Chapter 20: Securing DMVPN Tunnels

Instructor Materials

CCNP Enterprise: Advanced Routing

Chapter 20 Content

This chapter covers the following content:

- **Elements of Secure Transport -** This section explains the need for data integrity, data confidentiality, and data availability.
- **IPsec Fundamentals -** This section explains the core concepts involved with IP security encryption.
- **IPsec Tunnel Protection -** This section explains how IPsec protection integrates with DMVPN tunnels.

Elements of Secure Transport

- A properly designed network provides data confidentiality, integrity, and availability.
- Without these components, a business might lose potential customers if the customers do not think that their information is secure.

Data Terms

- **Data confidentiality** - Ensuring that data is viewable only by authorized users. Data confidentiality is maintained through encryption.
- **Data integrity** - Ensuring that data is modified only by authorized users. Information is valuable only if it is accurate. Inaccurate data can result in an unanticipated cost. Data integrity is maintained by using an encrypted digital signature, which is typically a checksum.
- **Data availability** - Ensuring that the network is always available allows for the secure transport of the data. Redundancy and proper design ensure data availability.

Typical WAN Network

Figure 20-1 shows the traditional approach to securing data on a network. The entire controlled infrastructure (enterprise and SP) is assumed to be safe. Traffic is encrypted only when exposed to the public internet.

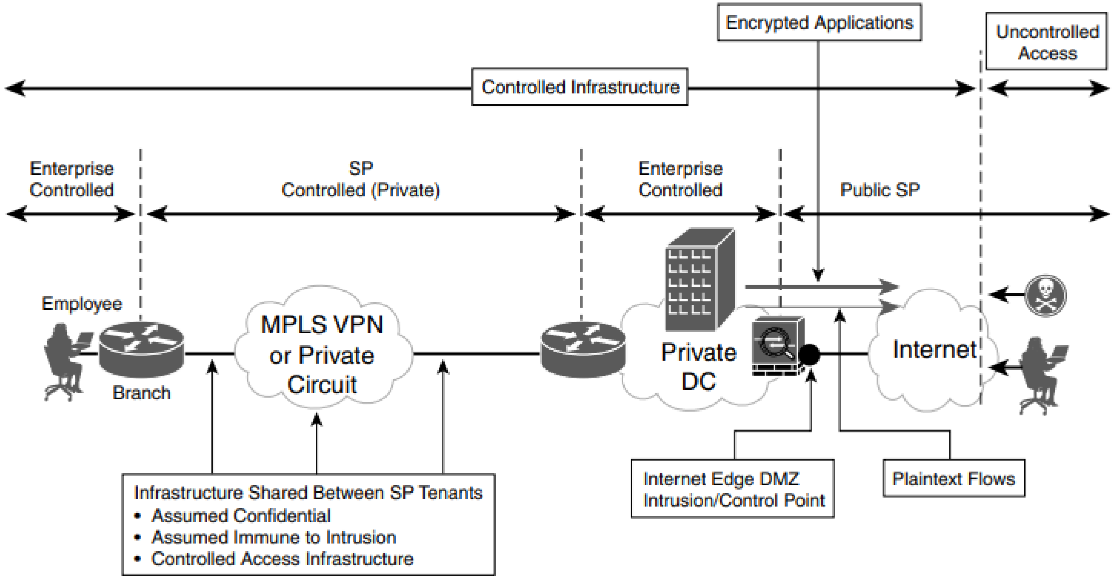

Figure 20-1 *Typical WAN Network*

Internet as a WAN Transport

- In Figure 20-2, the internet is used as the transport for the WAN. The internet does not provide controlled access and cannot guarantee data integrity or data confidentiality.
- Data confidentiality and integrity are maintained by adding IPsec encryption to the DMVPN tunnel that uses the internet as a transport. IPsec is a set of industry standards defined in RFC 2401 to secure IP-based network traffic.

