timeout timeout
                    }
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify how the system should detect dead IPsec VPN peers.

Use the **set** form of this command to configure dead peer detection.

Use the **delete** form of this command to remove dead peer detection configuration.

Use the **show** form of this command to view dead peer detection configuration.

security vpn ipsec ike-group <name> lifetime <lifetime>

Specifies how long an IKE group key can stay in effect.

Syntax **set security vpn ipsec ike-group** *name* **lifetime** *lifetime*

delete security vpn ipsec ike-group *name* **lifetime**

show security vpn ipsec ike-group *name* **lifetime**

Command Default An IKE key stays in effect for 8 hours.

Parameters *name*

The name to be used to refer to this IKE configuration.

lifetime

The time, in seconds, that any key created during IKE Phase 1 negotiation can persist before the next negotiation is triggered. The numbers range from 30 through 86400 (that is, 24 hours). The default is 28800 (8 hours).

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                    lifetime lifetime
                }
            }
    }
}
```

Usage Guidelines Use this command to specify the lifetime of an IKE key.

Use the **set** form of this command to specify key lifetime.

Use the **delete** form of this command to restore the default key lifetime.

Use the **show** form of this command to view key lifetime configuration.

security vpn ipsec ike-group <name> proposal <num>

Specifies the IKE group proposal number.

Syntax

set security vpn ipsec ike-group *name* **proposal** *num*

delete security vpn ipsec ike-group proposal

show security vpn ipsec ike-group proposal

Parameters

name

The name to be used to refer to the IKE configuration.

proposal

Multi-node. An integer uniquely identifying an IKE proposal.

You can define up to 10 proposals within a single IKE configuration by creating multiple **proposal** configuration nodes. Each proposal must have a unique identifier.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                        proposal num {
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to create an IKE proposal. The proposal will be used in IKE Phase 1 negotiation.

Use the **set** form of this command to create an IKE proposal.

Use the **delete** form of this command to remove an IKE proposal and all its configuration.

Use the **show** form of this command to view IKE proposal configuration.

security vpn ipsec ike-group <name> proposal <num> dh-group <group>

Specifies the Oakley group to be proposed for Diffie-Hellman key exchanges.

Syntax

set security vpn ipsec ike-group *name* **proposal** *num* **dh-group** *group*

delete security vpn ipsec ike-group proposal *num* **dh-group**

show security vpn ipsec ike-group proposal *num* **dh-group**

Parameters

name

The name to be used to refer to the IKE configuration.

proposal

An integer uniquely identifying an IKE proposal.

group

The Oakley group to be used in Diffie-Hellman key exchanges. Supported values are as follows:

2—Oakley group 2.

5—Oakley group 5.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                        proposal num {
                            dh-group group
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the Oakley group to be proposed for Diffie-Hellman key exchanges.

Use the **set** form of this command to specify the Oakley group.

Use the **delete** form of this command to remove Oakley group configuration.

Use the **show** form of this command to view Oakley group configuration.

security vpn ipsec ike-group <name> proposal <num> encryption <cipher>

Specifies the encryption cipher to be proposed in IKE Phase 1 negotiation.

Syntax

set security vpn ipsec ike-group *name* **proposal** *num* **encryption** *cipher*

delete security vpn ipsec ike-group proposal *num* **encryption**

show security vpn ipsec ike-group proposal *num* **encryption**

Command Default

The default is **aes128**.

Parameters

name

The name to be used to refer to the IKE configuration.

proposal

An integer uniquely identifying an IKE proposal.

cipher

The encryption cipher to be used in IKE Phase 1 negotiation. Supported values are as follows:

aes128—Advanced Encryption Standard with a 128-bit key.

aes256—Advanced Encryption Standard with a 256-bit key.

3des—Triple-DES (Data Encryption Standard).

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                        proposal num {
                            encryption cipher
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the encryption cipher to be proposed in IKE Phase 1 negotiation.

Use the **set** form of this command to set the encryption cipher.

Use the **delete** form of this command to restore the default encryption cipher.

Use the **show** form of this command to view encryption cipher configuration.

security vpn ipsec ike-group <name> proposal <num> hash <hash>

Specifies the hash algorithm to be proposed.

Syntax

set security vpn ipsec ike-group *name* **proposal** *num* **hash** *hash*

delete security vpn ipsec ike-group proposal *num* **hash**

show security vpn ipsec ike-group proposal *num* **hash**

Command Default

The default is **sha1**.

Parameters

name

The name to be used to refer to the IKE configuration.

proposal

An integer uniquely identifying an IKE proposal.

hash

The hash algorithm to be used. Supported values are as follows:

sha1: The SHA-1 variant of the Secure Hash Algorithm.

md5: Version 5 of the message digest algorithm.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ike-group name {
                        proposal num {
                                hash hash
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the hash algorithm to be proposed in an IKE proposal.

Use the **set** form of this command to specify the hash algorithm to be proposed.

Use the **delete** form of this command to restore default hash algorithm configuration.

Use the **show** form of this command to view IKE proposal hash algorithm configuration.

security vpn ipsec ipsec-interfaces interface <if-name>

Enables IPsec VPN on an interface.

Syntax **set security vpn ipsec ipsec-interfaces interface** *if-name*

delete security vpn ipsec ipsec-interfaces interface *if-name*

show security vpn ipsec ipsec-interfaces interface

Parameters *if-name*

Multi-node. The name of a network interface to be used for IPsec VPN. The network interface must already be created and configured.

You can enable IPsec VPN on more than one interface by creating multiple **interface** configuration nodes.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                ipsec-interfaces {
                        interface if-name
                }
            }
    }
}
```

Usage Guidelines Use this command to configure IPsec on an interface.

Use the **set** form of this command to enable IPsec on an interface.

Use the **delete** form of this command to remove the IPsec interfaces configuration.

NOTE
If you delete an interface from IPsec configuration, IPsec connections referencing this tunnel will no longer operate. If you attempt to enable a connection referencing the IP address of a deleted interface, an error will result.

Use the **show** form of this command to view IPsec interfaces configuration.

security vpn ipsec logging

Specifies logging options for IPsec VPN.

Syntax

set security vpn ipsec logging [**log-modes** *mode*]

delete security vpn ipsec logging [**log-modes**]

show security vpn ipsec logging [**log-modes**]

Parameters

log-modes *mode*

Mandatory. Multi-node. The log mode to be used for IPsec log messages. Supported values are as follows:

all—Enables all logging options.

raw—Shows the raw bytes of messages.

crypt—Shows the encryption and decryption of messages.

parsing—Shows the structure of input messages.

emitting— Shows the structure of output messages.

control—Shows the decision-making process of the IKE daemon (Pluto).

private—Allows debugging output with private keys.

