

Chapter 12

IPSec Services Configuration Guidelines

To configure Internet Protocol Security (IPSec) services, you include the following statements at the [edit services ipsec-vpn] hierarchy level of the configuration:

```
[edit services ipsec-vpn]
ike {
  proposal proposal-name {
    authentication-algorithm (md5 | sha1);
    authentication-method (dsa-signatures | pre-shared-keys | rsa-signatures);
    description description;
    dh-group (group1 | group2);
    encryption-algorithm (3des-cbc | des-cbc);
    lifetime-seconds seconds;
  }
  policy policy-name {
    description description;
    local-id {
      ipv4_addr [ values ];
      key_id [ values ];
    }
    mode (aggressive | main);
    pre-shared-key (ascii-text key | hexadecimal key);
    proposals [ proposal-names ];
    remote-id {
      ipv4_addr [ values ];
      key_id [ values ];
    }
  }
}
```

```

ipsec {
  proposal proposal-name {
    authentication-algorithm (hmac-md5-96 | hmac-sha1-96);
    description description;
    encryption-algorithm (3des-cbc | des-cbc);
    lifetime-seconds seconds;
    protocol (ah | esp | bundle);
  }
  policy policy-name {
    description description;
    perfect-forward-secrecy {
      keys (group1 | group2);
    }
    proposals [ proposal-names ];
  }
}
rule rule-name {
  match-direction (input | output);
  term term-name {
    from {
      destination-address address;
      source-address address;
    }
    then {
      backup-remote-gateway address;
      clear-dont-fragment-bit;
      dynamic {
        ike-policy policy-name;
        ipsec-policy policy-name;
      }
      manual {
        direction (inbound | outbound | bidirectional) {
          authentication {
            algorithm (hmac-md5-96 | hmac-sha1-96);
            key (ascii-text key | hexadecimal key);
          }
          auxiliary-spi spi-value;
          encryption {
            algorithm (des-cbc | 3des-cbc);
            key (ascii-text key | hexadecimal key);
          }
          protocol (ah | bundle | esp);
          spi spi-value;
        }
      }
      no-anti-replay;
      remote-gateway address;
      syslog;
    }
  }
}
rule-set rule-set-name {
  [ rule rule-names ];
}

```

This chapter includes the following sections:

Minimum Security Association Configurations on page 113

Configuring Security Associations on page 114

Configuring an IKE Proposal on page 120

Configuring an IKE Policy for Preshared Keys on page 122

Configuring an IPSec Proposal on page 126

Configuring an IPSec Policy on page 128

Configuring IPSec Service Rules on page 131

Example: Configuring IPSec Services on page 136

Minimum Security Association Configurations

The following sections show the minimum configurations necessary to set up security associations (SAs) for IPSec services:

Manual SA Configuration on page 113

Dynamic SA Configuration on page 114

Manual SA Configuration

To define a manual SA configuration, you must include at least the following statements at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
manual {
  direction (inbound | outbound | bidirectional) {
    authentication {
      algorithm (hmac-md5-96 | hmac-sha1-96);
      key (ascii-text key | hexadecimal key);
    }
    encryption {
      algorithm (des-cbc | 3des-cbc);
      key (ascii-text key | hexadecimal key);
    }
    protocol (ah | esp | bundle);
    spi spi-value;
  }
}
```

Dynamic SA Configuration

To define a dynamic SA configuration, you must include at least the following statements at the [edit services ipsec-vpn] hierarchy level:

```
[edit services ipsec-vpn]
ike {
  proposal proposal-name {
    authentication-algorithm (md5 | sha1);
    authentication-method pre-shared-keys;
    dh-group (group1 | group2);
    encryption-algorithm (3des-cbc | des-cbc);
  }
  policy policy-name {
    proposal [ ike-proposal-names ];
    pre-shared-key (ascii-text key | hexadecimal key);
  }
}
ipsec {
  policy policy-name {
    proposals [ ipsec-proposal-names ];
  }
  proposal proposal-name {
    authentication-algorithm (hmac-md5-96 | hmac-sha1-96);
    encryption-algorithm (3des-cbc | des-cbc);
    protocol (ah | esp | bundle);
  }
}
)
```

You must also include the ipsec-policy statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then dynamic] hierarchy level.

Configuring Security Associations

To use IPSec services, you create a security association (SA) between hosts. An SA is a simplex connection that allows two hosts to communicate with each other securely by means of IPSec. You can configure two types of SAs:

Manual—Requires no negotiation; all values, including the keys, are static and specified in the configuration. As a result, each peer must have the same configured options for communication to take place. For information about how to configure a manual SA, see “Configuring Manual Security Associations” on page 115.

Dynamic—Specifies proposals to be negotiated with the tunnel peer. The keys are generated as part of the negotiation and therefore do not need to be specified in the configuration. The dynamic SA includes one or more proposal statements, which allow you to prioritize a list of protocols and algorithms to be negotiated with the peer. For information about how to configure a dynamic SA, see “Configuring Dynamic Security Associations” on page 119.

This section includes the following topics:

Configuring Manual Security Associations on page 115

Configuring Dynamic Security Associations on page 119

Configuring Manual Security Associations

Manual SAs require no negotiation; all values, including the keys, are static and specified in the configuration. As a result, each peer must have the same configured options for communication to take place.

To configure a manual IPSec security association, include statements at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual]
direction (inbound | outbound | bidirectional) {
  authentication {
    algorithm (hmac-md5-96 | hmac-sha1-96);
    key (ascii-text key | hexadecimal key);
  }
  auxiliary-spi auxiliary-spi-value;
  encryption {
    algorithm (des-cbc | 3des-cbc);
    key (ascii-text key | hexadecimal key);
  }
  protocol (ah | esp | bundle);
  spi spi-value;
}
}
```