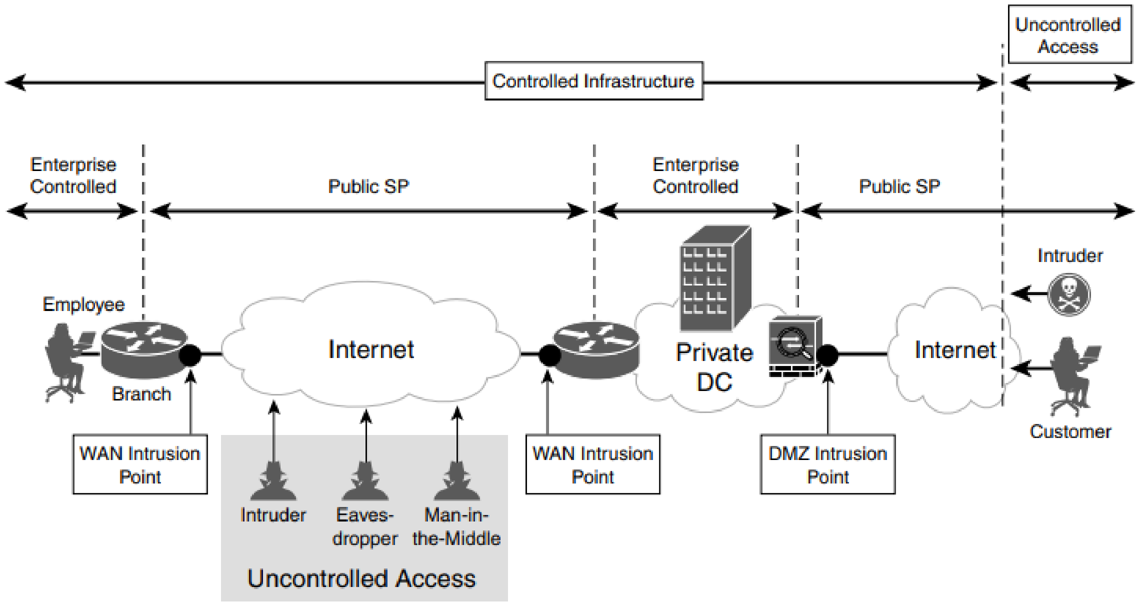

Figure 20-2 *Internet as a WAN Transport*

IPsec Fundamentals

- DMVPN tunnels are not encrypted by default, but they can be encrypted by using IPsec.
- IPsec provides encryption through cryptographically based security.
- The IPsec security architecture is composed of the following independent components: security protocols, security associations, and key management.

IPsec with DMVPN Tunnels

When IPsec is integrated with DMVPN tunnels, the encrypted DMVPN tunnels provide a secure overlay network over any transport with the following functions:

- **Origin authentication** - Authentication of origin is accomplished by Pre-Shared Key (static) or through certificate-based authentication (dynamic).
- **Data confidentiality** - Ensuring that data is viewable only by authorized users. Data confidentiality is maintained through encryption. A variety of encryption algorithms are used to preserve confidentiality.
- **Data integrity** - Hashing algorithms ensure that packets are not modified in transit.
- **Replay detection** - This provides protection against hackers trying to capture and insert network traffic.
- **Periodic rekey** - New security keys are created between endpoints every specified time interval or within a specific volume of traffic.
- **Perfect forward secrecy** - Each session key is derived independently of the previous key. A compromise of one key does not mean compromise of future keys.

Security Protocols

IPsec uses two protocols to provide data integrity and confidentiality. The protocols can be applied individually or combined based on need.

- **Authentication Header** - The IP authentication header provides data integrity, authentication, and protection from hackers replaying packets. It uses protocol number 51 (located in the IP header) to create a digital signature to ensure that the packet has not been modified during transport.

- **Encapsulating Security Payload (ESP)** - The Encapsulating Security Payload (ESP) provides data confidentiality, authentication, and protection from hackers replaying packets. Typically, payload refers to the actual data minus any headers, but in the context of ESP, the payload is the portion of the original packet that is encapsulated in the IPsec headers. ESP uses the protocol number 50 located in the IP header.

Key Management and Security Associations

- **Key Management** – Part of secure encryption is communicating the keys used to encrypt and decrypt traffic that is being transported over the insecure network. The process of generating, distributing, and storing these keys is called key management. IPSec uses Internet Key Exchange (IKE) protocol by default. IKEv2 provides mutual authentication of each party. IKEv2 introduced support of Extensible Authentication Protocol (EAP) (certificate-based authentication), reduction of bandwidth consumption, Network Address Translation (NAT), and the ability to detect whether a tunnel is still alive.
- **Security Associations (SAs)** – SAs contain the security parameters that were agreed upon between the two endpoint devices. There are two types of SAs:
  - **IKE SA -** Used for control plane functions like IPsec key management and management of IPsec SAs. Can have one IKE SA between endpoints.
  - **IPsec SA -** Used for data plane functions to secure data transmitted between two different sites. IPsec SAs are unidirectional. They require one inbound and one outbound to exchange network traffic between two sites.