You can configure multiple log modes, by creating more than one **log-mode** configuration node.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                logging {
                        log-modes mode
                }
            }
    }
}
```

Usage Guidelines

Use this command to define logging options for IPsec VPN.

When this command is set, the system uses the Brocade vRouter's internal VPN logging daemon for IPsec log messages.

The IPsec process generates log messages during operation. You can direct the system to send IPsec log messages to syslog. The result will depend on how the system syslog is configured.

Keep in mind that in the current implementation, the main syslog file reports only messages of severity warning and above, regardless of the severity level configured. If you want to configure a different level of severity for log messages (for example, if you want to see debug messages during troubleshooting), you must configure syslog to send messages into a different file, which you define within syslog.

Configuring log modes is optional. When a log mode is not configured, IPsec log messages consist mostly of IPsec startup and shutdown messages. The log modes allow you to direct the system to inspect the IPsec packets and report the results.

Note that some log modes (for example, *all* and *control*) generate several log messages per packet. Using any of these options may severely degrade system performance.

VPN IPsec log messages use standard syslog levels of severity.

Use the **set** form of this command to specify logging modes for IPsec VPN.

Use the **delete** form of this command to remove the logging configuration.

Use the **show** form of this command to view the logging configuration.

security vpn ipsec nat-networks allowed-network <ipv4net>

Specifies the private network addresses that remote hosts behind a NAT device may use.

Syntax

set security vpn ipsec nat-networks allowed-network *ipv4net* [**exclude** *ipv4net-exclude*]

delete security vpn ipsec nat-networks allowed-network *ipv4net* [**exclude** *ipv4net-exclude*]

show security vpn ipsec nat-networks allowed-network [*ipv4net* [**exclude**]]

Parameters

ipv4net

Multi-node. An IPv4 network of private IP addresses that remote hosts behind a NAT device may use.

ipv4net-exclude

Multi-node. An IPv4 network to be excluded from the allowed network range. These are the RFC 1918 (“private”) IP addresses being used on the network internal to this VPN gateway.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                nat-networks {
                    allowed-network ipv4net {
                        exclude ipv4net-exclude
                    }
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify RFC 1918 private IP addresses for remote networks that may reside behind a NAT device.

Unlike public IP addresses, private IP addresses may be re-used between sites. That means that private IP address ranges behind a NAT device at the far end of the VPN connection may overlap or be coextensive with private IP addresses on the internal network behind this VPN gateway, causing routing problems. For this reason, you must specify the allowed private network addresses that reside behind a NAT device, excluding internal network addresses.

Table 48 lists the three blocks of the IP address space that the Internet Assigned Numbers Authority (IANA) has reserved for private internets.

TABLE 48 IP addresses reserved for private networks

Network	Prefix
10.0.0.0-10.255.255.255	10.0.0.0/8
172.16.0.0-172.31.255.255	172.16.0.0/12
192.168.0.0-192.168.255.255	192.168.0.0/16

Use the **set** form of this command to specify the private network addresses that remote hosts behind a NAT device may use.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec nat-traversal <state>

Specifies whether the local VPN gateway proposes NAT Traversal capability.

Syntax **set security vpn ipsec nat-traversal** *state*

delete security vpn ipsec nat-traversal

show security vpn ipsec nat-traversal

Parameters *state*

Enables or disables RFC 3947 NAT Traversal. Supported values are as follows:

enable—Enables NAT Traversal.

disable—Disables NAT Traversal.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                nat-traversal state
        }
    }
}
```

Usage Guidelines Use this command to direct the Brocade vRouter to propose RFC 3947 NAT Traversal support during IKE negotiation.

Regardless of the setting of this parameter, if the far-end VPN peer requests NAT Traversal, the Brocade vRouter will use NAT Traversal.

Use the **set** form of this command to specify whether the system proposes NAT traversal capability.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec profile <profile-name>

Defines an IPsec profile.

Syntax **set security vpn ipsec profile** *profile-name*

delete security vpn ipsec profile *profile-name*

show security vpn ipsec profile *profile-name*

Parameters *profile-name*

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name
            }
    }
}
```

Usage Guidelines Use this command to define an IPsec configuration profile to associate with a pre-defined tunnel interface.

Use the **set** form of this command to define an IPsec configuration profile.

Use the **delete** form of this command to remove the profile configuration.

Use the **show** form of this command to view the profile configuration.

security vpn ipsec profile <profile-name> authentication mode <mode>

Defines an IPsec profile authentication mode.

Syntax

set security vpn ipsec profile *profile-name* **authentication mode** *mode*

delete security vpn ipsec profile *profile-name* **authentication mode**

show security vpn ipsec profile *profile-name* **authentication mode**

Parameters

profile-name

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

mode

The authentication method to be used for this profile.

Supported values are as follows:

pre-shared-secret—Uses a pre-shared secret for authentication.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name {
                    authentication {
                        mode mode
                    }
                }
            }
        }
}
```

Usage Guidelines

Use this command to specify the authentication method to use for an IPsec configuration profile.

Use the **set** form of this command to specify the authentication method to use for an IPsec configuration profile.

Use the **delete** form of this command to remove the authentication mode configuration.

Use the **show** form of this command to view the authentication mode configuration.

security vpn ipsec profile <profile-name> authentication pre-shared-secret <secret>

Specifies the pre-shared secret used to authenticate the VPN peer.

Syntax

set security vpn ipsec profile *profile-name* **authentication pre-shared-secret** *secret*

delete security vpn ipsec profile *profile-name* **authentication pre-shared-secret**

show security vpn ipsec profile *profile-name* **authentication pre-shared-secret**

Parameters

profile-name

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

secret

The pre-shared secret used to authenticate the VPN peer.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name {
                    authentication {
                        pre-shared-secret secret
                    }
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify the pre-shared secret used to authenticate the VPN peer.

Use the **set** form of this command to specify the pre-shared secret to use for an IPsec configuration profile.

Use the **delete** form of this command to remove the pre-shared secret configuration.

Use the **show** form of this command to view the pre-shared secret configuration.

security vpn ipsec profile <profile-name> bind tunnel <tunx>

Specifies the tunnel interface to associate the IPsec profile configuration with.

Syntax

set security vpn ipsec profile *profile-name* **bind tunnel** *tunx*

delete security vpn ipsec profile *profile-name* **bind tunnel**

show security vpn ipsec profile *profile-name* **bind tunnel**

Parameters

profile-name

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

tunx

The name of the tunnel interface to associate the IPsec profile configuration with.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name {
                        bind {
                            tunnel tunx
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the tunnel interface to associate the IPsec profile configuration with.

Use the **set** form of this command to specify the tunnel interface to associate the IPsec profile configuration with.

Use the **delete** form of this command to remove the bind configuration.

Use the **show** form of this command to view the bind configuration.

security vpn ipsec profile <profile-name> esp-group <name>

Specifies the ESP group to use for the IPsec profile configuration.

Syntax

set security vpn ipsec profile *profile-name* **esp-group** *name*

delete security vpn ipsec profile *profile-name* **esp-group**

show security vpn ipsec profile *profile-name* **esp-group**

Parameters

profile-name

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

name

The name of the ESP group to be used for the IPsec profile configuration. The ESP group must have already been defined using security vpn ipsec esp-group <name> on page 101.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name {
                        esp-group name
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify the ESP group to use for the IPsec profile configuration.

Use the **set** form of this command to specify the ESP group to use for the IPsec profile configuration.

Use the **delete** form of this command to remove the ESP group configuration.

Use the **show** form of this command to view the ESP group configuration.

security vpn ipsec profile <profile-name> ike-group <name>

Specifies the IKE group to use for the IPsec profile configuration.

Syntax

set security vpn ipsec profile *profile-name* **ike-group** *name*

delete security vpn ipsec profile *profile-name* **ike-group**

show security vpn ipsec profile *profile-name* **ike-group**

Parameters

profile-name

Multi-node. The name of the IPsec configuration profile.

You can define more than one IPsec profile by creating multiple **profile** configuration nodes.

name

The name of the IKE group to be used for the IPsec profile configuration. The IKE group must have already been defined using security vpn ipsec ike-group <name> on page 109.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                profile profile-name {
                ike-group name
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the IKE group to use for the IPsec profile configuration.

Use the **set** form of this command to specify the IKE group to use for the IPsec profile configuration.

Use the **delete** form of this command to remove the IKE group configuration.

Use the **show** form of this command to view the IKE group configuration.

security vpn ipsec site-to-site peer <peer>

Defines a site-to-site connection between the Brocade vRouter and another VPN gateway.