To configure manual SA statements, do the following:

Configuring Direction on page 115

Configuring the Protocol on page 116

Configuring the Security Parameter Index on page 117

Configuring the Auxiliary Security Parameter Index on page 117

Configuring Authentication on page 118

Configuring Encryption on page 118

Configuring Direction

The direction statement specifies inbound or outbound IPSec processing. If you want to define different algorithms, keys, or security parameter index (SPI) values for each direction, you configure the inbound and outbound options. If you want the same attributes in both directions, use the bidirectional option.

To configure the direction of IPSec processing, include the direction statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual]
manual (inbound | outbound | bidirectional) {
...
}
```

Example: Configuring Inbound and Outbound Direction Statements

Define different algorithms, keys, and security parameter index values for each direction:

```
[edit services ipsec-vpn rule rule-name term term-name then manual]
direction inbound {
  protocol esp;
  spi 16384;
  encryption {
    algorithm 3des-cbc;
    key ascii-text 23456789012345678901234;
  }
direction outbound {
  protocol esp;
  spi 24576;
  encryption {
    algorithm 3des-cbc;
    key ascii-text 12345678901234567890abcd;
  }
}
```

Example: Configuring Bidirectional Statement

Define one set of algorithms, keys, and security parameter index values that is valid in both directions:

```
[edit services ipsec-vpn rule rule-name term term-name then manual]
direction bidirectional {
  protocol ah;
  spi 20001;
  authentication {
    algorithm hmac-md5-96;
    key ascii-text 123456789012abcd;
  }
}
```

Configuring the Protocol

IPSec uses two protocols to protect IP traffic: Encapsulating Security Payload (ESP) and Authentication Header (AH). The AH protocol is used for strong authentication. A third option, bundle, uses AH authentication and ESP encryption; it does not use ESP authentication because AH provides stronger authentication of IP packets.

To configure the IPSec protocol, include the protocol statement and specify ah, esp, or bundle at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual direction *direction*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual direction
direction]
protocol (ah | bundle | esp);
```

Configuring the Security Parameter Index

An SPI is an arbitrary value that uniquely identifies which SA to use at the receiving host. The sending host uses the SPI to identify and select which SA to use to secure every packet. The receiving host uses the SPI to identify and select the encryption algorithm and key used to decrypt packets.



NOTE: Each manual SA must have a unique SPI and protocol combination.

Use the auxiliary SPI when you configure the protocol statement to use the bundle option.

To configure the SPI, include the spi statement and specify a value (from 256 through 16,639) at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual direction *direction*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual direction
direction]
spi spi-value;
```

Configuring the Auxiliary Security Parameter Index

Use the auxiliary SPI when you configure the protocol statement to use the bundle option.



NOTE: Each manual SA must have a unique SPI and protocol combination.

To configure the auxiliary SPI, include the auxiliary-spi statement and specify a value (from 256 through 16,639) at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual direction *direction*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual direction
direction]
auxiliary-spi auxiliary-spi-value;
```

Configuring Authentication

To configure an authentication algorithm, include the authentication statement and specify an authentication algorithm and a key at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual direction *direction*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual direction
direction]
authentication {
  algorithm (hmac-md5-96 | hmac-sha1-96);
  key (ascii-text key | hexadecimal key);
}
```

The algorithm can be one of the following:

hmac-md5-96—Hash algorithm that authenticates packet data. It produces a 128-bit authenticator value and a 96-bit digest.

hmac-sha1-96—Hash algorithm that authenticates packet data. It produces a 160-bit authenticator value and a 96-bit digest.

The key can be one of the following:

ascii-text—ASCII text key. With the hmac-md5-96 option, the key contains 16 ASCII characters. With the hmac-sha1-96 option, the key contains 20 ASCII characters.

hexadecimal—Hexadecimal key. With the hmac-md5-96 option, the key contains 32 hexadecimal characters. With the hmac-sha1-96 option, the key contains 40 hexadecimal characters.

Configuring Encryption

To configure IPSec encryption, include the encryption statement and specify an algorithm and key at the [edit services ipsec-vpn rule *rule-name* term *term-name* then manual direction *direction*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then manual direction
direction]
encryption {
  algorithm (des-cbc | 3des-cbc);
  key (ascii-text key | hexadecimal key);
}
```

The algorithm can be one of the following:

des-cbc—Encryption algorithm that has a block size of 8 bytes; its key size is 64 bits long.

