JUNOS OS MULTIPOINT VPN CONFIGURATION WITH NEXT-HOP TUNNEL BINDING

Table of Contents

Table of figures

Introduction

Juniper Networks® Junos® operating system runs on Juniper Networks J Series Services Routers and Juniper Networks SRX Series Services Gateways. Junos OS provides not only a powerful operating system, but also a rich IP services tool kit. It has enhanced security and VPN capabilities via Juniper's firewall/IPsec VPN platforms, which includes the Juniper Networks SSG Series Secure Services Gateways. This application note discusses configuration of a multipoint topology, which is commonly used for hub-and-spoke environments, using these systems.

Scope

Route-based VPNs from a central hub device to multiple spoke devices are used in this application. (Multipoint with policy-based VPNs is not supported in Junos OS.)

This document is intended for network design and security engineers, as well as implementation partners supporting customers requiring secure connectivity over public networks.

Design Considerations

This document applies to the following devices:

- J Series Services Routers running:
  - Junos OS 9.4 and above
  - Junos OS with Enhanced Services 8.5 through 9.3
- SRX Series Services Gateways

There are many ways to implement a hub-and-spoke VPN topology using the concepts of route-based VPNs. One way is to configure a separate secure tunnel (st0) logical unit for every spoke site. However, if a device has too many peers, the number of required interfaces becomes a concern from a scaling and management perspective. For example, for the SSG Series, a limitation applies to the maximum number of tunnel interfaces that can be configured for the platform. With Junos OS, a limitation applies to the maximum number of logical interface units. To allow for easier management and scalability, Junos OS supports multipoint secure tunnel interfaces with the next-hop tunnel binding (NHTB) feature. This allows a device to bind multiple IPsec security associations (SAs) to a single secure tunnel interface.

By default, the secure tunnel interface operates as a point-to-point type link. For our hub-and-spoke example, the Junos OS-based hub device configures an st0 interface as type multipoint, which is configured in the st0 unit hierarchy. Multipoint configuration is only required on the hub site—the spokes continue to use the default point-to-point mode.

As already mentioned, multiple IPsec VPN tunnels can be bound to a single st0 interface unit. To link a specific destination to a specific IPsec tunnel bound to the same st0 interface, the Junos OS-based device uses two tables: the inet.0 route table and the NHTB table. The Junos OS-based device maps the next-hop IP address specified in the route table entry to a particular VPN tunnel specified in the NHTB table. With this technique, a single st0 interface can support many VPN tunnels.

Using this method, the maximum number of IPsec tunnels is not limited by the number of st0 interfaces that can be created. Limitations are limited by either route table capacity or the maximum number of dedicated IPsec tunnels allowed—whichever is lower. To see the maximum route and tunnel capacities by platform, refer to the relevant product data sheet.

Route-to-Tunnel Mapping

To sort traffic among multiple IPsec VPN tunnels bound to the same st0 interface, the security device maps the next-hop gateway IP address specified in the route to a particular IPsec tunnel name. The mapping of entries in the route table to entries in the NHTB table is shown below. In Figure 1, the local device routes traffic sent from 10.10.10.10 to 10.30.10.10 through the st0.0 interface and through VPN2.

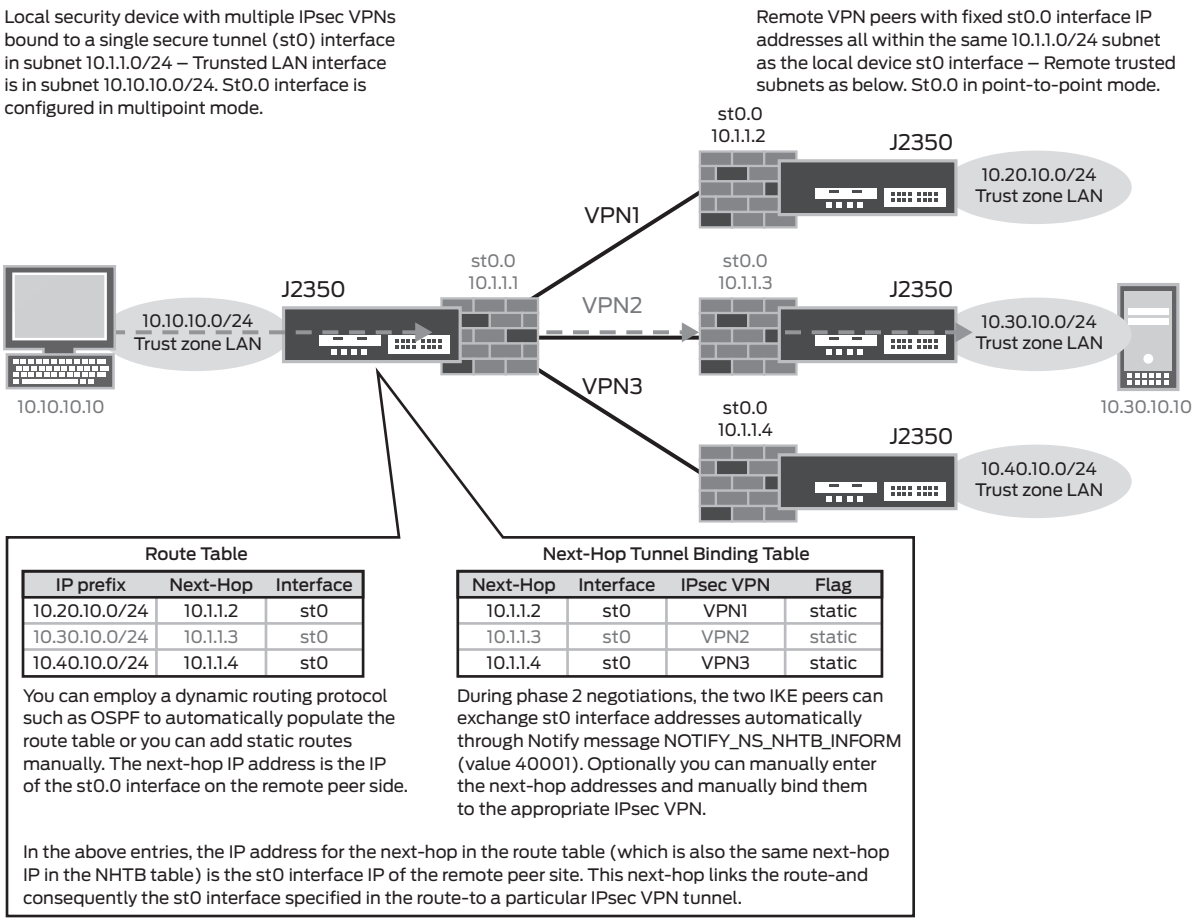

Figure 1: NHTB routes and table entries

The hub device uses the IP address of the remote peer's st0 interface as the next hop. The static route can be entered manually, or a dynamic routing protocol such as OSPF can be allowed to automatically enter the route, referencing the peer's st0 interface IP address as the next hop in the route table. The same IP address must also be entered as the next hop in the NHTB table, along with the appropriate IPsec VPN name. In this way, the route and NHTB tables are linked. Regarding the NHTB table, there are again two options: the next hop can be entered manually, or the Junos OS-based device can be allowed to obtain it automatically from the remote peer during Phase 2 negotiations using the NOTIFY_NS_NHTB_INFORM message. Note that this functionality currently only applies if both peers are Junos OS-based devices.

For the purposes of this application note, we will focus on configuration and verification of a multipoint environment in a hub-and-spoke topology with two spokes. One spoke (Westford) is a Junos OS-based device running software version 8.5, whereas the other spoke (Sunnyvale) is an SSG Series device running Juniper Networks ScreenOS® Software 5.4.0 to outline interoperability requirements (refer to the Network Diagram in Figure 2). Additional spokes are easily added by duplicating the configurations from any existing spokes, changing IP addresses as needed, and adding any additional static routes for new spoke local LANs.

Troubleshooting and configuration details of route-based VPNs, along with other Junos OS-specific application notes, can be found on Juniper Networks' Knowledge Base at **http://kb.juniper.net**. In particular, article KB10182 lists several application notes related to VPN configuration and troubleshooting. For more details on the concepts of NHTB, route-based VPNs, and interface types, please refer to the complete documentation available at **www.juniper.net/techpubs**.

Description and Deployment Scenario

Network Diagram

Refer to Figure 2 below for the network topology used for this configuration example.