Syntax

set security vpn ipsec site-to-site peer *peer*

delete security vpn ipsec site-to-site peer *peer*

show security vpn ipsec site-to-site peer *peer*

Parameters

peer

Multi-node. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

You can define more than one VPN peer by creating multiple **peer** configuration nodes.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer
                }
            }
    }
}
```

Usage Guidelines

Use this command to define a site-to-site connection with another VPN peer.

For peers that have a known IP address or hostname, specify the IP address or hostname (IPv4 networks only) of the peer. For those that have a known authentication ID (prefixed with "@") specify the authentication ID of the peer. For peers where the IP address is unknown—for example, in the scenario where there are multiple "road warrior" peers—specify 0.0.0.0 as the peer, meaning there are multiple possible peers.

Use the **set** form of this command to define a site-to-site connection with another VPN peer.

Use the **delete** form of this command to remove the peer configuration.

Use the **show** form of this command to view the peer configuration.

security vpn ipsec site-to-site peer <peer> authentication id <id>

Specifies local authentication credentials to send to the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication id** *id*

delete security vpn ipsec site-to-site peer *peer* **authentication id**

show security vpn ipsec site-to-site peer *peer* **authentication id**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

id

The local authentication credentials to send to the VPN peer. Can be specified if the **local-address** address for the peer is set to **any** (which means the external address of the interface is dynamic); ignored otherwise. Use the format @ *id* to specify the *id*.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                            id id
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the local authentication credentials to send to the VPN peer.

Use the **set** form of this command to specify the local authentication credentials to send to the VPN peer.

Use the **delete** form of this command to remove the local authentication credentials.

Use the **show** form of this command to view the local authentication credentials.

security vpn ipsec site-to-site peer <peer> authentication mode <mode>

Specifies the authentication method to be used for the connection with the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication mode** *mode*

delete security vpn ipsec site-to-site peer *peer* **authentication mode**

show security vpn ipsec site-to-site peer *peer* **authentication mode**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

mode

Specifies the authentication method to be used for this connection. Supported values are as follows:

pre-shared-secret—Uses a pre-shared secret for authentication.

rsa—Uses an RSA digital signature for authentication.

x509—Uses X.509 V.3 certificates for authentication.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                        mode mode
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the authentication method to be used for the connection to the VPN peer.

Use the **set** form of this command to specify the authentication method to be used for the connection to the VPN peer.

Use the **delete** form of this command to remove the authentication method configuration.

Use the **show** form of this command to view the authentication method configuration.

security vpn ipsec site-to-site peer <peer> authentication pre-shared-secret <secret>

Specifies the pre-shared secret used to authenticate the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication pre-shared-secret** *secret*

delete security vpn ipsec site-to-site peer *peer* **authentication pre-shared-secret**

show security vpn ipsec site-to-site peer *peer* **authentication pre-shared-secret**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

secret

Specifies the pre-shared secret to be used to authenticate the VPN peer.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        authentication {
                            pre-shared-secret secret
                        }
                    }
                }
            }
        }
}
```

Usage Guidelines

Use this command to specify the pre-shared secret used to authenticate the VPN peer. The pre-shared-secret set here is only valid if the authentication mode is set to pre-shared-secret.

Use the **set** form of this command to specify the pre-shared secret used to authenticate the VPN peer.

Use the **delete** form of this command to remove the pre-shared secret configuration.

Use the **show** form of this command to view the pre-shared secret configuration.

security vpn ipsec site-to-site peer <peer> authentication remote-id <id>

Specifies the authentication credentials of the VPN peer.

Syntax **set security vpn ipsec site-to-site peer** *peer* **authentication remote-id** *id*

delete security vpn ipsec site-to-site peer *peer* **authentication remote-id**

show security vpn ipsec site-to-site peer *peer* **authentication remote-id**

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

id

The authentication credentials of the remote VPN peer. The id can be an IP address, a hostname (IPv4 networks only), an authentication ID in the form @ *id* , or, for X.509, a string specifying the “distinguished name” of the certificate for the remote end of the tunnel.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                        remote-id id
                    }
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify the authentication credentials of the VPN peer. The **remote-id** is an override to the default authentication - the peer IP address. The remote peer uses an authentication ID for authentication when its IP address is dynamic or it identifies itself with a different IP address or hostname (IPv4 networks only). An example of this is when the remote peer is behind a NAT device.

Another case where *remote-id* is required is for X.509 authentication. In this case, a string specifying the “distinguished name” of the certificate for the remote end of the tunnel is used. For example, the string “C=US, ST=CA, O=ABC Company, CN=test, E=root@abcco.com” specifies the information included in the X.509 certificate for the peer.

Use the **set** form of this command to specify the authentication credentials of the VPN peer.

Use the **delete** form of this command to remove the remote peer authentication credentials.

Use the **show** form of this command to view the remote peer authentication credentials.

security vpn ipsec site-to-site peer <peer> authentication rsa-key-name <name>

Specifies the name of the digital signature used to authenticate the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication rsa-key-name** *name*

delete security vpn ipsec site-to-site peer *peer* **authentication rsa-key-name**

show security vpn ipsec site-to-site peer *peer* **authentication rsa-key-name**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

name

The name of the digital signature used to authenticate the VPN peer.

To record an RSA digital signature for a VPN peer, use the **set** form of security vpn rsa-keys on page 158.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        authentication {
                            rsa-key-name name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the name of the digital signature to use to authenticate the VPN peer. The **rsa-key-name** set here is only valid if the **authentication mode** is set to **rsa**.

Use the **set** form of this command to specify the name of the digital signature to use to authenticate the VPN peer.

Use the **delete** form of this command to remove the name of the digital signature.

Use the **show** form of this command to view the name of the digital signature.

security vpn ipsec site-to-site peer <peer> authentication x509 ca-cert-file <file-name>

Specifies the name of an X.509 Certificate Authority (CA) certificate file for IPsec authentication of the VPN peer.

Syntax **set security vpn ipsec site-to-site peer** *peer* **authentication x509 ca-cert-file** *file-name*

delete security vpn ipsec site-to-site peer *peer* **authentication x509 ca-cert-file**

show security vpn ipsec site-to-site peer *peer* **authentication x509 ca-cert-file**

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

file-name

The certificate file name. This parameter is mandatory if **authentication mode** is **x509**.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                        x509 {
                            ca-cert-file file-name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify the name of an X.509 Certificate Authority (CA) certificate file. The X.509 CA certificate is used for IPsec authentication for the VPN peer.

Certificate and key files are assumed to be in /config/auth unless an absolute path is specified.

Use the **set** form of this command to specify the name of the CA certificate file.

Use the **delete** form of this command to remove the name of the CA certificate file.

Use the **show** form of this command to display CA certificate file configuration.

security vpn ipsec site-to-site peer <peer> authentication x509 cert-file <file-name>

Specifies the name of the VPN server's certificate file for IPsec authentication of the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication x509 cert-file** *file-name*

delete security vpn ipsec site-to-site peer *peer* **authentication x509 cert-file**

show security vpn ipsec site-to-site peer *peer* **authentication x509 cert-file**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

file-name

The name of the VPN server's certificate file. This parameter is mandatory if **authentication mode** is **x509**.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                        x509 {
                            cert-file file-name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the name to the VPN server's certificate file. The VPN server's certificate certifies the identity of the VPN server.

Certificate and key files are assumed to be in /config/auth unless an absolute path is specified.

Use the **set** form of this command to specify the name of the VPN server's certificate file.

Use the **delete** form of this command to remove the name of the VPN server's certificate file.

Use the **show** form of this command to display VPN server certificate file configuration.

security vpn ipsec site-to-site peer <peer> authentication x509 crl-file <file-name>

Specifies the name of an X.509 Certificate Revocation List (CRL) file for IPsec authentication of the VPN peer.

Syntax **set security vpn ipsec site-to-site peer** *peer* **authentication x509 crl-file** *file-name*

delete security vpn ipsec site-to-site peer *peer* **authentication x509 crl-file**

show security vpn ipsec site-to-site peer *peer* **authentication x509 crl-file**

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

file-name

The name of the CRL file.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    authentication {
                        x509 {
                            crl-file file-name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify the name of a Certificate Revocation List (CRL) file.

A CRL is a time-stamped signed data structure issued by the Certificate Authority (CA) identifying revoked certificates. When the remote user attempts to log on to the system, the system checks both the remote user's certificate signature and also the CRL to make sure that the remote user's certificate serial number is not on the CRL. If it is, the login attempt will be refused.