DMVPN Packet Headers

ESP Modes - Traditional IPsec provides two ESP modes of packet protection:

- **Tunnel Mode** – Encrypts the entire original packet and adds a new set of IPsec headers. These new headers are used to route the packet and also to provide overlay functions.
- **Transport Mode** – Encrypts and authenticates only the packet payload. This mode does not provide overlay functions and routes based on the original IP headers.

Figure 20-3 shows an original packet, an IPsec packet in transport mode, and an IPsec packet in tunnel mode.

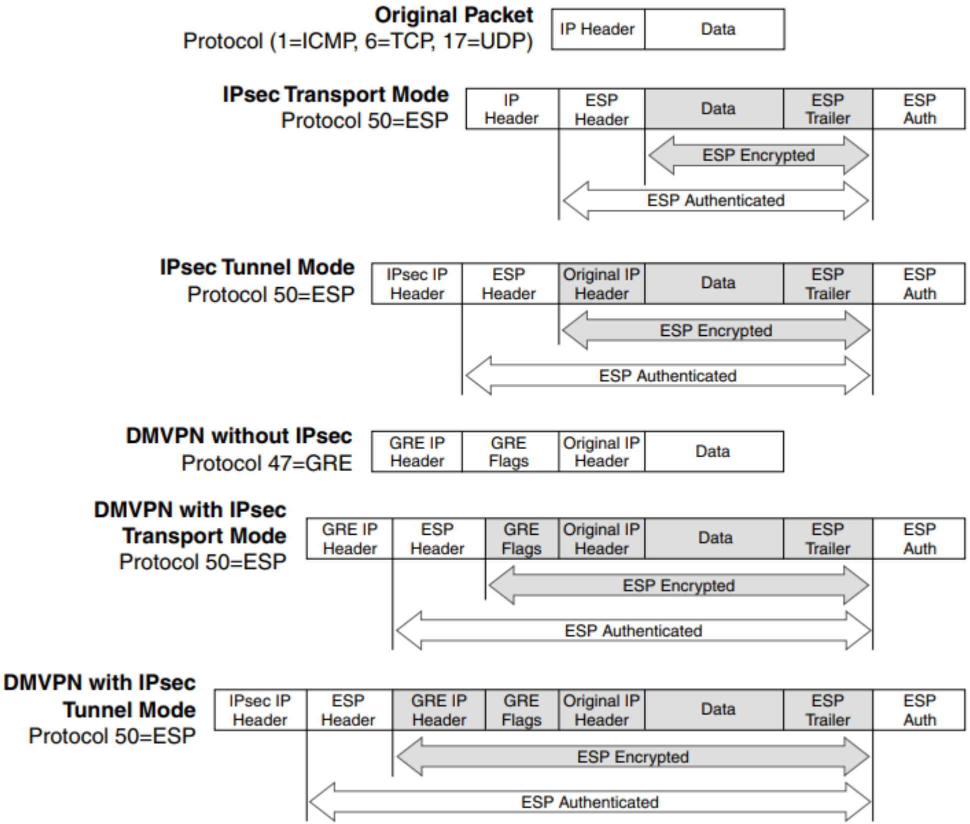

Figure 20-3 *DMVPN Packet Headers*

ESP Modes

- **DMVPN Without IPsec** - In unencrypted DMVPN packets, the original packets have GRE flags added to them. Then the new GRE IP header is added for routing the packets on the transport (underlay) network. The GRE IP header adds an extra 20 bytes of overhead, and the GRE flags add an extra 4 bytes. These packets use the protocol field of GRE (47).
- **DMVPN with IPsec in Transport Mode** - For encrypted DMVPN packets that use ESP transport mode, the original packets have the GRE flags added, then that portion of the packets is encrypted. A signature for the encrypted payload is added, and then a GRE IP header is added for routing the packets on the transport network. The GRE IP header adds an extra 20 bytes of overhead, the GRE flags add an extra 4 bytes, and depending on the encryption mechanism, a varying number of bytes are added for the encrypted signature. These packets use the protocol field of ESP (50).
- **DMVPN with IPsec in Tunnel Mode** - For encrypted DMVPN packets that use ESP tunnel mode, the original packets have GRE flags added to them, and then a new GRE IP header is added for routing the packets on the transport network. That portion of the packets is encrypted, a signature for the encrypted payload is added, and a new IPsec IP header is added for routing the packets on the transport network. The GRE IP header adds an extra 20 bytes of overhead, the GRE flags add an extra 4 bytes, the IPsec IP header adds an extra 20 bytes, and depending on the encryption mechanism, a varying number of bytes are added for the encrypted signature. These packets use the IP protocol field of ESP (50). IPsec tunnel mode for DMVPN does not add value, transport mode should be used for encrypted DMVPN tunnels.