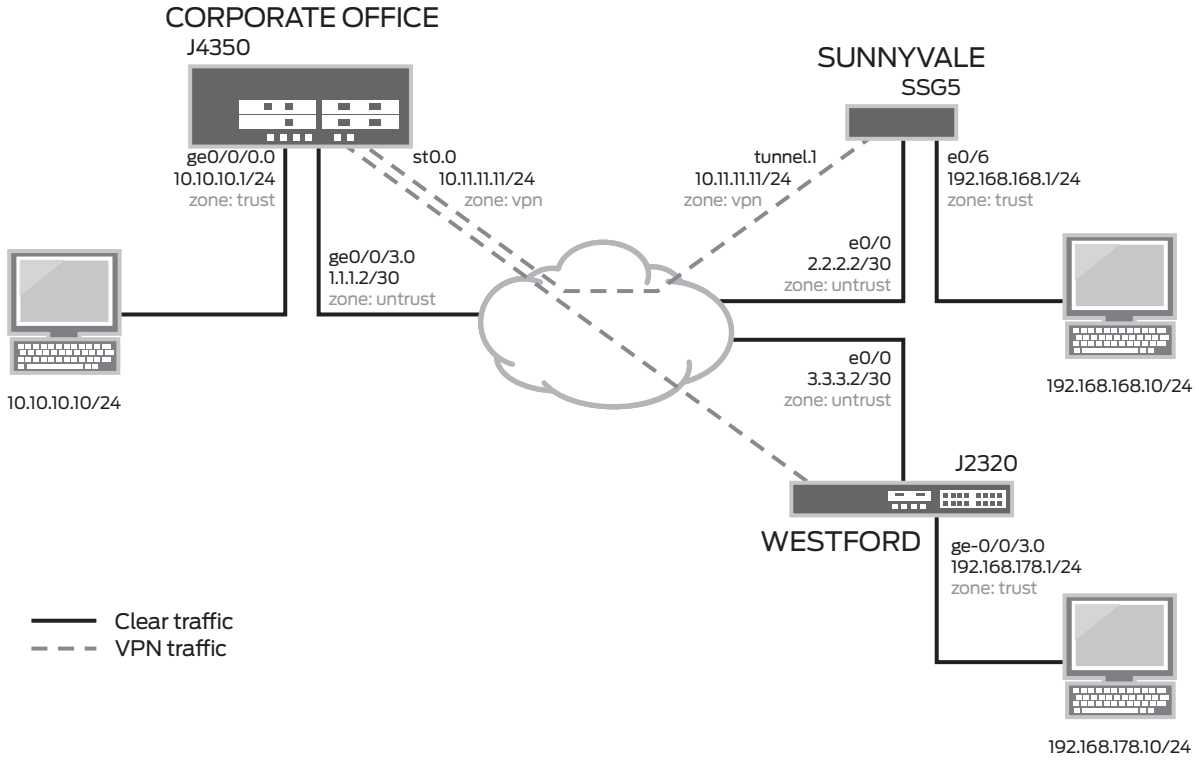

Figure 2: Network topology

Configuration Steps

This example assumes the following:

- Corporate Office internal LAN interface is ge-0/0/0.0 in zone “trust” and will have a private IP subnet. Corporate Office Internet interface is ge-0/0/3.0 in zone “untrust” and will have a public IP.
- Westford Office internal LAN interface is ge-0/0/3.0 in zone “trust” and will have a private IP subnet. Westford Office Internet interface is ge-0/0/0.0 in zone “untrust” and will have a public IP.
- The secure tunnel interface st0.0 for Corporate and Westford is in “vpn” zone to allow for configuring unique policies specifically for tunnel (encrypted) traffic while maintaining unique policies for clear (unencrypted) traffic.
- All st0 interfaces for all peers will have an IP address configured within the same logical subnet. Having all peer tunnel interface IPs within the same logical subnet is recommended but not absolutely required. However, if OSPF is configured with a P2MP link type, this is mandatory.
- Allow all traffic from all remote offices (spokes) to your corporate LAN (hub) and vice versa. Allow all traffic from spoke to spoke. However for one spoke to reach the other, first route traffic through the hub.
- Although a static NHTB entry is not required between Westford and Corporate (they are both Junos OS-based devices), a static NHTB entry is required to Sunnyvale since the SSG Series is not a Junos OS-based device.
- The Juniper Networks SSG5 Secure Services Gateway has already been preconfigured with the correct information from this example.

Basic Steps to Configure on Corporate Office (Hub)

1. Configure the IP addresses for ge-0/0/0.0, ge-0/0/3.0, and st0.0 interfaces.
2. Configure the default route to Internet next hop and also static routes for each remote office LAN. Optionally, a dynamic routing protocol such as OSPF can be used instead, but those details are not included in this application note.
3. Configure security zones and bind the interfaces to the appropriate zones. Also be sure to enable necessary host-inbound services on the interfaces or the zone. For this example, Internet Key Exchange (IKE) service must be enabled on either the ge-0/0/3 interface or the zone "untrust."
4. Configure address book entries for each zone.
5. Configure phase 1 (IKE) gateway settings for both remote offices. For this example, we are using a typical proposal set, however, a different proposal can be created if necessary.
6. Configure phase 2 (IPsec) VPN settings for both remote offices. Optionally, VPN monitor settings can be configured if desired. For this example, we are using standard proposal set and Perfect Forward Secrecy (PFS) group 2, however, a different proposal can be created if necessary.
7. Bind st0.0 interface to the IPsec VPN.
8. Configure st0.0 for multipoint. Configure the NHTB entries for any non-Junos OS spoke sites.

   Note: If establishing a VPN between two devices running Junos OS, it is not necessary to configure NHTB since the hub device is able to obtain the NHTB entry automatically during phase 2 negotiations. However, if the VPN is configured to establish tunnel "on-traffic," the hub site can not initiate the VPN, since without an NHTB entry, the route for that remote peer will not be in an active state. Thus, either the tunnel will always need to be initiated from the spoke, or the hub should have "establish-tunnel" immediately configured.
9. Configure security policies to permit remote office traffic into the Corporate Office LAN and vice versa.
10. Configure outgoing zone "trust" to zone "untrust" permit policy with source Network Address Translation (NAT) for non-encrypted Internet traffic.
11. Configure intra-zone policy in zone "vpn" to allow spokes to communicate with each other. Intra-zone traffic is defined as traffic that ingresses and egresses out of the same zone. By default, intra-zone traffic is denied.
12. Configure tcp-mss for IPsec traffic to eliminate the possibility of fragmented TCP traffic. This will lessen the resource utilization on the device and improve performance.

Basic Steps to Configure on Westford Site (Spoke)

1. Configure the IP addresses for ge-0/0/0.0, ge-0/0/3.0, and st0.0 interfaces.
2. Configure the default route to Internet next hop and also a static route for the Corporate Office LAN.
3. Configure security zones and bind the interfaces to the appropriate zones. Also be sure to enable necessary host-inbound services on the interfaces or the zone. For this example, IKE service must be enabled on either the ge-0/0/0 interface or to zone "untrust."
4. Configure address book entries for each zone.
5. Configure phase 1 (IKE) gateway settings. As noted before, we are using standard proposal set.
6. Configure phase 2 (IPsec) VPN settings. As noted above, a standard proposal set and PFS group 2 is used.
7. Bind st0.0 interface to the IPsec VPN.
8. Configure security policies to permit Westford Office traffic into the Corporate Office LAN and vice versa.
9. Configure outgoing zone "trust" to zone "untrust" permit policy with source NAT for unencrypted Internet traffic.
10. Configure tcp-mss for IPsec traffic to eliminate the possibility of fragmented TCP traffic.

Configuration Example

Corporate Office (Hub)

To begin, enter configuration mode with either the configure or edit command.

Configure IP addresses for private LAN, public Internet, and secure tunnel (st0) interfaces

```
set interfaces ge-0/0/0 unit 0 family inet address 10.10.10.1/24
set interfaces ge-0/0/3 unit 0 family inet address 1.1.1.2/30
set interfaces st0 unit 0 family inet address 10.11.11.10/24
```

Junos OS uses the concept of units for the logical component of an interface. In this example, unit 0 and family inet (IPv4) are used. Though not mandatory, it is recommended for st0 interfaces that all peers have an IP address within the same logical subnet.

Configure default route and routes for tunnel traffic

```
set routing-options static route 0.0.0.0/0 next-hop 1.1.1.1
set routing-options static route 192.168.168.0/24 next-hop 10.11.11.11
set routing-options static route 192.168.178.0/24 next-hop 10.11.11.12
```

For static route, the gateway IP address would typically be specified as the next hop. For route-based VPNs with multipoint, specify the remote peer st0 interface IP as the next hop.

Configure security zones and assign interfaces to the zones

```
set security zones security-zone trust interfaces ge-0/0/0.0
set security zones security-zone untrust interfaces ge-0/0/3.0
set security zones security-zone vpn interfaces st0.0
```

Creating a unique zone for tunnel traffic allows a set of policies to be specifically created for VPN traffic while still maintaining separation of policies for non-VPN traffic. Also, “deny” policies can be created to exclude specific hosts to access the VPN.

Note: If the st0 interface is terminated in the same zone as the trusted LAN, and if a policy exists to allow intra-zone traffic on that zone, no additional security policies are required.