3des-cbc—Encryption algorithm that has a block size of 24 bytes; its key size is 192 bits long.



NOTE: For a list of Data Encryption Standard (DES) encryption algorithm Weak and Semi-Weak keys, see RFC 2409.

For 3des-cbc, the first 8 bytes should differ from the second 8 bytes, and the second 8 bytes should be the same as the third 8 bytes.

The key can be one of the following:

ascii-text—ASCII text key. With the des-cbc option, the key contains 8 ASCII characters. With the 3des-cbc option, the key contains 24 ASCII characters.

hexadecimal—Hexadecimal key. With the des-cbc option, the key contains 16 hexadecimal characters. With the 3des-cbc option, the key contains 48 hexadecimal characters.



NOTE: You cannot configure encryption when you use the Authentication Header protocol.

Configuring Dynamic Security Associations

You configure dynamic SAs with a set of proposals that are negotiated by the security gateways. The keys are generated as part of the negotiation and therefore do not need to be specified in the configuration. The dynamic SA includes one or more proposals, which allow you to prioritize a list of protocols and algorithms to be negotiated with the peer.

To enable a dynamic SA, follow these steps:

1. Configure IKE proposals and IKE policies associated with these proposals.
2. Configure IPSec proposals and an IPSec policy associated with these proposals.
3. Associate an SA with an IPSec policy by configuring the dynamic statement.

For more information about IKE policies and proposals, see “Configuring an IKE Policy for Preshared Keys” on page 122 and “Configuring an IKE Proposal” on page 120. For more information about IPSec policies and proposals, see “Configuring an IPSec Policy” on page 128.

To configure a dynamic SA, include the dynamic statement and specify an IPSec policy name at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level. The ike-policy statement is optional unless you use the preshared key authentication method.

```
[edit services ipsec-vpn rule rule-name term term-name then]
dynamic {
    ike-policy policy-name;
    ipsec-policy policy-name;
}
```



NOTE: If you want to establish a dynamic SA, the attributes in at least one configured IPSec and IKE proposal must match those of its peer.

Configuring an IKE Proposal

Dynamic SAs require Internet Key Exchange (IKE) configuration. With dynamic SAs, you configure IKE first, and then the SA. IKE creates the dynamic SAs and negotiates them for IPSec. The IKE configuration defines the algorithms and keys used to establish the secure IKE connection with the peer security gateway.

You can configure one or more IKE proposals. Each proposal is a list of IKE attributes to protect the IKE connection between the IKE host and its peer.

To configure an IKE proposal, include the proposal statement and specify a name at the [edit services ipsec-vpn ike] hierarchy level:

```
[edit services ipsec-vpn ike]
proposal proposal-name {
    authentication-algorithm (md5 | sha1);
    authentication-method (dsa-signatures | pre-shared-key | rsa-signatures);
    dh-group (group1 | group2);
    encryption-algorithm (3des-cbc | des-cbc);
    lifetime-seconds seconds;
}
```

This section includes the following topics:

Configuring an IKE Authentication Algorithm on page 121

Configuring an IKE Authentication Method on page 121

Configuring an IKE Diffie-Hellman Group on page 121

Configuring an IKE Encryption Algorithm on page 122

Configuring an IKE Lifetime on page 122

Example: Configuring an IKE Proposal on page 122

Configuring an IKE Authentication Algorithm

To configure an IKE authentication algorithm, include the authentication-algorithm statement at the [edit services ipsec-vpn ike proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ike proposal proposal-name]
authentication-algorithm (md5 | sha1);
```

The authentication algorithm can be one of the following:

md5—Produces a 128-bit digest.

sha1—Produces a 160-bit digest.

Configuring an IKE Authentication Method

To configure an IKE authentication method, include the authentication-method statement at the [edit services ipsec-vpn ike proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ike proposal proposal-name]
authentication-method (dsa-signatures | pre-shared-keys | rsa-signatures);
```

The authentication method can be one of the following:

dsa-signatures—Digital Signature Algorithm

pre-shared-keys—Preshared keys

rsa-signatures—RSA signatures

Configuring an IKE Diffie-Hellman Group

Diffie-Hellman is a public-key cryptography scheme that allows two parties to establish a shared secret over an insecure communications channel. It is also used within IKE to establish session keys.