IPsec Tunnel Protection

- Enabling IPsec protection on a DMVPN network requires that all devices have IPsec protection enabled.
- If some routers have IPsec enabled and others do not, devices with mismatched settings will not be able to establish connections on the tunnel interfaces.

Pre-Shared Key Authentication

- The first scenario for deploying IPsec tunnel protection is with the use of static Pre-Shared Key, which involves the creation of the following:
  - IKEv2 keyring
  - IKEv2 profile
  - IPsec transform set
  - IPsec profile

- In this section, emphasis is on the DMVPN routers that are attached to the internet, as shown in Figure 20-4.

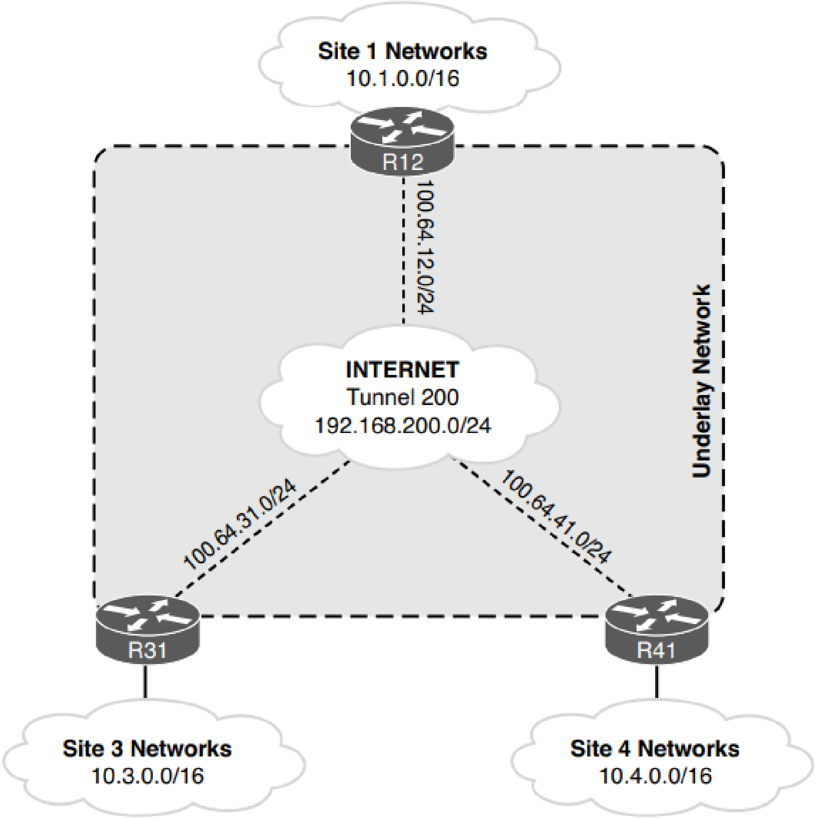

Figure 20-4 *Sample DMVPN Network*

IKEv2 Keyring

The IKEv2 keyring is a repository of the pre-shared keys and is created with the following steps:

Step 1. Create the keyring with the command **crypto ikev2 keyring** *keyring-name*.

Step 2. Create the peer with the command **peer** *peer-name*. Multiple peers can exist in a keyring. Each peer has a matching qualifier and can use a different password.

Step 3. Identify the IP address so that the appropriate peer configuration is used, based on the remote device's IP address. The command **address** *network subnet-mask* defines the IP address range.

Step 4. Define the pre-shared key with the command **pre-shared-key** *secure-key*. Generally a long and alphanumeric password is used for increased security.

Example 20-1 demonstrates a simple keyring that is used to secure the DMVPN routers on the internet.

Example 20-1 *IKEv2 Keyring*

```
crypto ikev2 keyring DMVPN-KEYRING-INET
 peer ANY
 address 0.0.0.0 0.0.0.0
 pre-shared-key CISCO456
```

IKEv2 Profile

The IKEv2 profile is a collection of nonnegotiable security parameters used during the IKE security association:

Step 1. Define the IKEv2 profile by using the command **crypto ikev2 profile** *ike-profile-name*.