Configure host-inbound services for each zone

```
set security zones security-zone trust host-inbound-traffic system-services all
set security zones security-zone untrust host-inbound-traffic system-services ike
```

Host-inbound services are for traffic destined for the Junos OS-based device. This includes but is not limited to FTP, HTTP, HTTPS, IKE, ping, rlogin, rsh, SNMP, SSH, Telnet, Trivial File Transfer Protocol (TFTP), and traceroute. For this example, we are assuming that we want to allow all such services from zone “trust.” For security reasons, we are only allowing IKE on the Internet-facing zone “untrust” (this is required for IKE negotiations to occur). However, other services such as for management or troubleshooting can also be individually enabled, if required.

Configure address book entries for each zone

```
set security zones security-zone trust address-book address local-net 10.10.10.0/24
set security zones security-zone vpn address-book address sunnyvale-net
192.168.168.0/24
set security zones security-zone vpn address-book address westford-net
192.168.178.0/24
```

For this example, we are using address book object names “local-net,” “sunnyvale-net,” and “westford-net.” Additional address book entries can be added for any additional spokes, as needed.

Configure IKE policy for main mode, predefined standard proposal-set, and pre-shared key

```
set security ike policy ike-policy1 mode main
set security ike policy ike-policy1 proposal-set standard
set security ike policy ike-policy1 pre-shared-key ascii-text “secretkey”
```

For the purposes of this application note, a “standard” proposal set is used, which includes preshared-group2-3des-sha1 and preshared-group2-aes128-sha1 proposals. However, a unique proposal can be created and specified in the IKE policy in accordance with corporate security policy. The same IKE policy can be used for all spoke VPNs, if desired.

Configure spoke IKE gateways (phase 1) with peer IP address, IKE policy, and outgoing interface

```
set security ike gateway westford-gate ike-policy ike-policy1
set security ike gateway westford-gate address 3.3.3.2
set security ike gateway westford-gate external-interface ge-0/0/3.0
set security ike gateway sunnyvale-gate ike-policy ike-policy1
set security ike gateway sunnyvale-gate address 2.2.2.2
set security ike gateway sunnyvale-gate external-interface ge-0/0/3.0
```

A remote IKE peer can be identified by IP address, FQDN/u-FQDN, or ASN1-DN (public key infrastructure certificates). For this example, we are identifying the peer by IP address. Therefore, the gateway address should be the remote peer’s public IP address. It is important also to specify the correct external interface. If either the peer address or external interface specified is incorrect, the IKE gateway will not be properly identified during phase 1 negotiations.

Configure IPsec policy for standard proposal set

```
set security ipsec policy vpn-policy1 proposal-set standard
set security ipsec policy vpn-policy1 perfect-forward-secrecy keys group2
```

As mentioned for phase 1, we are using standard proposal set for the purposes of this application note, which includes esp-group2-3des-sha1 and esp-group2-aes128-sha1 proposals. However, if required, a unique proposal may be created and specified in the IPsec policy.

Configure IPsec VPNs with IKE gateway and IPsec policy, bind to same st0 interface

```
set security ipsec vpn westford-vpn ike gateway westford-gate
set security ipsec vpn westford-vpn ike ipsec-policy vpn-policy1
set security ipsec vpn westford-vpn bind-interface st0.0
set security ipsec vpn sunnyvale-vpn ike gateway sunnyvale-gate
set security ipsec vpn sunnyvale-vpn ike ipsec-policy vpn-policy1
set security ipsec vpn sunnyvale-vpn bind-interface st0.0
```

Binding an st0 interface indicates that this VPN as a route-based VPN. If st0 interface is not specified, phase 2 cannot complete negotiations when this is a route-based VPN.

Configure st0 interface as multipoint, add static NHTB entry for Sunnyvale Office

```
set interfaces st0 unit 0 multipoint
set interfaces st0 unit 0 family inet next-hop-tunnel 10.11.11.11 ipsec-vpn
sunnyvale-vpn
```

As previously mentioned, Sunnyvale site is not a Junos OS-based device. Thus, a static NHTB entry is required. Optionally, a static NHTB entry can also be configured for Westford site, if desired.

Configure security policies for tunnel traffic in both directions for all spokes

```
edit security policies from-zone trust to-zone vpn
 ## Entering zone “trust” to zone “vpn” hierarchy
set policy local-to-spokes match source-address local-net
set policy local-to-spokes match destination-address sunnyvale-net
set policy local-to-spokes match destination-address westford-net
set policy local-to-spokes match application any
set policy local-to-spokes then permit
exit
```

```
edit security policies from-zone vpn to-zone trust
 ## Enter zone "vpn" to zone "trust" hierarchy
set policy spokes-to-local match source-address sunnyvale-net
set policy spokes-to-local match source-address westford-net
set policy spokes-to-local match destination-address local-net
set policy spokes-to-local match application any
set policy spokes-to-local then permit
exit
```

A security policy permits traffic in one direction but also allows all reply traffic without the need for a reverse direction policy. However, since traffic may be initiated from either direction, bidirectional policies are required. More granular policies can be created between zone "vpn" and zone "trust" and can permit or deny accordingly. Note that the policies are regular non-tunnel policies, thus the policies do *not* specify the IPsec profile. Also note that NAT can be enabled on the policies if required, but that is beyond the scope of this application note. If more spoke sites are added, simply add the additional source/destination match entries corresponding to the new spoke local LANs to permit the traffic.

Configure security policy for Internet traffic

```
edit security policies from-zone trust to-zone untrust
 ## Entering from-zone "trust" to-zone "untrust" hierarchy
set policy any-permit match source-address any
set policy any-permit match destination-address any
set policy any-permit match application any
set policy any-permit then permit source-nat interface
exit
```

This policy will permit all traffic from zone "trust" to zone "untrust." By specifying "source-nat interface," the device will translate the source IP and port for outgoing traffic using the IP address of the egress interface as the source IP and a random higher port for the source port. If required, more granular policies can be created to permit/deny certain traffic.

Configure intra-zone policy in vpn zone for spoke-to-spoke traffic

```
edit security policies from-zone vpn to-zone vpn
 ## Entering from-zone "vpn" to-zone "vpn" hierarchy
set policy spoke-to-spoke match source-address any
set policy spoke-to-spoke match destination-address any
set policy spoke-to-spoke match application any
set policy spoke-to-spoke then permit
exit
```

This policy will permit all traffic from zone "vpn" to zone "vpn," which means that this is intra-zone traffic. Without such a policy, all traffic from one spoke to another will be dropped. If required, more granular policies can be created to permit/deny certain IP prefixes or protocols.

Configure tcp-mss to eliminate fragmentation of TCP traffic across tunnel

```
set security flow tcp-mss ipsec-vpn mss 1350
```

Tcp-mss is negotiated as part of the TCP 3-way handshake. It limits the maximum size of a TCP segment to better fit the maximum transmission unit (MTU) limits on a network. This is especially important for VPN traffic, as the IPsec encapsulation, overhead along with the IP and frame overhead, can cause the resulting Encapsulating Security Payload (ESP) packet to exceed the MTU of the physical interface, thus causing fragmentation. Fragmentation increases bandwidth and device resources and is always best avoided. Note that the value of 1350 is a recommended starting point for most Ethernet-based networks with an MTU of 1500 or greater. This value may need to be altered if any device in the path has lower MTU or if there is any added overhead such as Point-to-Point Protocol (PPP), Frame Relay, etc. As a general rule, experiment with different tcp-mss values to obtain optimal performance.

Westford Site (Spoke)

To begin, enter configuration mode with either the “configure” or “edit” command.. Many of the details are the same as with the Corporate Office (Hub) configuration details. Thus for Westford configuration, only differences from the hub site are highlighted.

Configure IP addresses for private LAN, public Internet, and secure tunnel (st0) interfaces

```
set interfaces ge-0/0/0 unit 0 family inet address 3.3.3.2/30
set interfaces ge-0/0/3 unit 0 family inet address 192.168.178.1/24
set interfaces st0 unit 0 family inet address 10.11.11.12/24
```

As previously stated, for st0 interfaces it is recommended (though not mandatory) that all peers have an IP address within the same logical subnet. Thus, Westford st0 interface is within the same subnet as Corporate Office st0 interface.

Configure default route and routes for tunnel traffic

```
set routing-options static route 0.0.0.0/0 next-hop 1.1.1.1
set routing-options static route 10.10.10.0/24 next-hop 10.11.11.10
set routing-options static route 192.168.168.0/24 next-hop 10.11.11.10
```

Since Westford is a spoke site, the st0 interface type is point to point. For next hop, you can specify the hub site st0 interface IP, or you can simply specify st0.0 as the next hop.

Configure security zones and assign interfaces to the zones

```
set security zones security-zone trust interfaces ge-0/0/3.0
set security zones security-zone untrust interfaces ge-0/0/0.0
set security zones security-zone vpn interfaces st0.0
```

All details here are the same as with the Corporate Office (Hub).