To configure an IKE Diffie-Hellman group, include the dh-group statement at the [edit services ipsec-vpn ike proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ike proposal proposal-name]
dh-group (group1 | group2);
```

The group can be one of the following:

group1—Specifies that IKE use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.

group2—Specifies that IKE use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.

group2 provides more security but requires more processing time.

Configuring an IKE Encryption Algorithm

To configure an IKE encryption algorithm, include the encryption-algorithm statement at the [edit services ipsec-vpn ike proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ike proposal proposal-name]  
  encryption-algorithm (3des-cbc | des-cbc);
```

The encryption algorithm can be one of the following:

3des-cbc—Encryption algorithm that has a key size of 24 bytes; its key size is 192 bits long.

des-cbc—Encryption algorithm that has a key size of 8 bytes; its key size is 56 bits long.

Configuring an IKE Lifetime

The lifetime-seconds statement sets the lifetime of an IKE SA. When the IKE SA expires, it is replaced by a new SA (and SPI) or terminated.

To configure the IKE SA lifetime, include the lifetime-seconds statement at the [edit services ipsec-vpn ike proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ike proposal proposal-name]  
  lifetime-seconds seconds;
```

By default, the IKE SA lifetime is 3600 seconds. The range is from 180 through 86,400 seconds.

Example: Configuring an IKE Proposal

Configure an IKE proposal:

```
[edit services ipsec-vpn ike]  
  proposal ike-proposal {  
    authentication-method pre-shared-keys;  
    dh-group group1;  
    authentication-algorithm sha1;  
    encryption-algorithm 3des-cbc;  
  }
```

Configuring an IKE Policy for Preshared Keys

An IKE policy defines a combination of security parameters (IKE proposals) to be used during IKE negotiation. It defines a peer address, the preshared key for the given peer, and the proposals needed for that connection. During the IKE negotiation, IKE looks for an IKE policy that is the same on both peers. The peer that initiates the negotiation sends all its policies to the remote peer, and the remote peer tries to find a match.

A match is made when both policies from the two peers have a proposal that contains the same configured attributes. If the lifetimes are not identical, the shorter lifetime between the two policies (from the host and peer) is used. The configured preshared key must also match its peer.

You can create multiple, prioritized proposals at each peer to ensure that at least one proposal matches a remote peer's proposal.

First, you configure one or more IKE proposals; then you associate these proposals with an IKE policy. You can also prioritize a list of proposals used by IKE in the policy statement by listing the proposals you want to use, from first to last.

To configure an IKE policy, include the policy statement and specify a policy name at the [edit services ipsec-vpn ike] hierarchy level:

```
[edit services ipsec-vpn ike]
policy policy-name {
  description description;
  local-id {
    ipv4_addr [ values ];
    key_id [ values ];
  }
  mode (aggressive | main);
  pre-shared-key (ascii-text key | hexadecimal key);
  proposals [ proposal-names ];
  remote-id {
    ipv4_addr [ values ];
    key_id [ values ];
  }
}
```

This section includes the following topics:

Configuring the IKE Policy Mode on page 123

Configuring IKE Policy Proposals on page 124

Configuring an IKE Policy Preshared Key on page 124

Configuring an IKE Policy Description on page 124

Configuring Local and Remote IDs on page 124

For an example of an IKE policy configuration, see “Example: Configuring an IKE Policy” on page 125.

Configuring the IKE Policy Mode

IKE policy has two modes: aggressive and main. By default, main mode is enabled. Main mode uses six messages, in three exchanges, to establish the IKE SA. (These three steps are IKE SA negotiation, a Diffie-Hellman exchange, and authentication of the peer.) Main mode also allows a peer to hide its identity.

Aggressive mode also establishes an authenticated IKE SA and keys. However, aggressive mode uses half the number of messages, has less negotiation power, and does not provide identity protection. The peer can use the aggressive or main mode to start IKE negotiation; the remote peer accepts the mode sent by the peer.

To configure IKE policy mode, include the mode statement and specify aggressive or main at the [edit services ipsec-vpn ike policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
mode (aggressive | main);
```

Configuring IKE Policy Proposals

The IKE policy includes a list of one or more proposals associated with an IKE policy.

To configure an IKE policy proposal, include the proposals statement and specify one or more proposal names at the [edit services ipsec-vpn ike policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
proposals [ proposal-names ];
```

Configuring an IKE Policy Preshared Key

IKE policy preshared keys authenticate peers. You must manually configure a preshared key, which must match that of its peer. The preshared key can be an ASCII text (alphanumeric) key or a hexadecimal key.

To configure an IKE policy preshared key, include the pre-shared-key statement and a key at the [edit services ipsec-vpn ike policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
pre-shared-key (ascii-text key | hexadecimal key);
```

Configuring an IKE Policy Description

To specify a description for an IKE policy, include the description statement at the [edit services ipsec-vpn ike policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
description description;
```

The text description is optional.

Configuring Local and Remote IDs

You can optionally specify local identifiers for use in IKE phase 1 negotiation. If the local-id statement is omitted, the local gateway address is used.

To specify one or more local IDs, include the local-id statement at the [edit services ipsec-vpn ike policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
local-id {
    ipv4_addr [ values ];
    key_id [ values ];
}
```

You can also specify remote gateway identifiers for which the IKE policy is used. The remote gateway address in which this policy is defined is added by default.