Step 2. Define the peer IP address with the command **match identity remote address** *ip-address*.

Step 3. Optionally set the local router's identity based on an IP address by using the command **identity local** address *ip-address*.

Step 4. If Front Door VRF (FVRF) is used on the DMVPN tunnel, associate the FVRF instance with the IKEv2 profile with the command **match fvrf** {*vrf-name* **| any**}.

Step 5. Define the authentication method for connection requests received by remote peers by using the command **authentication local** {**pre-share** | **rsa-sig**}. The **pre-share** keyword is used for static keys, and **rsa-sig** is used for certificate-based authentication.

Step 6. Define the authentication method for connection requests sent to remote peers by using the command **authentication remote** {**pre-share** | **rsa-sig**}.

Step 7. For pre-shared authentication, associate the IKEv2 keyring with the IKEv2 profile by using the command **keyring local** *keyring-name*.

IPsec Tunnel Protection

IKEv2 Profile (Cont.)

The IKEv2 profile settings are displayed with the command **show crypto ikev2 profile**, as shown in Example 20-3. Notice that the authentication, FVRF, IKE keyring, and identity IP address are displayed along with the IKE lifetime.

Example 20-3 *Display of IKEv2 Profile Settings*

```
R12-DC1-Hub2# show crypto ikev2 profile
IKEv2 profile: DMVPN-IKE-PROFILE-INET
 Ref Count: 1
 Match criteria:
  Fvrf: INET01
  Local address/interface: none
  Identities:
   address 0.0.0.0
  Certificate maps: none
 Local identity: none
 Remote identity: none
 Local authentication method: pre-share
 Remote authentication method(s): pre-share
 EAP options: none
 Keyring: DMVPN-KEYRING-INET
 Trustpoint(s): none
 Lifetime: 86400 seconds
 DPD: disabled
 NAT-keepalive: disabled
 Ivrf: none
 Virtual-template: none
 mode auto: none
 AAA AnyConnect EAP authentication mlist: none
 AAA EAP authentication mlist: none
 AAA Accounting: none
 AAA group authorization: none
 AAA user authorization: none
```

IPsec Transform Set

The transform set identifies the security protocols (such as ESP) for encrypting traffic. It specifies the protocol ESP or authentication header that is used to authenticate the data:

Step 1. Create the transform set and identify the transforms by using the command **crypto ipsec transform-set** *transform-set-name* [*esp-encryption-name*] [*esp-authentication-name*] [*ah-authentication-name*].

Step 2. Configure the ESP mode by using the command **mode** {**transport** | **tunnel**}.

Example 20-4 provides a sample IPsec transform set.

Example 20-4 *Sample IPsec Transform Set*

```
crypto ipsec transform-set AES256/SHA/TRANSPORT esp-aes 256 esp-sha-hmac
 mode transport
```

The transform set can be verified with the command **show crypto ipsec transform-set**, as shown in Example 20-5.

Example 20-5 *Verification of the IPsec Transform Set*

```
R12-DC1-Hub2# show crypto ipsec transform-set
! Output omitted for brevity
Transform set AES256/SHA/TRANSPORT: { esp-256-aes esp-sha-hmac }
  will negotiate = { Transport, },
```

IPsec Profile

The IPsec profile combines the IPsec transform set and the IKEv2 profile:

Step 1. Create the IPsec profile by using the command **crypto ipsec profile** *profile-name*.

Step 2. Specify the transform set by using the command **set transform-set** *transform-set-name.*

Step 3. Specify the IKEv2 profile by using the command **set ikev2-profile** *ike-profile-name*.

Example 20-6 provides a sample IPsec profile configuration.

Example 20-6 *Sample IPsec Profile*

```
crypto ipsec profile DMVPN-IPSEC-PROFILE-INET
 set transform-set AES256/SHA/TRANSPORT
 set ikev2-profile DMVPN-IKE-PROFILE-INET
```

The command **show crypto ipsec profile** displays the components of the IPsec profile, as shown in Example 20-7.