Configure host-inbound services for each zone

```
set security zones security-zone trust host-inbound-traffic system-services all
set security zones security-zone untrust host-inbound-traffic system-services ike
```

All details here are the same as with the Corporate Office (Hub).

Configure address book entries for each zone

```
set security zones security-zone trust address-book address local-net
192.168.178.0/24
set security zones security-zone vpn address-book address corp-net 10.10.10.0/24
set security zones security-zone vpn address-book address sunnyvale-net
192.168.168.0/24
```

For this example, we are using address book object names “local-net,” “sunnyvale-net,” and “corp-net.” If additional spokes are added, either more address book entries must be created for each spoke local LAN, or a single address book entry that encompasses all spoke local LANs is required.

Configure IKE policy for main mode, predefined standard proposal-set, and pre-shared key

```
set security ike policy ike-policy1 mode main
set security ike policy ike-policy1 proposal-set standard
set security ike policy ike-policy1 pre-shared-key ascii-text “secretkey”
```

All details here are the same as with the Corporate Office (Hub).

Configure IKE gateway (phase 1) with peer IP address, IKE policy, and outgoing interface

```
set security ike gateway corp-gate address 1.1.1.2
set security ike gateway corp-gate ike-policy ike-policy1
set security ike gateway corp-gate external-interface ge-0/0/0.0
```

All details here are the same as with the Corporate Office (Hub) except the external interface for Westford is ge-0/0/0.0 and the peer address is the Corporate Office public IP address.

Configure IPsec policy for standard proposal set

```
set security ipsec policy vpn-policy1 proposal-set standard
set security ipsec policy vpn-policy1 perfect-forward-secrecy keys group2
```

All details here are the same as with the Corporate Office (Hub).

Configure IPsec VPN with IKE gateway and IPsec policy, bind to st0 interface

```
set security ipsec vpn corp-vpn ike gateway corp-gate
set security ipsec vpn corp-vpn ike ipsec-policy vpn-policy1
set security ipsec vpn corp-vpn bind-interface st0.0
```

All details here are the same as with the Corporate Office (Hub).

Configure security policies for tunnel traffic in both directions

```
edit security policies from-zone trust to-zone vpn
 ## Entering zone “trust” to zone “vpn” hierarchy
set policy to-corp match source-address local-net
set policy to-corp match destination-address corp-net
set policy to-corp match destination-address sunnyvale-net
set policy to-corp match application any
set policy to-corp then permit
exit
edit security policies from-zone vpn to-zone trust
 ## Enter zone “vpn” to zone “trust” hierarchy
set policy from-corp match source-address corp-net
set policy from-corp match source-address sunnyvale-net
set policy from-corp match destination-address local-net
set policy from-corp match application any
set policy from-corp then permit
exit
```

All details here are the same as with the Corporate Office (Hub), except the remote subnets we are interested in are both the Corporate Office local LAN and any other spoke local LANs.

Configure security policy for Internet traffic

```
edit security policies from-zone trust to-zone untrust
 ## Entering from-zone “trust” to-zone “untrust” hierarchy
set policy any-permit match source-address any
set policy any-permit match destination-address any
set policy any-permit match application any
set policy any-permit then permit source-nat interface
exit
```

All details here are the same as with the Corporate Office (Hub).

Configure tcp-mss to eliminate fragmentation of TCP traffic across tunnel

```
set security flow tcp-mss ipsec-vpn mss 1350
```

All details here are the same as with the Corporate Office (Hub).

SSG Series Configuration Example

The focus of this application note is on Junos OS configuration and troubleshooting for multipoint VPNs. For the purpose of completing the diagram above, a sample of relevant configurations is provided from an SSG5 device. However, the concepts with regard to configuration of route-based VPNs for Juniper's firewall/VPN products are well documented in the Concepts and Examples (C&E) guides. Thus, we will not focus on the SSG Series configuration here. For reference, the SSG Series C&E guides can be found at: **www.juniper.net/techpubs/software/screenos/.**

Configuration Example for SSG5

```
set zone name "VPN"
set interface ethernet0/6 zone "Trust"
set interface ethernet0/0 zone "Untrust"
set interface "tunnel.1" zone "VPN"
set interface ethernet0/6 ip 192.168.168.1/24
set interface ethernet0/6 route
set interface ethernet0/0 ip 2.2.2.2/30
set interface ethernet0/0 route
set interface tunnel.1 ip 10.11.11.11/24
set flow tcp-mss 1350
set address "Trust" "sunnyvale-net" 192.168.168.0 255.255.255.0
set address "VPN" "corp-net" 10.10.10.0 255.255.255.0
set address "VPN" "westford-net" 192.168.178.0 255.255.255.0
set ike gateway "corp-ike" address 1.1.1.2 Main outgoing-interface ethernet0/0
preshare "secretkey" sec-level standard
set vpn "corp-vpn" gateway "corp-ike" replay tunnel idletime 0 sec-level standard
set vpn "corp-vpn" bind interface tunnel.1
set policy id 1 from "Trust" to "Untrust" "ANY" "ANY" "ANY" nat src permit
set policy id 2 from "Trust" to "VPN" "sunnyvale-net" "corp-net" "ANY" permit
set policy id 2
set dst-address "westford-net"
exit
set policy id 3 from "VPN" to "Trust" "corp-net" "sunnyvale-net" "ANY" permit
set policy id 3
set src-address "westford-net"
exit
set route 10.10.10.0/24 interface tunnel.1
set route 192.168.178.0/24 interface tunnel.1
set route 0.0.0.0/0 interface ethernet0/0 gateway 2.2.2.1
```

Verifying VPN Connection

Confirm IKE (Phase 1) Status

The first step to confirm VPN status is to check the status of any IKE phase 1 security associations. The command-line interface (CLI) command run on the Corporate Office (Hub) device is shown below:

```
root@CORPORATE> show security ike security-associations
Index   Remote Address  State  Initiator cookie  Responder cookie  Mode
6       3.3.3.2         UP     94906ae2263bbd8e  1c35e4c3fc54d6d3  Main
7       2.2.2.2         UP     7e7a1c0367dfe73c  f284221c656a5fbc  Main
```

We can see that the remote peers are the two spoke sites 3.3.3.2 (Westford) and 2.2.2.2 (Sunnyvale). The state shows UP for both. If the state shows DOWN, or if there are no IKE SAs present, then there is a problem with phase 1 establishment. Confirm that the remote IP address, IKE policy, and external interfaces are all correct. Common errors include incorrect IKE policy parameters such as wrong mode type (aggressive or main), pre-shared keys, or phase 1 proposals (all must match on both peers). Incorrect external interface is another common configuration error. This interface must be the interface that would receive the IKE packets. If configurations have been checked and verified as correct, check the kmd log for any errors or run traceoptions (see troubleshooting section later in this application note).

Note also index numbers for each spoke peer. This value is unique for each IKE security association and allows you to receive more details from that particular SA. Example details for Westford SA index 6 are shown below.

```
root@CORPORATE> show security ike security-associations index 6 detail
IKE peer 3.3.3.2, Index 6,
  Role: Responder, State: UP
  Initiator cookie: 94906ae2263bbd8e, Responder cookie: 1c35e4c3fc54d6d3
  Exchange type: Main, Authentication method: Pre-shared-keys
  Local: 1.1.1.2:500, Remote: 3.3.3.2:500
  Lifetime: Expires in 3571 seconds
  Algorithms:
   Authentication        : sha1
   Encryption            : 3des-cbc
   Pseudo random function: hmac-sha1
  Traffic statistics:
   Input  bytes  :                1128
   Output bytes  :                 988
   Input  packets:                   6
   Output packets:                   5
  Flags: Caller notification sent
  IPSec security associations: 1 created, 0 deleted
  Phase 2 negotiations in progress: 1

    Negotiation type: Quick mode, Role: Responder, Message ID: 1350777248
    Local: 1.1.1.2:500, Remote: 3.3.3.2:500
    Local identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
    Remote identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
    Flags: Caller notification sent, Waiting for done
```

The detail command gives much more information which includes the role (initiator or responder). This is useful to know because troubleshooting is usually always best done on the peer that has the responder role. Also shown are details regarding the authentication and encryption algorithms used, the phase 1 lifetime, and traffic statistics. Traffic statistics can be used to verify that traffic is flowing properly in both directions. Finally, also note the number of IPsec security associations created or in progress. This can help to determine the existence of any completed phase 2 negotiations.