To specify one or more remote IDs, include the `remote-id` statement at the `[edit services ipsec-vpn ike policy policy-name]` hierarchy level:

```
[edit services ipsec-vpn ike policy policy-name]
remote-id {
  ipv4_addr [ values ];
  key_id [ values ];
}
```

Example: Configuring an IKE Policy

Define two IKE policies: policy 10.1.1.2 and policy 10.1.1.1. Each policy is associated with proposal-1 and proposal-2.

```
[edit services ipsec-vpn]
ike {
  proposal proposal-1 {
    authentication-method pre-shared-keys;
    dh-group group1;
    authentication-algorithm sha1;
    encryption-algorithm 3des-cbc;
    lifetime-seconds 1000;
  }
  proposal proposal-2 {
    authentication-method pre-shared-keys;
    dh-group group2;
    authentication-algorithm md5;
    encryption-algorithm des-cbc;
    lifetime-seconds 10000;
  }
  proposal proposal-3 {
    authentication-method rsa-signatures;
    dh-group group2;
    authentication-algorithm md5;
    encryption-algorithm des-cbc;
    lifetime-seconds 10000;
  }
  policy 10.1.1.2 {
    mode main;
    proposals [ proposal-1 proposal-2 ];
    pre-shared-key ascii-text example-pre-shared-key;
  }
  policy 10.1.1.1 {
    local-certificate certificate-file-name;
    local-key-pair private-public-key-file;
    mode aggressive;
    proposals [ proposal-2 proposal-3 ];
    pre-shared-key hexadecimal 0102030abbcd;
  }
}
```



NOTE: Updates to the current IKE proposal and policy configuration are not applied to the current IKE SA; updates are applied to new IKE SAs.

If you want the new updates to take immediate effect, you must clear the existing IKE security associations so that they will be re-established with the changed configuration. For information about how to clear the current IKE security association, see the *JUNOS Protocols, Class of Service, and System Basics Command Reference*.

Configuring an IPsec Proposal

An IPsec proposal lists protocols and algorithms (security services) to be negotiated with the remote IPsec peer.

To configure an IPsec proposal, include the proposal statement and specify an IPsec proposal name at the [edit services ipsec-vpn ipsec] hierarchy level:

```
[edit services ipsec-vpn ipsec]
proposal proposal-name {
  authentication-algorithm (hmac-md5-96 | hmac-sha1-96);
  description description;
  encryption-algorithm (3des-cbc | des-cbc);
  lifetime-seconds seconds;
  protocol (ah | esp | bundle);
}
```

This section discusses the following topics:

Configuring an Authentication Algorithm on page 126

Configuring an IPsec Proposal Description on page 127

Configuring an Encryption Algorithm on page 127

Configuring the IPsec Lifetime on page 127

Configuring the Protocol for the Dynamic SA on page 128

Configuring an Authentication Algorithm

To configure an IPsec authentication algorithm, include the authentication-algorithm statement at the [edit services ipsec-vpn ipsec proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec proposal proposal-name]
authentication-algorithm (hmac-md5-96 | hmac-sha1-96);
```

The authentication algorithm can be one of the following:

hmac-md5-96—Hash algorithm that authenticates packet data. It produces a 128-bit digest. Only 96 bits are used for authentication.

hmac-sha1-96—Hash algorithm that authenticates packet data. It produces a 160-bit digest. Only 96 bits are used for authentication.

Configuring an IPSec Proposal Description

To specify a description for an IPSec proposal, include the description statement at the [edit services ipsec-vpn ipsec proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec proposal proposal-name]
description description;
```

The text description is optional.

Configuring an Encryption Algorithm

To configure an IPSec encryption algorithm, include the encryption-algorithm statement at the [edit services ipsec-vpn ipsec proposal *proposal-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec proposal proposal-name]
encryption-algorithm (3des-cbc | des-cbc);
```

The encryption algorithm can be one of the following:

3des-cbc—Encryption algorithm that has a block size of 24 bytes; its key size is 192 bits long.

des-cbc—Encryption algorithm that has a block size of 8 bytes; its key size is 48 bits long.



NOTE: We recommend that you use the 3DES-CBC encryption algorithm.

Configuring the IPSec Lifetime

The IPSec lifetime-seconds statement sets the lifetime of an IPSec SA. When the IPSec SA expires, it is replaced by a new SA (and SPI) or terminated. If you do not configure a lifetime and a lifetime is not sent by a responder, the default lifetime is 28,800 seconds.

To configure the IPSec lifetime, include the `lifetime-seconds` statement and specify the number of seconds (from 180 through 86,400) at the `[edit services ipsec-vpn ipsec proposal proposal-name]` hierarchy level:

```
[edit services ipsec-vpn ipsec proposal proposal-name]
lifetime-seconds seconds;
```



NOTE: When a dynamic SA is created, two types of lifetimes are used: hard and soft. The hard lifetime specifies the lifetime of the SA. The soft lifetime, which is derived from the hard lifetime, informs the IPSec key management system that the SA is about to expire. This allows the key management system to negotiate a new SA before the hard lifetime expires.

Configuring the Protocol for the Dynamic SA

The protocol statement sets the protocol for a dynamic SA. IPSec uses two protocols to protect IP traffic: Encapsulating Security Payload (ESP) and Authentication Header (AH). The ESP protocol can support authentication, encryption, or both. The AH protocol is used for strong authentication. AH also authenticates the IP packet. The bundle option uses AH authentication and ESP encryption; it does not use ESP authentication because AH provides stronger authentication of IP packets.

To configure the protocol for a dynamic SA, include the protocol statement and specify the `ah`, `esp`, or `bundle` option at the `[edit services ipsec-vpn ipsec proposal proposal-name]` hierarchy level:

```
[edit services ipsec-vpn ipsec proposal proposal-name]
protocol (ah | esp | bundle);
```

Configuring an IPSec Policy

An IPSec policy defines a combination of security parameters (IPSec proposals) used during IPSec negotiation. It defines Perfect Forward Secrecy (PFS) and the proposals needed for the connection. During the IPSec negotiation, IPSec looks for a proposal that is the same on both peers. The peer that initiates the negotiation sends all its policies to the remote peer, and the remote peer tries to find a match.

A match is made when both policies from the two peers have a proposal that contains the same configured attributes. If the lifetimes are not identical, the shorter lifetime between the two policies (from the host and peer) is used.

You can create multiple, prioritized IPSec proposals at each peer to ensure that at least one proposal matches a remote peer's proposal.

First, you configure one or more IPSec proposals; then you associate these proposals with an IPSec policy. You can prioritize a list of proposals used by IPSec in the policy statement by listing the proposals you want to use, from first to last.

To configure an IPSec policy, include the policy statement, and specify the policy name and one or more proposals you want to associate with this policy at the [edit services ipsec-vpn ipsec] hierarchy level:

```
[edit services ipsec-vpn ipsec]
policy policy-name {
  description description;
  perfect-forward-security {
    keys (group1 | group2);
  }
  proposals [ proposal-names ];
}
```

This section includes the following topics related to configuring an IPSec policy:

Configuring an IPSec Policy Description on page 129

Configuring Perfect Forward Secrecy on page 129

Configuring IPSec Policy Proposals on page 130

Example: IPSec Policy Configuration on page 130

Configuring an IPSec Policy Description

To specify a description for an IPSec policy, include the description statement at the [edit services ipsec-vpn ipsec policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec policy policy-name]
description description;
```

The text description is optional.

Configuring Perfect Forward Secrecy

PFS provides additional security by means of a Diffie-Hellman shared secret value. With PFS, if one key is compromised, previous and subsequent keys are secure because they are not derived from previous keys. This statement is optional.

To configure PFS, include the perfect-forward-security statement and specify a Diffie-Hellman group at the [edit services ipsec-vpn ipsec policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec policy policy-name]
perfect-forward-security {
  keys (group1 | group2);
}
```

The key can be one of the following:

group1—Specifies that IKE use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.

group2—Specifies that IKE use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.

group2 provides more security than group1, but requires more processing time.

Configuring IPsec Policy Proposals

The IPsec policy includes a list of one or more proposals associated with an IPsec policy.

To configure an IPsec policy proposal, include the proposals statement and specify one or more proposal names at the [edit services ipsec-vpn ipsec policy *policy-name*] hierarchy level:

```
[edit services ipsec-vpn ipsec policy policy-name]
proposals [ proposal-names ];
```

Example: IPsec Policy Configuration

Define an IPsec policy, dynamic policy-1, that is associated with two proposals (dynamic-1 and dynamic-2):