The file is assumed to be in /config/auth unless an absolute path is specified.

Use the **set** form of this command to specify the name of the CRL file.

Use the **delete** form of this command to remove the name of the CRL file.

Use the **show** form of this command to display CRL file configuration.

security vpn ipsec site-to-site peer <peer> authentication x509 key file <file-name>

Specifies the name of the VPN server's private key file for IPsec authentication of the VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **authentication x509 key file** *file-name*

delete security vpn ipsec site-to-site peer *peer* **authentication x509 key file**

show security vpn ipsec site-to-site peer *peer* **authentication x509 key file**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

file-name

The name of the VPN server's private key file. This parameter is mandatory if **authentication mode** is **x509**.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        authentication {
                            x509 {
                                key {
                                    file file-name
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the name of the VPN server's private key file. The VPN server's private key certifies the identity of the VPN server.

The file is assumed to be in /config/auth unless an absolute path is specified.

Use the **set** form of this command to specify the location of the VPN server's private key file.

Use the **delete** form of this command to remove the location of the VPN server's private key file.

Use the **show** form of this command to display VPN server private key file configuration.

security vpn ipsec site-to-site peer <peer> authentication x509 key password <password>

Specifies the password that protects the VPN server's private key.

Syntax **set security vpn l2tp remote-access ipsec-settings authentication x509 key password** *password*

delete security vpn l2tp remote-access ipsec-settings authentication x509 key password

show security vpn l2tp remote-access ipsec-settings authentication x509 key password

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

password

The password protecting the VPN server's private key file.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            l2tp {
                remote-access {
                    ipsec-settings {
                        authentication {
                            x509 {
                                key {
                                    password password
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify a password that protects the VPN server's private key.

Use the **set** form of this command to specify the password for the VPN server's private key.

Use the **delete** form of this command to remove the password for the VPN server's private key.

Use the **show** form of this command to display VPN servers private key password configuration.

security vpn ipsec site-to-site peer <peer> connection-type

Specifies the type of peer connection.

Syntax

set security vpn ipsec site-to-site peer *peer* **connection-type** { **initiate** | **respond** }

delete security vpn ipsec site-to-site peer *peer* **connection-type**

show security vpn ipsec site-to-site peer *peer* **connection-type**

Command Default

A connection to the remote peer is initiated by the local peer unless the remote peer is set to 0.0.0.0, @id, or any.

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

initiate

Indicates that the connection to the remote peer will be initiated by the local peer unless the remote peer is set to **0.0.0.0**, **@id**, or **any**.This is the default behavior.

respond

Indicates that the local peer will not initiate a connection to the remote peer, but will respond to connections initiated by the remote peer.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer{
                        connection-type [initiate|respond]
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the type of peer connection.

Use the **set** form of this command to specify the type of peer connection.

Use the **delete** form of this command to return the connection type to its default behavior.

Use the **show** form of this command to view connection type configuration.

security vpn ipsec site-to-site peer <peer> default-esp-group <name>

Specifies a default ESP configuration to use for all tunnels to the peer.

Syntax **set security vpn ipsec site-to-site peer** *peer* **default-esp-group** *name*

delete security vpn ipsec site-to-site peer *peer* **default-esp-group**

show security vpn ipsec site-to-site peer *peer* **default-esp-group**

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

name

Specifies the named ESP configuration (ESP group) to be used by default for all connections. The ESP group must have already been defined, using security vpn ipsec esp-group <name> on page 101.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    default-esp-group name
                }
            }
        }
    }
}
```

Usage Guidelines Use this command to specify a default ESP configuration to use for all tunnels to the peer. This setting can be overridden on a per-tunnel basis by using security vpn ipsec site-to-site peer <peer> tunnel <tunnel> esp-group <name> on page 150.

Use the **set** form of this command to specify an ESP configuration to use for all connections by default.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec site-to-site peer <peer> description <desc>

Specifies a description for a VPN peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **description** *desc*

delete security vpn ipsec site-to-site peer *peer* **description**

show security vpn ipsec site-to-site peer *peer* **description**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

desc

A brief description for the VPN peer. If the description contains space characters, it must be enclosed in double quotes.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    description desc
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify a description for the VPN peer.

Use the **set** form of this command to specify the description for the VPN peer.

Use the **delete** form of this command to remove the description for the VPN peer.

Use the **show** form of this command to view the description for the VPN peer.

security vpn ipsec site-to-site peer <peer> dhcp-interface <interface>

Specifies a DHCP client interface to use for the connection.

Syntax

set security vpn ipsec site-to-site peer *peer* **dhcp-interface** *interface*

delete security vpn ipsec site-to-site peer *peer* **dhcp-interface**

show security vpn ipsec site-to-site peer *peer* **dhcp-interface**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

interface

The interface to use for the VPN connection (e.g. dp0p1p1). Note that the interface must already have IPsec VPN enabled, using security vpn ipsec ipsec-interfaces interface <if-name> on page 116, and must be configured as a DHCP client.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer{
                        dhcp-interface interface
                    }
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify a DHCP client interface to use for the connection. The connection will be automatically restarted if the IP address changes.

NOTE
This option cannot be used if security vpn ipsec site-to-site peer <peer> local-address <address> on page 144 is also set.

Use the **set** form of this command to specify a DHCP interface to use for the connection.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec site-to-site peer <peer> ike-group <group>

Specifies the named IKE configuration to be used for a peer connection.

Syntax

set security vpn ipsec site-to-site peer *peer* **ike-group** *group*

delete security vpn ipsec site-to-site peer *peer* **ike-group**

show security vpn ipsec site-to-site peer *peer* **ike-group**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

group

Mandatory. The named IKE configuration to be used for this connection. The IKE configuration must have already been defined, using security vpn ipsec ike-group <name> on page 109.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer{
                        ike-group group
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify a named IKE configuration (an IKE group) to be used for an IPsec peer connection.

Use the **set** form of this command to specify the IKE group.

Use the **delete** form of this command to remove IKE group configuration.

Use the **show** form of this command to view IKE group configuration.

security vpn ipsec site-to-site peer <peer> local-address <address>

Specifies the local IP address to be used as the source IP for packets destined for the remote peer.

Syntax

set security vpn ipsec site-to-site peer *peer* **local-address** *address*

delete security vpn ipsec site-to-site peer *peer* **local-address**

show security vpn ipsec site-to-site peer *peer* **local-address**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

address

Mandatory. The local IPv4 or IPv6 address to be used as the source IP for packets destined for the remote peer. If the physical interface has a dynamic IPv4 address, then the **local-address** must be set to **any**.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    local-address address
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify the local IP address to be used as the source IP address for packets destined for the remote peer.

The interface associated with this address must already have IPsec VPN enabled, using security vpn ipsec ipsec-interfaces interface <if-name> on page 116.

The address type must match that of the peer. For example, if the peer address is IPv4, then the local-address must also be IPv4.

The local-address must be set to any in cases where the local external IPv4 address is dynamic or unknown; for example, when the address is supplied by a PPPoE connection or DHCP server. If you use an address of *any*, you must set the local authentication ID using security vpn ipsec site-to-site peer <peer> authentication id <id> on page 129.

When the *local-address* is set to *any*, the default route is used and the connection will not be automatically updated if the IP address changes (a reset vpn ipsec-peer <peer> on page 85 is required when the IP address changes). A better alternative for use with DHCP client interfaces is security vpn ipsec site-to-site peer <peer> dhcp-interface <interface> on page 142.

NOTE

The *local-address* option cannot be used if security vpn ipsec site-to-site peer <peer> dhcp-interface <interface> on page 142 is also set.

If the VPN tunnel is being clustered for high availability, the local-address attribute must be the cluster IP address, not the IP address configured for the physical interface. Otherwise, the local-address must be the address configured for the physical interface.

Use the **set** form of this command to specify the local IP address to be used as the source IP for packets destined for the remote peer.

Use the **delete** form of this command to remove local IP address configuration.

Use the **show** form of this command to view local IP address configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> allow-nat-networks <state>

Specifies whether or not a connection to a private network is allowed.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-nat-networks** *state*

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-nat-networks**

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-nat-networks**

Command Default

A connection to a private network is not allowed (disabled).

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple tunnel configuration nodes within the peer configuration.

state

Allows connection to a defined network of private IP addresses on a per-tunnel basis. Supported values are as follows:

enable—Allow connection to the private network.

disable—Do not allow connection to the private network.

This option is mandatory if the **allow-public-networks** is enabled; optional otherwise. The allowed private network must be defined by using security vpn ipsec nat-networks allowed-network <ipv4net> on page 119.

If this option is enabled, any value set for the **remote prefix** option is ignored.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
            site-to-site {
                peer peer {
                    tunnel tunnel {
                        allow-nat-networks state
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify whether or not a connection to a private network is allowed.

Use the **set** form of this command to specify whether or not a connection to a private network is allowed.

Use the **delete** form of this command to remove the configuration and return it to the default behavior.

Use the **show** form of this command to view the configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> allow-public-networks <state>

Specifies whether or not a connection to a public network is allowed.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-public-networks** *state*

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-public-networks**

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **allow-public-networks**

Command Default

A connection to a public network is not allowed (disabled).

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

state

Allows connections to public IP addresses on a per-tunnel basis. Supported values are as follows:

enable—Allows connections to public networks.

disable—Does not allow connections to public networks.

This option requires that the **allow-nat-networks** option be enabled, and that allowed NAT networks be specified by using security vpn ipsec nat-networks allowed-network <ipv4net> on page 119.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            allow-public-networks state
                        }
                    }
                }
            }
    }
}
```