Confirm IPsec (Phase 2) Status

Once IKE phase 1 is confirmed, run the following command to view IPsec (phase 2) security associations.

```
root@CORPORATE> show security ipsec security-associations
  total configured sa: 2
  ID    Gateway          Port  Algorithm       SPI      Life:sec/kb  Mon vsys
  <16384 2.2.2.2         500   ESP:3des/sha1   b2fc36f8 3564/ unlim   -   0
  >16384 2.2.2.2         500   ESP:3des/sha1   5d73929e 3564/ unlim   -   0
  total configured sa: 2
  ID    Gateway          Port  Algorithm       SPI      Life:sec/kb  Mon vsys
  <16385 3.3.3.2         500   ESP:3des/sha1   70f789c6 28756/unlim   -   0
  >16385 3.3.3.2         500   ESP:3des/sha1   80f4126d 28756/unlim   -   0
```

We can see that there are two IPsec SA pairs. Both are using port 500, which means NAT traversal is not used (nat-traversal would show port 4500 or random high port). Also for each SA, we can see the security parameter index (SPI) used for both directions, as well as the lifetime (in seconds) and usage limits or lifesize (in kilobytes). We see “28756/unlim” for 3.3.3.2 (Westford), which means that phase 2 lifetime is set to expire in 28756 seconds and there is no lifesize specified (thus it shows unlimited). Phase 2 lifetime can differ from phase 1 lifetime, since phase 2 is not dependent on phase 1 once the VPN is up. The Mon column refers to VPN monitoring status. If VPN monitoring has been enabled, it will show U (up) or D (down). A hyphen (-) means VPN monitoring is not enabled for this SA. For more details regarding VPN monitoring, refer to the complete documentation for Junos OS. Note that Vsys will always show 0.

Note also the ID number for each SA. This is the index value and is unique for each IPsec security association. View more details for a particular SA by specifying the index value. For example, details for Westford SA index 16385 are as follows:

```
root@CORPORATE> show security ipsec security-associations index 16385 detail
  Virtual-system: Root
  Local Gateway: 1.1.1.2, Remote Gateway: 3.3.3.2
  Local Identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
  Remote Identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
    DF-bit: clear
    Direction: inbound, SPI: 1895270854, AUX-SPI: 0
    Hard lifetime: Expires in 28729 seconds
    Lifesize Remaining:  Unlimited
    Soft lifetime: Expires in 28136 seconds
    Mode: tunnel, Type: dynamic, State: installed, VPN Monitoring: -
    Protocol: ESP, Authentication: hmac-sha1-96, Encryption: 3des-cbc
    Anti-replay service: enabled, Replay window size: 32

    Direction: outbound, SPI: 2163479149, AUX-SPI: 0
    Hard lifetime: Expires in 28729 seconds
    Lifesize Remaining:  Unlimited
    Soft lifetime: Expires in 28136 seconds
    Mode: tunnel, Type: dynamic, State: installed, VPN Monitoring: -
    Protocol: ESP, Authentication: hmac-sha1-96, Encryption: 3des-cbc
    Anti-replay service: enabled, Replay window size: 32
```

Local Identity and Remote Identity are shown above. These elements comprise the proxy ID for this SA. A proxy ID mismatch is a very common reason for a phase 2 failure to complete. If no IPsec SA is listed, confirm the phase 2 proposals, including that the proxy ID settings are correct for both peers. Note that for route-based VPNs, the default proxy ID is local=0.0.0.0/0, remote=0.0.0.0/0, service=any. This can cause issues if there are multiple route-based VPNs from the same peer IP. In this case, specify unique proxy IDs for each IPsec SA. For some third-party vendors, the proxy ID must be manually entered to match. Another common reason for phase 2 failing to complete may be failure to specify st0 interface binding. If IPsec cannot complete, check the kmd log or set traceoptions as detailed in the troubleshooting section of this application note.

Confirm Next-Hop Tunnel Bindings

Once phase 2 is complete for all peers, ensure that routing works properly and confirm that the NHTB table is established correctly. To show the NHTB table, run the following command.

```
root@CORPORATE> show security ipsec next-hop-tunnels
Next-hop gateway  interface   IPSec VPN name                  Flag
10.11.11.11       st0.0       sunnyvale-vpn                   Static
10.11.11.12       st0.0       westford-vpn                    Auto
```

Reference the network topology in Figure 2. The next-hop gateways are the IP addresses for the st0 interfaces of all remote spoke peers. The next hop should be associated with the correct IPsec VPN name. If no NHTB entry exists, there will be no way for the hub device to differentiate which IPsec VPN is associated with which next hop. The flag can have one of two options: static or auto. Static means that the NHTB has been manually configured in the st0.0 interface configurations, which is required if the peer is not a device running Junos OS. Auto means that the NHTB has not been configured, but the entry has been automatically populated into the table during phase 2 negotiations between two Junos OS-based devices.

There will not be an NHTB table on any of the spoke sites in this example. This is because from a spoke perspective, the st0 interface is still a point-to-point link with only one IPsec VPN binding. Thus, the same command shown above will not show any output on Westford Site.

Confirm Static Routes for Remote Peer Local LANs

In order for the NHTB to be used, the static route needs to also reference the spoke peer st0 IP address. Confirm the route to the remote peer using the following operational mode command: show route <dest-ip-prefix>. See the example below.

```
root@CORPORATE> show route 192.168.168.10

inet.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.168.0/24   *[Static/5] 00:08:33
                    > to 10.11.11.11 via st0.0

root@CORPORATE> show route 192.168.178.10

inet.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.178.0/24   *[Static/5] 00:04:04
                    > to 10.11.11.12 via st0.0
```

Note that the next hop is the remote peer st0 IP address and both routes point to st0.0 as the outgoing interface.

Check Statistics and Errors for an IPsec SA

The command shown below is used to check ESP and authentication header (AH) counters, and for any errors with a particular IPsec security association.

```
root@CORPORATE> show security ipsec statistics index 16385
ESP Statistics:
  Encrypted bytes:              920
  Decrypted bytes:             6208
  Encrypted packets:              5
  Decrypted packets:             87
AH Statistics:
  Input bytes:                    0
  Output bytes:                   0
  Input packets:                  0
  Output packets:                 0
Errors:
  AH authentication failures: 0, Replay errors: 0
  ESP authentication failures: 0, ESP decryption failures: 0
  Bad headers: 0, Bad trailers: 0
```

Typically, error values other than zero are not desirable. However, if you are experiencing packet loss issues across a VPN, one approach is to run the command that is shown above multiple times to confirm that the encrypted and decrypted packet counters are incrementing. Also, check to see if any of the error counters increment while you are experiencing the issue. It may also be necessary to enable security flow traceoptions (see troubleshooting section) to see which ESP packets are experiencing errors and why.

Test Traffic Flow Across the VPN

Once you have confirmed the status of IKE phase 1, phase 2, routes, and NHTB entries, the next step is to test traffic flow across the VPN. One way to test traffic flow is through pings. We can ping from local host PC to remote host PC. We can also initiate pings from the Junos OS-based device. An example of ping testing from the Junos OS-based device to the remote PC host on Sunnyvale site is shown below.

```
root@CORPORATE> ping 192.168.168.10 interface ge-0/0/0 count 5
PING 192.168.168.10 (192.168.168.10): 56 data bytes
64 bytes from 192.168.168.10: icmp_seq=0 ttl=127 time=8.287 ms
64 bytes from 192.168.168.10: icmp_seq=1 ttl=127 time=4.119 ms
```

```
64 bytes from 192.168.168.10: icmp_seq=2 ttl=127 time=5.399 ms
64 bytes from 192.168.168.10: icmp_seq=3 ttl=127 time=4.361 ms
64 bytes from 192.168.168.10: icmp_seq=4 ttl=127 time=5.137 ms

--- 192.168.168.10 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 4.119/5.461/8.287/1.490 ms
```

The same can be performed to a host on Westford site to confirm connectivity. Note that when initiating pings from the Junos OS-based device, the source interface must be specified to ensure that route lookup is correct and the appropriate zones can be referenced in policy lookup. In this case, because ge-0/0/0.0 resides in the same security zone as the local host PC, ge-0/0/0 must be specified in pings so that the policy lookup can be from zone “trust” to zone “vpn.”

Likewise, we can initiate a ping from the spoke site host PC to a host on the Corporate Office LAN. Also, we can initiate a ping from the SSG5 as shown below. Test pings from spoke to hub and also spoke to spoke, as shown in the example below.

```
ssg5-> ping 10.10.10.10 from ethernet0/6
Type escape sequence to abort

Sending 5, 100-byte ICMP Echos to 10.10.10.10, timeout is 1 seconds from ethernet0/6
!!!!!
Success Rate is 100 percent (5/5), round-trip time min/avg/max=4/4/5 ms

ssg5-> ping 192.168.178.10 from ethernet0/6
Type escape sequence to abort

Sending 5, 100-byte ICMP Echos to 192.168.178.10, timeout is 1 seconds from
ethernet0/6
!!!!!
Success Rate is 100 percent (5/5), round-trip time min/avg/max=8/8/10 ms
```

If pings fail from either direction, it may indicate an issue with routing, policy, end host, or perhaps an issue with the encryption/decryption of the ESP packets. One way to check this is to view IPsec statistics to see if any errors are reported. Also, you can confirm end host connectivity by pinging from a host on the same subnet as the end host. If it is reachable by other hosts, the issue is most likely not with the end host. For routing and policy issues, enable the security flow traceoptions as detailed below.