```
[edit services ipsec-vpn ipsec]
proposal dynamic-1 {
  protocol esp;
  authentication-algorithm hmac-md5-96;
  encryption-algorithm 3des-cbc;
  lifetime-seconds 6000;
}
proposal dynamic-2 {
  protocol esp;
  authentication-algorithm hmac-sha1-96;
  encryption-algorithm 3des-cbc;
  lifetime-seconds 6000;
}
policy dynamic-policy-1 {
  perfect-forward-secrecy {
    keys group1;
  }
  proposals [ dynamic-1 dynamic-2 ];
}
```



NOTE: Updates to the current IPSec proposal and policy configuration are not applied to the current IPSec SA; updates are applied to new IPSec SAs.

If you want the new updates to take immediate effect, you must clear the existing IPSec security associations so that they will be reestablished with the changed configuration. For information about how to clear the current IPSec security association, see the *JUNOS Protocols, Class of Service, and System Basics Command Reference*.

Configuring IPSec Service Rules

This section describes the following tasks for configuring IPSec services:

Configuring the IPSec Rule Set on page 131

Configuring IPSec Rule Content on page 132

Configuring the IPSec Rule Set

The rule-set statement defines a collection of Network Address Translation (NAT) rules that determine what actions the router software performs on packets in the data stream. You define each rule by specifying a rule name and configuring terms. Then, you specify the order of the rules by including the rule-set statement at the [edit services ipsec-vpn] hierarchy level:

```
[edit services ipsec-vpn]
rule-set rule-set-name {
  rule rule-name1;
  rule rule-name2;
  rule rule-name3;
  ...
}
```

The router software processes the rules in the order in which you specify them in the configuration. If a term in a rule matches the packet, the router performs the corresponding action and the rule processing stops. If no term in a rule matches the packet, processing continues to the next rule in the rule set. If none of the rules matches the packet, the packet is dropped by default.

Configuring IPSec Rule Content

To configure an IPSec rule, include the rule *rule-name* statement at the [edit services ipsec-vpn] hierarchy level:

```
[edit services ipsec-vpn]
rule rule-name {
  match-direction (input | output);
  term term-name {
    from {
      destination-address address;
      source-address address;
    }
    then {
      backup-remote-gateway address;
      clear-dont-fragment-bit;
      dynamic {
        ike-policy policy-name;
        ipsec-policy policy-name;
      }
      manual {
        direction (inbound | outbound | bidirectional) {
          authentication {
            algorithm (hmac-md5-96 | hmac-sha1-96);
            key (ascii-text key | hexadecimal key);
          }
          auxiliary-spi spi-value;
          encryption {
            algorithm (des-cbc | 3des-cbc);
            key (ascii-text key | hexadecimal key);
          }
          protocol (ah | bundle | esp);
          spi spi-value;
        }
      }
      no-anti-replay;
      remote-gateway address;
      syslog;
    }
  }
}
```

Each IPSec rule consists of a set of terms, similar to a firewall filter. A term consists of the following:

from statement—Specifies the match conditions and applications that are included and excluded.

then statement—Specifies the actions and action modifiers to be performed by the router software.

In addition, each rule includes a match-direction statement that specifies the direction in which the match is applied. To configure where the match is applied, include the match-direction (input | output) statement at the [edit services ipsec-vpn rule *rule-name*] hierarchy level:

```
[edit services ipsec-vpn]
rule rule-name {
    match-direction (input | output);
}
```

Configuring IPSec Match Conditions

To configure NAT match conditions, include the from statement at the [edit services ipsec-vpn rule *rule-name* term *term-name*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name]
from {
    destination-address address;
    source-address address;
}
```

You can use either the source address or the destination address as a match condition, in the same way that you would configure a firewall filter; for more information, see the *JUNOS Policy Framework Configuration Guide*.

Configuring IPSec Actions

To configure IPSec actions, include the then statement at the [edit services ipsec-vpn rule *rule-name* term *term-name*] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name]
then {
    backup-remote-gateway address;
    clear-dont-fragment-bit;
    dynamic {
        ike-policy policy-name;
        ipsec-policy policy-name;
    }
    manual {
        direction (inbound | outbound | bidirectional) {
            authentication {
                algorithm (hmac-md5-96 | hmac-sha1-96);
                key (ascii-text key | hexadecimal key);
            }
            auxiliary-spi spi-value;
            encryption {
                algorithm (des-cbc | 3des-cbc);
                key (ascii-text key | hexadecimal key);
            }
            protocol (ah | bundle | esp);
            spi spi-value;
        }
    }
    no-anti-replay;
    remote-gateway address;
    syslog;
}
```

The principal IPsec actions are to configure a dynamic or manual SA:

You configure a dynamic SA by including the dynamic statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level and referencing policies you have configured at the [edit services ipsec-vpn ipsec] and [edit services ipsec-vpn ike] hierarchy levels; for more information, see “Configuring Dynamic Security Associations” on page 119.

You configure a manual SA by including the manual statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level; for more information, see “Configuring Manual Security Associations” on page 115.

You can configure the following additional properties:

Enabling Packet Fragmentation on page 134

Configuring the Remote Address and Backup Remote Address on page 134

Disabling Anti-Replay on page 135

Enabling System Log Messages on page 135

Enabling Packet Fragmentation

To enable fragmentation of Internet Protocol version 4 (IPv4) packets in IPsec tunnels, include the clear-dont-fragment-bit statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
clear-dont-fragment-bit;
```

Setting the clear-dont-fragment-bit statement clears the Don't Fragment (DF) bit in the packet header, regardless of the packet size. If the packet size exceeds the tunnel maximum transmission unit (MTU) value, the packet is fragmented before encapsulation. For IPsec tunnels, the default MTU value is 1500 regardless of the interface MTU setting.