Usage Guidelines

Use this command to specify whether or not a connection to a public network is allowed.

Use the **set** form of this command to specify whether or not a connection to a public network is allowed.

Use the **delete** form of this command to remove the configuration and return it to the default behavior.

Use the **show** form of this command to view the configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> disable

Disables a VPN tunnel without discarding configuration.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **disable**

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **disable**

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel*

Command Default

The VPN tunnel configuration is enabled.

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            disable
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to disable the VPN tunnel without discarding configuration. The tunnel can then be re-enabled at a later time without the need to redefine the configuration.

Use the **set** form of this command to disable the tunnel.

Use the **delete** form of this command to enable the tunnel.

Use the **show** form of this command to view the VPN tunnel configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> esp-group <name>

Specifies an ESP configuration to use for this tunnel.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **esp-group** *name*

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **esp-group**

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **esp-group**

Command Default

The ESP group specified by security vpn ipsec site-to-site peer <peer> default-esp-group <name> on page 140 will be used.

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

name

Specifies the named ESP configuration (ESP group) to be used for this connection. The ESP group must have already been defined, using security vpn ipsec esp-group <name> on page 101.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            esp-group name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify an ESP configuration to use for this connection. It will override the ESP group specified by security vpn ipsec site-to-site peer <peer> default-esp-group <name> on page 140 which will be used by default.

Use the **set** form of this command to specify an ESP configuration to use for this connection.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> local

Defines local configuration options for the IPsec tunnel.

Syntax **set security vpn ipsec site-to-site peer** *peer* **tunnel** *tunnel* **local** [**port** *port* | **prefix** *prefix*]

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **local** [**port** | **prefix**]

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **local** [**port** | **prefix**]

Parameters *peer*

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

port

Applicable only when the protocol is TCP or UDP. The local port to match. Only traffic from or to this port on the local subnet will travel through this tunnel. Supported formats are as follows:

port-name—Matches the name of an IP service; for example, **http**. You can specify any service name in the file **/etc/services**.

port-num—Matches a port number. The numbers range from 1 through 65535.

The default is **all**.

prefix

Mandatory. The local subnet to which the remote VPN gateway will have access. For IPv4, the format is an IPv4 network address, where network address **0.0.0.0/0** means any local subnet. For IPv6, the format is an IPv6 network address, where network address **0::0/0** means any local subnet.

NOTE
The address type (IPv4 or IPv6) must match that of the **remote prefix**.

The default is the subnet the **local-address** is on.

Modes Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            local {
                                port port
                                prefix prefix
                        }
                    }
                }
```

```
                }
            }
        }
    }
```

Usage Guidelines Use this command to define local configuration options for the IPsec tunnel.

Use the **set** form of this command to set the local tunnel characteristics.

Use the **delete** form of this command to remove local tunnel configuration.

Use the **show** form of this command to view local tunnel configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> protocol <protocol>

Specifies the protocol to match for traffic to enter the tunnel.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **protocol** *protocol*

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **protocol**

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **protocol**

Command Default

The default is **all**.

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

protocol

Any protocol literals or numbers listed in the file **/etc/protocols** can be used. The keywords **tcp_udp** (for both TCP and UDP) and **all** (for all protocols) are also supported.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            protocol protocol
                        }
                    }
                }
            }
        }
    }
```

Usage Guidelines

Use this command to specify the protocol to match for traffic to enter the tunnel.

Use the **set** form of this command to specify the protocol.

Use the **delete** form of this command to remove protocol configuration.

Use the **show** form of this command to view protocol configuration.

security vpn ipsec site-to-site peer <peer> tunnel <tunnel> remote

Defines remote configuration options for the IPsec tunnel.

Syntax

set security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **remote** [**port** *port* | **prefix** *prefix*]

delete security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **remote** [**port** | **prefix**]

show security vpn ipsec site-to-site peer *peer* **tunnel** *tunnel* **remote** [**port** | **prefix**]

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address, an IPv6 address, a hostname (IPv4 networks only), an authentication ID, or **0.0.0.0**.

tunnel

Mandatory. Multi-node. An integer that uniquely identifies this tunnel configuration for this peer VPN gateway. Each tunnel corresponds to a distinct connection configuration. The numbers range from 0 through 4294967295.

A given VPN peer may have more than one tunnel configuration, but each peer must have at least one. To define more than one tunnel configuration for a peer, create multiple **tunnel** configuration nodes within the peer configuration.

port

Applicable only when the protocol is TCP or UDP. The remote port to match. Only traffic from or to this port on the remote subnet will travel through this tunnel. Supported formats are as follows:

port-name—Matches the name of an IP service; for example, **http**. You can specify any service name in the file **/etc/services**.

port-num—Matches a port number. The numbers range from 1through 65535.

The default is **all**.

prefix

Mandatory. The remote subnet behind the remote VPN gateway, to which the Brocade vRouter will have access. For IPv4, the format is an IPv4 network address, where network address **0.0.0.0/0** means any subnet behind the remote VPN gateway. For IPv6, the format is an IPv6 network address, where network address **0::0/0** means any local subnet.

NOTE
The address type (IPv4 or IPv6) must match that of the **local prefix**.

This option is ignored if **allowed-nat-networks** is enabled.