Troubleshooting Basics

Basic troubleshooting begins by first isolating the issue and then focusing the debugging efforts on the area where the problem is occurring. One common approach is to start with the lowest layer of the Open Systems Interconnection (OSI) model and work up the OSI stack to confirm at which layer the failure occurs.

Following this methodology, the first step to troubleshooting is to confirm the physical connectivity of the Internet link at the physical and data link levels. Next, using ping, confirm that the Junos OS-based device has connectivity to the Internet next hop, followed by confirming connectivity to the remote IKE peer. Confirm that IKE phase 1 can complete by running the verification commands as shown above. Once phase 1 is confirmed, confirm phase 2. Finally, confirm that traffic is flowing across the VPN. If the VPN is not in UP state, there is very little reason to test any transit traffic across the VPN. Likewise if phase 1 has not been successful, then looking at phase 2 issues will not give us additional useful troubleshooting information.

To troubleshoot issues further at the different levels, configure traceoptions. Traceoptions are enabled in configuration mode and are a part of a Junos OS configuration. This means that a configuration commit is necessary before a trace option will take affect. Likewise, removing traceoptions requires deleting or deactivating the configuration followed by a commit. By enabling a traceoptions flag, the data from the trace option is written to a log file which may be predetermined or manually configured and stored in persistent memory. This means that any trace logs are retained even after a system reboot. Keep in mind the available storage on flash before implementing traceoptions. You can check your available storage as shown on the next page.

```
root@CORPORATE> show system storage
Filesystem              Size       Used      Avail  Capacity   Mounted on
/dev/ad0s1a             213M       136M        75M       65%  /
devfs                   1.0K       1.0K         0B      100%  /dev
devfs                   1.0K       1.0K         0B      100%  /dev/
/dev/md0                144M       144M         0B      100%  /junos
/cf                     213M       136M        75M       65%  /junos/cf
devfs                   1.0K       1.0K         0B      100%  /junos/dev/
procfs                  4.0K       4.0K         0B      100%  /proc
/dev/bo0s1e              24M        13K        24M        0%  /config
/dev/md1                168M       7.3M       147M        5%  /mfs
/dev/md2                 58M        38K        53M        0%  /jail/tmp
/dev/md3                7.7M       108K       7.0M        1%  /jail/var
devfs                   1.0K       1.0K         0B      100%  /jail/dev
/dev/md4                1.9M       6.0K       1.7M        0%  /jail/html/oem
```

As shown above, /dev/ad0s1a represents the onboard flash memory and is currently at 65% capacity. You can also view available storage on Juniper Networks J-Web Software homepage under System Storage. The output of all traceoptions writes to logs stored in directory /var/log. To view a list of all logs in /var/log, run the operational mode command “show log.”

Checking Traceoptions Logs

As noted earlier, enabling traceoptions begins the logging of the output to the filenames specified or to the default log file for the trace option. Go to the appropriate log to view the trace output. The commands to view the appropriate logs are shown as follows:

```
root@CORPORATE> show log kmd
root@CORPORATE> show log security-trace
root@CORPORATE> show log messages
```

Logs can also be uploaded to an FTP server with the “file copy” command. The syntax is as follows: file copy <filename> <destination> as shown below.

```
root@CORPORATE> file copy /var/log/kmd ftp://10.10.10.10/kmd.log
ftp://10.10.10.10/kmd.log                    100% of   35 kB   12 MBps
```

Troubleshooting IKE and IPsec Issues

To view success or failure messages in IKE or IPsec, view the kmd log with the command “show log kmd.” Although the kmd log will give a general reason for any failure, it may be necessary to obtain additional details. For this, we can enable IKE traceoptions. Note that as a general rule, it is always best to troubleshoot on the peer that has the role of responder.

Enable IKE Traceoptions for Phase 1 and Phase 2 Negotiation Issues

Below is an example of all IKE traceoptions.

```
root@CORPORATE> configure
Entering configuration mode

[edit]
root@CORPORATE# edit security ike traceoptions

[edit security ike traceoptions]
root@CORPORATE# set file ?
Possible completions:
  <filename>           Name of file in which to write trace information
  files                Maximum number of trace files (2..1000)
  match                 Regular expression for lines to be logged
  no-world-readable     Don’t allow any user to read the log file
  size                  Maximum trace file size (10240..1073741824)
  world-readable        Allow any user to read the log file
```

```
[edit security ike traceoptions]
root@CORPORATE# set flag ?
Possible completions:
  all                  Trace everything
  certificates         Trace certificate events
  database             Trace security associations database events
  general              Trace general events
  ike                  Trace IKE module processing
  parse                Trace configuration processing
  policy-manager       Trace policy manager processing
  routing-socket       Trace routing socket messages
  timer                Trace internal timer events
```

By default, if no file name is specified, all IKE traceoptions write to the kmd log. However, you can specify a different file name if desired. To write trace data to the log, you must specify at least one flag option. Option "file size" determines the maximum size of a log file in bytes. For example, 1m or 1000000 will generate a maximum file size of 1 MB. Option "file files" determines the maximum number of log files that are generated and stored in flash. Remember to commit the changes to start the trace.

Below is an example of recommended traceoptions for troubleshooting most IKE-related issues.

```
[edit]
root@CORPORATE# edit security ike traceoptions
[edit security ike traceoptions]
root@CORPORATE# set file size 1m
root@CORPORATE# set flag policy-manager
root@CORPORATE# set flag ike
root@CORPORATE# set flag routing-socket
root@CORPORATE# commit
```

Review Kmd Log for Success/Failure Messages

Below are some excerpts of successful phase 1 and phase 2 completions and some failure instances from "show log kmd."

Phase 1 and phase 2 successful

```
Oct  8 10:41:40 Phase-1 [responder] done for local=ipv4(udp:500,[0..3]=1.1.1.2) remo
te=ipv4(udp:500,[0..3]=2.2.2.2)

Oct  8 10:41:51 Phase-2 [responder] done for p1_local=ipv4(udp:500,[0..3]=1.1.1.2)
p1_remote=ipv4(udp:500,[0..3]=2.2.2.2) p2_local=ipv4_subnet(a
ny:0,[0..7]=10.10.10.0/24) p2_remote=ipv4_subnet(any:0,[0..7]=192.168.168.0/24)
```

The local address is 1.1.1.2 and the remote peer is 2.2.2.2. The output "udp:500" indicates that no nat-traversal was negotiated. You should see a phase 1 done message along with the role (initiator or responder). Next you should also see a phase 2 "done" message with proxy ID information. At this point, you can confirm that the IPsec SA is up using the verification commands mentioned previously.

Phase 1 failing to complete, example 1

```
Oct  8 10:31:10 Phase-1 [responder] failed with error(No proposal chosen) for
local=unknown(any:0,[0..0]=) remote=ipv4(any:0,[0..3]=2.2.2.2)

Oct  8 10:31:10 1.1.1.2:500 (Responder) <-> 2.2.2.2:500 { 011359c9 ddef501d -
2216ed2a bfc50f5f [-1] / 0x00000000 } IP; Error = No proposal chosen (14)
```

The local address is 1.1.1.2 and the remote peer is 2.2.2.2. The role is responder. The reason for failing is due to "No proposal chosen." This is likely due to mismatched phase 1 proposals. To resolve this issue, confirm that phase 1 proposals match on both peers.

Phase 1 failing to complete, example 2

```
Oct  8 10:39:40 Unable to find phase-1 policy as remote peer:2.2.2.2 is not recognized.

Oct  8 10:39:40 KMD_PM_P1_POLICY_LOOKUP_FAILURE: Policy lookup for Phase-1
[responder] failed for p1_local=ipv4(any:0,[0..3]=1.1.1.2) p1_remote=ipv4(a
ny:0,[0..3]=2.2.2.2)

Oct  8 10:39:40 1.1.1.2:500 (Responder) <-> 2.2.2.2:500 { 18983055 dbe1d0af -
a4d6d829 f9ed3bba [-1] / 0x00000000 } IP; Error = No proposal chosen (14)
```

The local address is 1.1.1.2 and the remote peer is 2.2.2.2. The role is responder. The reason for failing may seem to indicate No proposal was chosen. However, in this case we also see a message that peer:2.2.2.2 is not recognized. Peer not recognized could be due to an incorrect peer address, mismatched peer ID type, or incorrect peer ID, depending on whether this is a dynamic or static VPN. These must be checked first before the phase 1 proposal is checked. To resolve this issue, confirm that the local peer has the correct peer IP address. Also confirm that the peer is configured with IKE ID type as IP address.