Configuring the Remote Address and Backup Remote Address

To specify the remote address to which the IPsec traffic is directed, include the remote-gateway statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
remote-gateway address;
```

To specify a backup remote address, include the backup-remote-gateway statement at the [edit services ipsec-vpn rule *rule-name* term *term-name* then] hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
backup-remote-gateway address;
```

Configuring the `backup-remote-gateway` statement enables the dead peer detection (DPD) protocol, which monitors the tunnel state and remote peer availability. When the primary tunnel defined by the `remote-gateway` statement is active, the backup tunnel is in standby mode. If the DPD protocol determines that the primary remote gateway address is no longer reachable, a new tunnel is established to the backup address.

If there is no incoming traffic from a peer during a defined interval of 10 seconds, the router detects a tunnel as inactive. A global timer polls all tunnels every 10 seconds and the AS PIC sends a message listing any inactive tunnels. If a tunnel becomes inactive, the router takes the following steps to failover to the backup address:

1. The AS PIC message triggers the DPD protocol to send a hello message to the peer.
2. If no acknowledgment is received, two retries are sent at 2-second intervals, and then the tunnel is declared dead.
3. Failover takes place if the tunnel is declared dead or there is an IPSec Phase 1 negotiation timeout. The primary tunnel is put in standby mode and the backup becomes active.
4. If the negotiation to the backup tunnel times out, the router switches back to the primary tunnel. If both peers are down, it tries the failover six times. It then stops failing over and reverts to the original configuration, with the primary tunnel active and the backup in standby mode.

For more information on the DPD protocol, see RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*.

Disabling Anti-Replay

To disable the IPSec anti-replay feature, include the `no-anti-replay` statement at the `[edit services ipsec-vpn rule rule-name term term-name then]` hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
no-anti-replay;
```

By default, anti-replay service is enabled. Occasionally this can cause interoperability issues with other vendors' equipment.

Enabling System Log Messages

To record an alert in the system logging facility, include the `syslog` statement at the `[edit services ipsec-vpn rule rule-name term term-name then]` hierarchy level:

```
[edit services ipsec-vpn rule rule-name term term-name then]
syslog;
```

Example: Configuring IPSec Services

Following is the configuration of the provider edge (PE) router, demonstrating the usage of next-hop service sets and dynamic SA configuration:

```
[edit interfaces]
so-0/0/0 {
  no-keepalives;
  encapsulation cisco-hdlc;
  unit 0 {
    family inet {
      address 6.6.6.6/32;
    }
  }
}
so-2/2/0 {
  description "teller so-0/2/0";
  no-keepalives;
  encapsulation cisco-hdlc;
  unit 0 {
    family inet {
      address 21.21.1.1/16;
    }
  }
}
sp-3/1/0 {
  unit 0 {
    family inet {
      address 7.7.7.7/32;
    }
  }
  unit 1 {
    family inet;
    service-domain inside;
  }
  unit 2 {
    family inet;
    service-domain outside;
  }
}

[edit policy-options]
policy-statement vpn-export {
  then {
    community add vpn-comm;
    accept;
  }
}
policy-statement vpn-import {
  term a {
    from community vpn-comm;
    then accept;
  }
}
community vpn-comm members target:100:20;
```

```

[edit routing-instances]
vrf {
  instance-type vrf;
  interface sp-3/1/0.1;  # Inside sp interface
  interface so-0/0/0.0;
  route-distinguisher 192.168.0.1:1;
  vrf-import vpn-import;
  vrf-export vpn-export;
  routing-options {
    static {
      route 0.0.0.0/0 next-hop so-0/0/0.0;
      route 11.11.11.1/32 next-hop so-0/0/0.0;
      route 8.8.8.1/32 next-hop sp-3/1/0.1;
    }
  }
}

[edit services]
ipsec-vpn {
  rule rule-1 {
    term term-1 {
      then {
        remote-gateway 21.21.2.1;
        dynamic {
          ike-policy ike-policy;
        }
      }
    }
  }
  match-direction input;
}
ike {
  policy ike-policy {
    pre-shared-key ascii-text "$9$ExmcSeMWxdVYBI";
  }
}
}
service-set service-set-1 {
  ipsec-vpn {
    local-gateway 21.21.1.1;
  }
  ipsec-vpn-rules rule-1;
  next-hop-service {
    inside-service-interface sp-3/1/0.1;
    outside-service-interface sp-3/1/0.2;
  }
}
}

```