The default is the subnet of the peer.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        tunnel tunnel {
                            remote {
                                port port
                                prefix prefix
```

```
                                    }
                                }
                            }
                        }
                    }
                }
            }
```

Usage Guidelines Use this command to define local configuration options for the IPsec tunnel.

Use the **set** form of this command to set the local tunnel characteristics.

Use the **delete** form of this command to remove local tunnel configuration.

Use the **show** form of this command to view local tunnel configuration.

security vpn ipsec site-to-site peer <peer> vti bind <vtix>

Binds the IPsec site-to-site VPN tunnel to a virtual tunnel interface.

Syntax

set security vpn ipsec site-to-site peer *peer* **vti bind** *vtix*

delete security vpn ipsec site-to-site peer *peer* **vti bind**

show security vpn ipsec site-to-site peer *peer* **vti bind**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address.

vtix

Mandatory. The virtual tunnel interface to bind the IPsec site-to-site VPN tunnel to. The virtual tunnel interface must have already been defined, using interfaces vti <vtix> on page 165.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        vti {
                            bind vtix
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to bind an IPsec site-to-site VPN tunnel to a virtual tunnel interface.

Use the **set** form of this command to bind the IPsec site-to-site VPN tunnel to the specified virtual tunnel interface.

Use the **delete** form of this command to remove the bind to the virtual tunnel interface.

Use the **show** form of this command to view the bind configuration.

security vpn ipsec site-to-site peer <peer> vti esp-group <name>

Specifies the ESP configuration to use for the IPsec site-to-site VPN tunnel.

Syntax

set security vpn ipsec site-to-site peer *peer* **vti esp-group** *name*

delete security vpn ipsec site-to-site peer *peer* **vti esp-group**

show security vpn ipsec site-to-site peer *peer* **vti esp-group**

Parameters

peer

Mandatory. The address of the far-end VPN gateway. The format is an IPv4 address.

name

Mandatory. Specifies the named ESP configuration (ESP group) to be used for the connection. The ESP group must have already been defined, using security vpn ipsec esp-group <name> on page 101.

Modes

Configuration mode

Configuration Statement

```
security {
        vpn {
            ipsec {
                site-to-site {
                    peer peer {
                        vti {
                            esp-group name
                        }
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to specify an ESP configuration to use for this connection. It will override the ESP group specified by security vpn ipsec site-to-site peer <peer> default-esp-group <name> on page 140 which will be used by default.

Use the **set** form of this command to specify an ESP configuration to use for this VPN tunnel.

Use the **delete** form of this command to remove the configuration.

Use the **show** form of this command to view the configuration.

security vpn rsa-keys

Records RSA keys for the local host.

Syntax **set security vpn rsa-keys** [**local-key file** *file-name* | **rsa-key-name** *name* **rsa-key** *key*]

delete security vpn rsa-keys local-key file [**local-key file** | **rsa-key-name** [*name* **rsa-key**]]

show security vpn rsa-keys local-key file [**local-key file** | **rsa-key-name** [*name* **rsa-key**]]

Parameters *file-name*

Specifies the name and location of the file containing the RSA digital signature of the local host (both public key and private key). By default, the RSA digital signature for the local host is recorded in **/config/ipsec.d/rsa-keys/**.

name

A mnemonic name for the remote key. This is the name you refer to when configuring RSA configuration in site-to-site connections.

key

The RSA public key data for the remote peer.

Modes Configuration mode

```
security {
        vpn {
            rsa-keys {
                local-key {
                    file file-name
        }
            rsa-key-name name {
                rsa-key key
           }
    }
}
```

Usage Guidelines Use this command to view or change the location of the file containing RSA key information for the local host, or to record an RSA public key for a remote host.

The RSA digital signature for the local host can be generated using generate vpn rsa-key on page 83 in operational mode. Once generated, the key is stored at the location specified by the *local-key rsa-key-name* option. By default, this is the *localhost.key* file in the /config/ipsec.d/rsa-keys/ directory.

You must also enter the public key of the remote peer, as the *rsa-key-namenamersa-key* attribute. Digital signatures are lengthy, so to configure this value copy it as text into your clipboard and paste it into the configuration. Once recorded with a mnemonic name, you can refer to the RSA key by the name in site-to-site connection configurations.

Use the **set** form of this command to set RSA key configuration.

Use the **delete** form of this command to remove RSA key configuration.

Use the **show** form of this command to view RSA key configuration.

Virtual Tunnel Interface Overview

This chapter provides a brief overview of virtual tunnel interfaces.

Virtual tunnel interfaces

A virtual tunnel interface provides a termination point for a site-to-site IPsec VPN tunnel and allows it to behave like other routable interfaces. In addition to simplifying the IPsec configuration, it enables many common capabilities to be used because the endpoint is associated with an actual interface.

Traffic being routed to a virtual tunnel interface is encrypted prior to being sent through the tunnel. Traffic arriving from a virtual tunnel interface is decrypted prior to its exposure to the routing system.

The virtual tunnel interface on the Brocade vRouter is compatible with third party VTI/route-based VPN connections and is sometimes required for connectivity with public cloud offerings.

Benefits of virtual tunnel interfaces

The virtual tunnel interface provides the following benefits over non-VTI IPsec VPN connections:

1. They are capable of having traffic routed to them.
2. They are capable of passing routing protocols over them.
3. They do not require local or remote subnets to be specified.
4. They operate as if the peer interfaces are directly connected.

Restrictions and limitations

The virtual tunnel interface has the following restrictions and limitations:

1. It is only supported with IPv4, and not IPv6.
2. Only unicast and multicast IP traffic is allowed.
3. The Brocade vRouter uses *fwmark* in the kernel sk_buff to uniquely identify virtual tunnel interfaces (as well as entities associated with other features). For this purpose, the Brocade vRouter uses *fwmark* greater than or equal to 0x7FFF FFFF. If you intend to use *fwmark* directly for another purpose, you should not use values greater than or equal to 0x7FFF FFFF.
4. Because the virtual tunnel interface and IP-in-IP tunnels use the same IP protocol type, it is not possible to use both of these tunnel types between the same tunnel endpoints.
5. The virtual tunnel interface does not support Time to Live (TTL) and Type of Service (ToS).

6. The IPsec mode must be configured as tunnel. See Restrictions and limitations.
7. Unlike other site-to-site IPsec VPN tunnels, the local and remote proxies are implicitly 0.0.0.0/0 so the remote and local subnets do not need to be specified explicitly.

Virtual Tunnel Interface Configuration

This chapter provides configuration examples for virtual tunnel interfaces.

Creating a virtual tunnel interface

To create a virtual tunnel interface, perform the following steps in configuration mode.

TABLE 49 Creating a virtual tunnel interface

Step	Command

Create a vti interface and assign it an IP address.	`vyatta@vyatta# set interfaces vti vti0 address 192.0.2.249/30`
	
	`[edit]`

Commit the configuration.	`vyatta@vyatta# commit`

View the configuration.	`vyatta@vyatta# show interfaces vti`
	
	`vti0 {`
	`address 192.0.2.249/30`
	`}`

Once the virtual tunnel interface is created an IPsec site-to-site VPN tunnel can be bound to it. For a complete configuration example showing the creation of the virtual tunnel interface and the binding of an IPsec site-to-site VPN tunnel to it, see Basic site-to-site connection using a virtual tunnel interface on page 73.

Virtual Tunnel Interface Commands

clear interfaces vti counters

Clears statistics counters for virtual tunnel interfaces.

Syntax **clear interfaces vti** [*vtix*] **counters**

Command Default Clears counters for all virtual tunnel interfaces.

Parameters *vtix*

Clears statistics for the specified virtual tunnel interface.

Modes Operational mode

Usage Guidelines Use this command to clear counters on virtual tunnel interfaces.

interfaces vti <vtix>

Defines a virtual tunnel interface.

Syntax

set interfaces vti *vtix*

delete interfaces vti *vtix*

show interfaces vti *vtix*

Parameters

vtix

Multi-node. The identifier for the virtual tunnel interface you are defining; for example **vti0**.

You can define multiple virtual tunnel interfaces by creating multiple **vti** configuration nodes.

Modes

Configuration mode

Configuration Statement

```
interfaces {
    vti vtix {
    }
}
```

Usage Guidelines

Use this command to define a virtual tunnel interface.

Use the **set** form of this command to create a virtual tunnel interface.

Use the **delete** form of this command to remove a virtual tunnel interface.

Use the **show** form of this command to view virtual tunnel interface configuration.

interfaces vti <vtix> address <ipv4>

Sets an IP address and network prefix for a virtual tunnel interface.

Syntax

set interfaces vti *vtix* **address** *ipv4*

delete interfaces vti *vtix* **address** [*ipv4*]

show interfaces vti *vtix* **address**

Parameters

vtix

The identifier of the virtual tunnel interface. The identifiers range from **vti0** through **vti** *x*, where x is a positive integer.

ipv4

Defines an IPv4 address on this interface. The format is *ip-address* / *prefix* (for example, 192.168.1.77/24).