Phase 1 failing to complete, example 3

```
Oct  8 10:36:20 1.1.1.2:500 (Responder) <-> 2.2.2.2:500 { e9211eb9 b59d543c -
766a826d bd1d5ca1 [-1] / 0x00000000 } IP; Invalid next payload type = 17

Oct  8 10:36:20 Phase-1 [responder] failed with error(Invalid payload type) for
local=unknown(any:0,[0..0]=) remote=ipv4(any:0,[0..3]=2.2.2.2)
```

The remote peer is 2.2.2.2. Invalid payload type usually means that there has been a problem with the decryption of the IKE packet due to a pre-shared key mismatch. To resolve this issue, confirm that pre-shared keys match on both peers.

Phase 1 successful, phase 2 failing to complete, example 1

```
Oct  8 10:53:34 Phase-1 [responder] done for local=ipv4(udp:500,[0..3]=1.1.1.2)
remote=ipv4(udp:500,[0..3]=2.2.2.2)

Oct  8 10:53:34 1.1.1.2:500 (Responder) <-> 2.2.2.2:500 { cd9dff36 4888d398 - 6b0d3933
f0bc8e26 [0] / 0x1747248b } QM; Error = No proposal chosen (14)
```

The local address is 1.1.1.2 and the remote peer is 2.2.2.2. We can clearly see that phase 1 was successful based on the "Phase-1 [responder] done" message. The reason for failing is due to No proposal chosen during phase 2 negotiations. The issue is likely a phase 2 proposal mismatch between the two peers. To resolve this issue, confirm that phase 2 proposals match on both peers.

Phase 1 successful, phase 2 failing to complete, example 2

```
Oct  8 10:56:00 Phase-1 [responder] done for local=ipv4(udp:500,[0..3]=1.1.1.2)
remote=ipv4(udp:500,[0..3]=2.2.2.2)

Oct  8 10:56:00 Failed to match the peer proxy ids p2_remote=ipv4_subnet(a
ny:0,[0..7]=192.168.168.0/24) p2_local=ipv4_subnet(any:0,[0..7]=10.10.20.0/24) for
the remote
peer:ipv4(udp:500,[0..3]=2.2.2.2)

Oct  8 10:56:00 KMD_PM_P2_POLICY_LOOKUP_FAILURE: Policy lookup for Phase-2
[responder] failed for p1_local=ipv4(udp:500,[0..3]=1.1.1.2) p1_remote=ipv4(
udp:500,[0..3]=2.2.2.2) p2_local=ipv4_subnet(any:0,[0..7]=10.10.20.0/24) p2_
remote=ipv4_subnet(any:0,[0..7]=192.168.168.0/24)

Oct  8 10:56:00 1.1.1.2:500 (Responder) <-> 2.2.2.2:500 { 41f638eb cc22bbfe -
43fd0e85 b4f619d5 [0] / 0xc77fafcf } QM; Error = No proposal chosen (14)
```

Phase 1 was successful. The reason for failing indicates that No proposal was chosen. We also see the message "Failed

to match the peer proxy ids" which means that the proxy ID did not match what was expected.

We can see that we received a phase 2 proxy ID of (remote=192.168.168.0/24, local=10.10.20.0/24, service=any). This does not match the configurations on the local peer, so the proxy ID match has failed. This results in the error "No proposal chosen." To resolve this issue, configure either peer proxy ID so that it matches the other peer. Note that for a route-based VPN, the proxy ID is all zeroes (local=0.0.0.0/0, remote=0.0.0.0/0, service=any) by default. If the remote peer is specifying a proxy ID other than all zeroes, you must manually configure the proxy ID within the IPsec profile of the peer.

Troubleshooting Flow Issues

If the IPsec tunnel is up but traffic does not appear to be passing through the tunnel, there is probably a problem with the route lookup, security policy, or some other flow issue. Enable security flow traceoptions to learn how Junos OS is handling the traffic, and to determine if there is a problem with routing, policy, or some other flow-related issues.

The details of flow traceoptions output is beyond the scope of this application note. However, such flow trace output information is available in the application note titled: Junos OS Enhanced Services Route-Based VPN Configuration and Troubleshooting.

Note: Enabling flow traceoptions can cause an increase in system CPU and memory utilization. Therefore, enabling flow traceoptions is not recommended during peak traffic load times or if CPU utilization is very high. Enabling packet filters is also highly recommended to lower resource utilizations and to facilitate pinpointing the packets of interest. Finally be sure to delete or deactivate all flow traceoptions and remove any unnecessary log files from flash after completing troubleshooting.

Enabling Security Flow Traceoptions for Routing or Policy Issues

See the following example of flow traceoptions.

```
[edit]
root@CORPORATE# edit security flow traceoptions

[edit security flow traceoptions]
root@CORPORATE# set file ?
Possible completions:
  <filename>           Name of file in which to write trace information
  files                Maximum number of trace files (2..1000)
  match                 Regular expression for lines to be logged
  no-world-readable     Don't allow any user to read the log file
  size                  Maximum trace file size (10240..1073741824)
  world-readable        Allow any user to read the log file

[edit security flow traceoptions]
root@CORPORATE# set flag ?
Possible completions:
  ager                  Ager events
  all                   All events
  basic-datapath        Basic packet flow
  cli                   CLI configuration and commands changes
  errors                Flow errors
  fragmentation         Ip fragmentation and reassembly events
  high-availability     Flow high-availability information
  host-traffic         Flow host-traffic information
  lookup                Flow lookup events
  multicast             Multicast flow information
  packet-drops          Packet drops
  route                 Route information
  session               Session creation and deletion events
  session-scan          Session scan information
  tcp-advanced          Advanced TCP packet flow
```

```
  tcp-basic            TCP packet flow
  tunnel               Tunnel information
```

If no file name is specified, all flow traceoptions output writes to the security trace log by default. However, you can specify a different file name, if desired. To write trace data to the log, you must specify at least one flag option. Option "file size" determines the maximum size of a log file in bytes. For example, 1m or 1000000 will generate a maximum file size of 1 MB. Option "file files" determines the maximum number of log files that are generated and stored in flash. Remember to commit the changes to start the trace.

Junos OS has the ability to configure packet filters to limit the scope of the traffic to be captured by the flow traceoptions. You can filter the output based on source/destination IP, source/destination port, interface, and IP protocol. Up to 64 filters can be configured. Furthermore, a packet filter will also match the reverse direction to capture the reply traffic, assuming the source of the original packet matches the filter. See the following example of flow packet filter options.

```
[edit security flow traceoptions]
root@CORPORATE# set packet-filter <filter-name> ?
Possible completions:
+ apply-groups         Groups from which to inherit configuration data
+ apply-groups-except  Don't inherit configuration data from these groups
  destination-port     Match TCP/UDP destination port
  destination-prefix   Destination IPv4 address prefix
  interface            Logical interface
  protocol             Match IP protocol type
  source-port          Match TCP/UDP source port
  source-prefix        Source IPv4 address prefix
```

Terms listed within the same packet filter act as a Boolean logical AND statement. This means that all statements within the packet filter need to match to write the output to the log. A listing of multiple filter names acts as a logical OR. The following example uses packet filters of traceoptions for troubleshooting traffic flows from Westford to the Corporate Office.

```
[edit]
root@CORPORATE# edit security flow traceoptions

[edit security flow traceoptions]
root@CORPORATE# set file size 1m files 3
root@CORPORATE# set flag basic-datapath
root@CORPORATE# set packet-filter remote-to-local source-prefix 192.168.178.0/24
root@CORPORATE# set packet-filter remote-to-local destination-prefix 10.10.10.0/24
root@CORPORATE# set packet-filter local-to-remote source-prefix 10.10.10.0/24
root@CORPORATE# set packet-filter local-to-remote destination-prefix 192.168.178.0/24
root@CORPORATE# set packet-filter remote-esp protocol 50
root@CORPORATE# set packet-filter remote-esp source-prefix 3.3.3.2/32
root@CORPORATE> commit
```

The following example details the reasoning behind each flow traceoptions setting.

```
[edit security flow traceoptions]
root@CORPORATE# show
file flow-trace-log size 1m files 3;
flag basic-datapath;
```

The log file "security-trace" has been set to 1 MB and up to 3 files are created. This is needed due to the nature of flow traceoptions, where a single file can become full fairly quickly depending on how much traffic is captured. Flag "basic-datapath" will show details for most flow-related problems.

```
packet-filter remote-to-local {
    source-prefix 192.168.168.0/24;
    destination-prefix 10.10.10.0/24;
}
```

The filter shown above captures the decapsulated or unencrypted traffic from remote PC to local PC. Since there are multiple terms, this acts as a Boolean logical AND, meaning that the source IP and destination IP must both match the

filter. If the source IP matches but the destination IP does not, the packet will not be captured. Since packet filters are bidirectional, it is not necessary to configure a filter for the reply traffic.

```
packet-filter local-to-remote {
    source-prefix 10.10.10.0/24;
    destination-prefix 192.168.178.0/24;
}
```

No filter is required for capturing the reply traffic. However, a filter will only capture packets which were originally sourced from the specified side. The “local-to-remote” filter shown above may still be required to capture traffic which sources from local to remote side.

```
packet-filter remote-esp {
    protocol 50;
    source-prefix 3.3.3.2/32;
}
```

The filter shown above is optional and depends on whether or not the previous filter is able to capture any packets. This filter will capture all ESP (IP protocol 50) or encrypted packets from remote peer 2.2.2.2. Note, however, that this last filter will capture *all* encrypted traffic from 2.2.2.2, including packets that perhaps we are not interested in seeing. If the unencrypted traffic is captured, this last filter may not be necessary.