You can define multiple IP addresses for a single virtual tunnel interface, by creating multiple **address** configuration nodes.

Modes

Configuration mode

Configuration Statement

```
interfaces {
    vti vtix {
        address ipv4
    }
}
```

Usage Guidelines

Use this command to set the IP address and network prefix for a virtual tunnel interface.

Use the **set** form of this command to set the IP address and network prefix. You can set more than one IP address for the interface by creating multiple address configuration nodes.

Use the **delete** form of this command to remove IP address configuration.

Use the **show** form of this command to view IP address configuration.

interfaces vti <vtix> description <description>

Specifies a description for a virtual tunnel interface.

Syntax **set interfaces vti** *vtix* **description** *description*

delete interfaces vti *vtix* **description**

show interfaces vti *vtix* **description**

Parameters *vtix*

The identifier of the virtual tunnel interface. The identifiers range from **vti0** through **vti** *x*, where x is a positive integer.

description

A mnemonic name or description for the virtual tunnel interface.

Modes Configuration mode

Configuration Statement

```
interfaces {
    vti vtix {
            description description
    }
}
```

Usage Guidelines Use this command to set a description for a virtual tunnel interface.

Use the **set** form of this command to specify the description.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to view description configuration.

interfaces vti <vtix> disable

Disables a virtual tunnel interface without discarding configuration.

Syntax

set interfaces vti *vtix* **disable**

delete interfaces vti *vtix* **disable**

show interfaces vti *vtix*

Parameters

vtix

The identifier of the virtual tunnel interface. The identifier ranges from **vti0** through **vti** *x*, where x is a positive integer.

Modes

Configuration mode

Configuration Statement

```
interfaces {
    vti vtix {
            disable
    }
}
```

Usage Guidelines

Use this command to disable a virtual tunnel interface without discarding configuration.

Use the **set** form of this command to disable the interface.

Use the **delete** form of this command to enable the interface.

Use the **show** form of this command to view virtual tunnel interface configuration.

interfaces vti <vtix> firewall <state>

Applies a firewall instance, or rule set, to an interface.

Syntax

set interfaces vti *vtix* **firewall** { **in** *firewall-name* | **l2** *name* | **out** *firewall-name* }

delete interfaces vti *vtix* **firewall** [**in** *firewall-name* | **l2** *name* | **out** *firewall-name*]

show interfaces vti *vtix* **firewall** [**in** | **l2** | **out**]

Parameters

interface

A type of interface. For detailed keywords and arguments, refer to Supported Interface Types on page 175.

in *firewall-name*

Applies a firewall rule set to inbound traffic on the specified interface.

l2

Applies a firewall rule set to bridge traffic.

out *firewall-name*

Applies a firewall rule set to outbound traffic on the specified interface.

Modes

Configuration mode

Configuration Statement

```
interfaces interface {
    vto vtix        firewall {
            in firewall-name
            l2 name
            out firewall-name
    }
}
```

Usage Guidelines

Use this command to apply an IPv6 firewall instance, or rule set, to an interface.

A firewall has no effect on traffic traversing the system or destined to the system until a firewall rule set has been applied to an interface or a virtual interface by using this command.

To use the firewall feature, you must define a firewall rule set as a named firewall instance by using the **security firewall name** command. You then apply the firewall instance to interfaces, virtual interfaces, or both by using this command. After the instance is applied, the instance acts as a packet filter.

The firewall instance filters packets in one of the following ways, depending on what you specify when you apply it.

- *in* —If you apply the rule set as *in*, the firewall filters packets entering the interface.
- *out* —If you apply the rule set as *out*, the firewall filters packets leaving the interface.

For each interface, you can apply up to three firewall instances: one firewall *in* instance, one firewall *out* instance, and one firewall local instance.

Make sure the firewall instance you apply to an interface is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to an interface, the implicit firewall rule of *allow all* is applied.

Use the **set** form of this command to apply an IPv6 firewall instance, or rule set, to an interface.

Use the **delete** form of this command to delete an IPv6 firewall instance, or rule set, from an interface.

Use the **show** form of this command to display the configuration of an IPv6 firewall instance, or rule set, for an interface.

interfaces vti <vtix> mtu <mtu>

Sets the MTU for a virtual tunnel interface.

Syntax **set interfaces vti** *vtix* **mtu** *mtu*

delete interfaces vti *vtix* **mtu**

show interfaces vti *vtix* **mtu**

Command Default If this value is not set, the default MTU of 1500 is used.

Parameters *vtix*

The identifier of the virtual tunnel interface. The identifiers range from **vti0** through **vti** *x*, where x is a positive integer.

mtu

Sets the MTU, in octets, for the interface. The numbers range from 68 through 9000.

Modes Configuration mode

Configuration Statement

```
interfaces {
    vti vtix {
        mtu mtu
    }
}
```

Usage Guidelines Use this command to set the maximum transmission unit (MTU) for an virtual tunnel interface.

During forwarding, IPv4 packets larger than the MTU are fragmented unless the “Don't Fragment” (DF) bit is set in the IP header. In that case, the packets are dropped and an ICMP “fragmentation needed” message is returned to the sender.

Use the **set** form of this command to specify the MTU.

Use the **delete** form of this command to remove MTU value and restore the default behavior.

Use the **show** form of this command to view MTU configuration.

monitor interfaces vti <vtix> traffic

Displays (captures) traffic on a virtual tunnel interface.

Syntax **monitor interfaces vti** *vtix* **traffic** [**detail** [**filter filter-name** | **unlimited** [**filter filter-name**]] | **filter filter-name** | **save** *filename* | **unlimited** [**filter filter-name**]]]

Parameters *vtix*

The identifier of an virtual tunnel interface. The identifiers range from **vti0** through **vtix,** where **x** is a non-negative integer.

detail

Provides detailed information about the monitored VRRP traffic.

filter-name

Applies the specific PCAP (packet capture) filter to traffic.

unlimited

Monitors an unlimited amount of traffic.

filename

Saves the monitored traffic to the specified file.

Modes Operational mode

Usage Guidelines Use this command to capture traffic on a virtual tunnel interface. Type <Ctrl>+c to stop the output.

Examples The following example shows captured data on interface vti0.

```
vyatta@vyatta:~$ monitor interfaces vti vti0 traffic
Capturing traffic on vti0 ...
  4.568357 192.168.1.254 -> 238.255.255.251 SSDP NOTIFY * HTTP/1.1
  4.568372 192.168.1.254 -> 238.255.255.251 SSDP NOTIFY * HTTP/1.1
...
```

show interfaces vti

Displays information and statistics about Virtual Tunnel interfaces.

Syntax **show interfaces vti** [*vtix*]

Command Default Information is displayed for all Virtual Tunnel interfaces.

Parameters *vtix*

Displays information for the specified Virtual Tunnel interface. The identifiers range from **vti0** through **vtix**, where x is a positive integer.

Modes Operational mode

Usage Guidelines Use this command to view operational status of Virtual Tunnel interfaces.

Examples The following example shows information for all Virtual Tunnel interfaces.

```
vyatta@vyatta:~$ show interfaces vti
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface        IP Address                     S/L  Description
---------        ----------                     ---  -----------
vti2             100.0.0.1/24                   u/u
```

The following example shows information for interface vti2.

```
vyatta@vyatta:~$ show interfaces vti vti2
vti2: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN
    link/ipip 12.0.0.1 peer 12.0.0.2
    inet 100.0.0.1/24 scope global vti2
    RX:  bytes    packets     errors    dropped    overrun      mcast
            84          1          0          0          0          0
    TX:  bytes    packets     errors    dropped    carrier collisions
            84          1          0          0          0          0
```

show interfaces vti detail

Displays detailed information about Virtual Tunnel interfaces.