Using the filters shown above, we can troubleshoot any traffic flow issues to and from the Corporate Office and Westford site. Additional filters can be configured for troubleshooting from Westford to Sunnyvale and vice versa. In addition, a single host can be specified with the /32 mask to help narrow the scope and avoid having too much data write to the trace log. Finally, as always, if any assistance is needed in interpreting the data from any of the traceoptions logs, you can contact your regional JTAC (Juniper Technical Assistance Center). To access the JTAC website, go to **www.juniper.net/customers/support/.**

Summary

Juniper Networks Junos operating system provides not only a powerful operating system, but also a rich IP services tool kit. It has enhanced security and VPN capabilities via Juniper’s firewall/IPsec VPN platforms that include Juniper Networks SSG Series Secure Services Gateways.

Appendix A: Show Configuration

The following example displays the “show configuration” command. For reference, traceoptions configurations are highlighted for troubleshooting purposes. Always remember to delete or deactivate traceoptions once troubleshooting is complete.

Corporate Office (Hub)

```
root@CORPORATE> show configuration | no-more

system {
    host-name CORPORATE;
    root-authentication {
        encrypted-password “$1$0wc5IQiB$MTQlktoQ9/nRF10Gntin./”; ## SECRET-DATA
    }
    services {
        ssh;
        web-management {
            http {
                interface ge-0/0/0.0;
            }
        }
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 10.10.10.1/24;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family inet {
                address 1.1.1.2/30;
            }
        }
    }
    st0 {
        unit 0 {
            multipoint;
            family inet {
                next-hop-tunnel 10.11.11.11 ipsec-vpn sunnyvale-vpn;
                address 10.11.11.10/24;
            }
        }
    }
```

```
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 1.1.1.1;
        route 192.168.168.0/24 next-hop 10.11.11.11;
        route 192.168.178.0/24 next-hop 10.11.11.12;
    }
}
security {
    ike {
        traceoptions {
            flag policy-manager;
            flag ike;
            flag routing-socket;
            flag general;
        }
        policy ike-policy1 {
            mode main;
            proposal-set standard;
            pre-shared-key ascii-text "$9$LrN7w2mPQF/t24jqmfn6rev"; ## SECRET-DATA
        }
        gateway sunnyvale-gate {
            ike-policy ike-policy1;
            address 2.2.2.2;
            external-interface ge-0/0/3.0;
        }
        gateway westford-gate {
            ike-policy ike-policy1;
            address 3.3.3.2;
            external-interface ge-0/0/3.0;
        }
    }
    ipsec {
        policy vpn-policy1 {
            perfect-forward-secrecy {
                keys group2;
            }
            proposal-set standard;
        }
        vpn sunnyvale-vpn {
            bind-interface st0.0;
            ike {
                gateway sunnyvale-gate;
                ipsec-policy vpn-policy1;
            }
        }
        vpn westford-vpn {
            bind-interface st0.0;
            ike {
                gateway westford-gate;
                ipsec-policy vpn-policy1;
            }
        }
    }
    zones {
        security-zone trust {
            address-book {
                address local-net 10.10.10.0/24;
            }
```

```
            host-inbound-traffic {
                system-services {
                    all;
                }
            }
            interfaces {
                ge-0/0/0.0;
            }
        }
        security-zone untrust {
            host-inbound-traffic {
                system-services {
                    ike;
                 }
            }
            interfaces {
                ge-0/0/3.0;
            }
        }
        security-zone vpn {
            address-book {
                address sunnyvale-net 192.168.168.0/24;
                address westford-net 192.168.178.0/24;
            }
            interfaces {
                st0.0;
            }
        }
    }
    policies {
        from-zone trust to-zone untrust {
            policy any-permit {
                match {
                    source-address any;
                    destination-address any;
                    application any;
                }
                then {
                    permit {
                        source-nat {
                            interface;
                        }
                    }
                }
            }
        }
        from-zone trust to-zone vpn {
            policy local-to-spokes {
                match {
                    source-address local-net;
                    destination-address [ sunnyvale-net westford-net ];
                    application any;
                }
                then {
                    permit;
                }
            }
        }
        from-zone vpn to-zone trust {
```

```
            policy spokes-to-local {
                match {
                    source-address [ sunnyvale-net westford-net ];
                    destination-address local-net;
                    application any;
                }
                then {
                    permit;
                }
            }
        }
        from-zone vpn to-zone vpn {
            policy spoke-to-spoke {
                match {
                    source-address any;
                    destination-address any;
                    application any;
                }
                then {
                    permit;
                }
            }
        }
    }
    flow {
        tcp-mss {
            ipsec-vpn {
                mss 1350;
            }
        }
    }
}
```

Westford Office (Spoke)

```
root@Westford> show configuration | no-more

system {
    host-name Westford;
    root-authentication {
        encrypted-password “$1$Qk3dVh9X$d3KOf3dhR6uQKhi8FWU.P0”; ## SECRET-DATA
    }
    services {
        web-management {
            http {
                interface ge-0/0/0.0;
            }
        }
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
```

```
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 3.3.3.2/30;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family inet {
                address 192.168.178.1/24;
            }
        }
    }
    st0 {
        unit 0 {
             family inet {
                address 10.11.11.12/24;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 1.1.1.1;
        route 10.10.10.0/24 next-hop 10.11.11.10;
        route 192.168.168.0/24 next-hop 10.11.11.10;
    }
}
security {
    ike {
        traceoptions {
            flag policy-manager;
            flag ike;
            flag routing-socket;
            flag general;
        }
        policy ike-policy1 {
            mode main;
            proposal-set standard;
            pre-shared-key ascii-text "$9$VNsaGF39A0IGDPQFnpu8X7"; ## SECRET-DATA
        }
        gateway corp-gate {
            ike-policy ike-policy1;
            address 1.1.1.2;
            external-interface ge-0/0/0.0;
        }
    }
    ipsec {
        policy vpn-policy1 {
            perfect-forward-secrecy {
                keys group2;
            }
            proposal-set standard;
        }
```

```
    vpn corp-vpn {
        bind-interface st0.0;
        ike {
            gateway corp-gate;
            ipsec-policy vpn-policy1;
        }
    }
}
zones {
    security-zone trust {
        address-book {
            address local-net 192.168.178.0/24;
        }
        host-inbound-traffic {
            system-services {
                all;
            }
        }
        interfaces {
            ge-0/0/3.0 {
            }
        }
    }
    security-zone untrust {
        host-inbound-traffic {
            system-services {
                ike;
            }
        }
        interfaces {
            ge-0/0/0.0 {
            }
        }
    }
    security-zone vpn {
        address-book {
            address corp-net 10.10.10.0/24;
            address sunnyvale-net 192.168.168.0/24;
        }
        interfaces {
            st0.0;
        }
    }
}
policies {
    from-zone trust to-zone untrust {
        policy any-permit {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                permit {
                    source-nat {
                        interface;
                    }
                }
             }
```

```
            }
        }
        from-zone vpn to-zone trust {
            policy from-corp {
                match {
                    source-address [ corp-net sunnyvale-net ];
                    destination-address local-net;
                    application any;
                }
                then {
                    permit;
                }
            }
        }
        from-zone trust to-zone vpn {
            policy to-corp {
                match {
                    source-address local-net;
                    destination-address [ corp-net sunnyvale-net ];
                    application any;
                }
                then {
                    permit;
                }
            }
        }
    }
    flow {
        tcp-mss {
            ipsec-vpn {
                mss 1350;
            }
        }
    }
}
```

About Juniper Networks

Juniper Networks, Inc. is the leader in high-performance networking. Juniper offers a high-performance network infrastructure that creates a responsive and trusted environment for accelerating the deployment of services and applications over a single network. This fuels high-performance businesses. Additional information can be found at **www.juniper.net**.

Corporate and Sales Headquarters
Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or 408.745.2000
Fax: 408.745.2100
www.juniper.net

APAC Headquarters
Juniper Networks (Hong Kong)
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

EMEA Headquarters
Juniper Networks Ireland
Airside Business Park
Swords, County Dublin, Ireland
Phone: 35.31.8903.600
EMEA Sales: 00800.4586.4737
Fax: 35.31.8903.601

To purchase Juniper Networks solutions, please contact your Juniper Networks representative at 1-866-298-6428 or authorized reseller.

3500177-001-EN Dec 2009

Printed on recycled paper