Syntax **show interfaces vti detail**

Modes Operational mode

Usage Guidelines Use this command to view detailed statistics and configuration information about Virtual Tunnel interfaces.

Examples The following example shows the first screen of output for show interfaces vti detail.

```
vyatta@vyatta:~$ show interfaces vti detail
vti2: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN
    link/ipip 12.0.0.1 peer 12.0.0.2
    inet 100.0.0.1/24 scope global vti2
    RX:  bytes    packets     errors    dropped    overrun      mcast
            84          1          0          0          0          0
    TX:  bytes    packets     errors    dropped    carrier collisions
            84          1          0          0          0          0
```

show interfaces vti <vtix> brief

Displays a brief status for an Virtual Tunnel interface.

Syntax **show interfaces vti** *vtix* **brief**

Parameters *vtix*

The identifier of an Virtual Tunnel interface. The identifiers range from **vti0** through **vtix**, where x is a positive integer.

Modes Operational mode

Usage Guidelines Use this command to view the status of a virtual tunnel interface.

Examples The following example shows brief status for interface vti2.

```
vyatta@vyatta:~$ show interfaces vti vti2 brief
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface        IP Address                     S/L  Description
---------        ----------                     ---  -----------
vti2             100.0.0.1/24                   u/u
```

Supported Interface Types

The following table shows the syntax and parameters of supported interface types. Depending on the command, some of these types may not apply.

Interface Type	Syntax	Parameters
Bridge	**bridge** *brx*	*brx*: The name of a bridge group. The name ranges from br0 through br999.
Data plane	**dataplane** *interface-name*	*interface-name*: The name of a data plane interface. Following are the supported formats of the interface name: • **dp***x***p***y***p***z***—The name of a data plane interface, where — **dp***x* specifies the data plane identifier (ID). Currently, only dp0 is supported. — p*y* specifies a physical or virtual PCI slot index (for example, p129). — **p***z* specifies a port index (for example, p1). For example, dp0p1p2, dp0p160p1, and dp0p192p1. • **dp***x***em***y* —The name of a data plane interface on a LAN-on-motherboard (LOM) device that does not have a PCI slot, where **em***y* specifies an embedded network interface number (typically, a small number). For example, dp0em3. • **dp***x***s***y* —The name of a data plane interface on a device that is installed on a virtual PCI slot, where *x***s***y* specifies an embedded network interface number (typically, a small number). For example, dp0s2. Currently, this format applies only when using the KVM or Hyper-V platforms. • **dp***x***P***n***p***y***p***z* —The name of a data plane interface on a device that is installed on a secondary PCI bus, where **P***n* specifies the bus number. You can use this format to name data plane interfaces on large physical devices with multiple PCI buses. For these devices, it is possible to have network interface cards installed on different buses with these cards having the same slot ID. The value of *n* must be an integer greater than 0. For example, dp0P1p162p1 and dp0P2p162p1.

Interface Type	Syntax	Parameters
Data plane vif	**dataplane** *interface-name* **vif** *vif-id* [**vlan** *vlan-id*]	*interface-name*: Refer to the preceding description. *vif-id*: A virtual interface ID. The ID ranges from 1 through 4094. *vlan-id*: The VLAN ID of a virtual interface. The ID ranges from 1 through 4094.
Loopback	**loopback lo** or **loopback lo***n*	*n*: The name of a loopback interface, where *n* ranges from 1 through 99999.
OpenVPN	**openvpn** *vtunx*	*vtunx*: The identifier of an OpenVPN interface. The identifier ranges from vtun0 through vtun*x*, where *x* is a nonnegative integer.
Tunnel	**tunnel** *tunx* or **tunnel** *tunx* **parameters**	*tunx*: The identifier of a tunnel interface you are defining. The identifier ranges from tun0 through tun*x*, where *x* is a nonnegative integer.
Virtual tunnel	**vti** *vtix*	*vtix*: The identifier of a virtual tunnel interface you are defining. The identifier ranges from vti0 through vti*x*, where *x* is a nonnegative integer. **Note:** This interface does not support IPv6.
VRRP	*parent-interface* **vrrp vrrp-group** *group*	*parent-interface*: The type and identifier of a parent interface; for example, data plane dp0p1p2 or bridge br999. *group*: A VRRP group identifier. The name of a VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number; for example, dp0p1p2v99. Note that VRRP interfaces support the same feature set as does the parent interface.

List of Acronyms

Acronym	Description
ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
AH	Authentication Header
AMI	Amazon Machine Image
API	Application Programming Interface
AS	autonomous system
ARP	Address Resolution Protocol
AWS	Amazon Web Services
BGP	Border Gateway Protocol
BIOS	Basic Input Output System
BPDU	Bridge Protocol Data Unit
CA	certificate authority
CCMP	AES in counter mode with CBC-MAC
CHAP	Challenge Handshake Authentication Protocol
CLI	command-line interface
DDNS	dynamic DNS
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol version 6
DLCI	data-link connection identifier
DMI	desktop management interface
DMVPN	dynamic multipoint VPN
DMZ	demilitarized zone
DN	distinguished name
DNS	Domain Name System
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
eBGP	external BGP
EBS	Amazon Elastic Block Storage
EC2	Amazon Elastic Compute Cloud
EGP	Exterior Gateway Protocol
ECMP	equal-cost multipath
ESP	Encapsulating Security Payload

Acronym	Description
FIB	Forwarding Information Base
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HDLC	High-Level Data Link Control
I/O	Input/Output
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IGMP	Internet Group Management Protocol
IGP	Interior Gateway Protocol
IPS	Intrusion Protection System
IKE	Internet Key Exchange
IP	Internet Protocol
IPOA	IP over ATM
IPsec	IP Security
IPv4	IP Version 4
IPv6	IP Version 6
ISAKMP	Internet Security Association and Key Management Protocol
ISM	Internet Standard Multicast
ISP	Internet Service Provider
KVM	Kernel-Based Virtual Machine
L2TP	Layer 2 Tunneling Protocol
LACP	Link Aggregation Control Protocol
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LLDP	Link Layer Discovery Protocol
MAC	medium access control
mGRE	multipoint GRE
MIB	Management Information Base
MLD	Multicast Listener Discovery
MLPPP	multilink PPP
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit
NAT	Network Address Translation
NBMA	Non-Broadcast Multi-Access
ND	Neighbor Discovery

Acronym	Description
NHRP	Next Hop Resolution Protocol
NIC	network interface card
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OSPFv2	OSPF Version 2
OSPFv3	OSPF Version 3
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PAT	Port Address Translation
PCI	peripheral component interconnect
PIM	Protocol Independent Multicast
PIM-DM	PIM Dense Mode
PIM-SM	PIM Sparse Mode
PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PTMU	Path Maximum Transfer Unit
PVC	permanent virtual circuit
QoS	quality of service
RADIUS	Remote Authentication Dial-In User Service
RHEL	Red Hat Enterprise Linux
RIB	Routing Information Base
RIP	Routing Information Protocol
RIPng	RIP next generation
RP	Rendezvous Point
RPF	Reverse Path Forwarding
RSA	Rivest, Shamir, and Adleman
Rx	receive
S3	Amazon Simple Storage Service
SLAAC	Stateless Address Auto-Configuration
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SONET	Synchronous Optical Network
SPT	Shortest Path Tree

Acronym	Description
SSH	Secure Shell
SSID	Service Set Identifier
SSM	Source-Specific Multicast
STP	Spanning Tree Protocol
TACACS+	Terminal Access Controller Access Control System Plus
TBF	Token Bucket Filter
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity Protocol
ToS	Type of Service
TSS	TCP Maximum Segment Size
Tx	transmit
UDP	User Datagram Protocol
VHD	virtual hard disk
vif	virtual interface
VLAN	virtual LAN
VPC	Amazon virtual private cloud
VPN	virtual private network
VRRP	Virtual Router Redundancy Protocol
WAN	wide area network
WAP	wireless access point
WPA	Wired Protected Access