Example 20-7 *Verification of the IPsec Profile*

```
R12-DC1-Hub2# show crypto ipsec profile
! Output omitted for brevity
IPSEC profile DMVPN-IPSEC-PROFILE-INET
          IKEv2 Profile: DMVPN-IKE-PROFILE-INET
          Security association lifetime: 4608000 kilobytes/3600 seconds
          Responder-Only (Y/N): N
          PFS (Y/N): N
          Mixed-mode : Disabled
          Transform sets={
                  AES256/SHA/TRANSPORT: { esp-256-aes esp-sha-hmac } ,
```

Encrypt the Tunnel Interface/IPsec Packet Replay Protection

- When all the required IPsec components have been configured, the IPsec profile is associated to the DMVPN tunnel interface with the command **tunnel protection ipsec profile** *profile-name* [**shared**]. The **shared** keyword is required for routers that terminate multiple encrypted DMVPN tunnels on the same transport interface. The command shares the IPsec security association database (SADB) among multiple DMVPN tunnels.
- Cisco IPsec includes an anti-replay mechanism that prevents intruders from duplicating encrypted packets. A unique sequence number is assigned to each encrypted packet. When a router decrypts the IPsec packets, it keeps track of the packets it has received. The IPsec anti-replay service rejects (discards) duplicate packets or old packets. The router maintains a sequence number window size (default of 64 packets). The minimum sequence number is the highest sequence number for a packet minus the window size. A packet is considered of age when the sequence number is between the minimum sequence number and the highest sequence number.
- The window size is increased globally with the command **crypto ipsec security-association replay window-size** *window-size*. Cisco recommends using the largest window size possible for the platform, which is 1024.

Dead Peer Detection/NAT Keepalives

- Dead Peer Detection (DPD) helps detect the loss of connectivity to a remote IPsec peer. When DPD is enabled in on-demand mode, the two routers check for connectivity only when traffic needs to be sent to the IPsec peer and the peer's active status is not certain. The router sends a DPD R-U-THERE request to query the status of the remote peer. If the remote router does not respond to the R-U-THERE request, the requesting router starts to transmit additional R-U-THERE messages every retry interval for a maximum of five retries. After that, the peer is declared dead. DPD is configured with the command **crypto ikev2 dpd** [*interval-time*] [*retry-time*] **on-demand** in the IKEv2 profile.
- NAT keepalives keep the dynamic NAT mapping alive during a connection between two peers. A NAT keepalive is a UDP packet that contains an unencrypted payload of 1 byte. When DPD is used to detect peer status, NAT keepalives are sent if the IPsec entity has not transmitted or received a packet within a specified time period. NAT keepalives are enabled with the command **crypto isakmp nat keepalive** *seconds*.

IPsec DMVPN Configuration with Pre-Shared Authentication

Example 20-9 displays the complete configuration to enable IPsec protection on the internet DMVPN tunnel on R12, R31, and R41 with all the settings from this section.

```
R12
crypto ikev2 keyring DMVPN-KEYRING-INET
 peer ANY
   address 0.0.0.0 0.0.0.0
   pre-shared-key CISCO456
!
crypto ikev2 profile DMVPN-IKE-PROFILE-INET
 match fvrf INET01
 match identity remote address 0.0.0.0
 authentication remote pre-share
 authentication local pre-share
 keyring local DMVPN-KEYRING-INET
!
crypto ipsec transform-set AES256/SHA/TRANSPORT esp-aes 256 esp-sha-hmac
 mode transport
!
crypto ipsec profile DMVPN-IPSEC-PROFILE-INET
 set transform-set AES256/SHA/TRANSPORT
 set ikev2-profile DMVPN-IKE-PROFILE-INET
!
interface Tunnel200
 tunnel protection ipsec profile DMVPN-IPSEC-PROFILE-INET
!
crypto ipsec security-association replay window-size 1024
```

```
R31 and R41
crypto ikev2 keyring DMVPN-KEYRING-INET
 peer ANY
  address 0.0.0.0 0.0.0.0
  pre-shared-key CISCO456
!
crypto ikev2 profile DMVPN-IKE-PROFILE-INET
 match fvrf INET01
 match identity remote address 0.0.0.0
 authentication remote pre-share
 authentication local pre-share
 keyring local DMVPN-KEYRING-INET
 dpd 40 5 on-demand
!
crypto ipsec transform-set AES256/SHA/TRANSPORT esp-aes 256 esp-sha-hmac
 mode transport

!
crypto ipsec profile DMVPN-IPSEC-PROFILE-INET
 set transform-set AES256/SHA/TRANSPORT
 set ikev2-profile DMVPN-IKE-PROFILE-INET
!
interface Tunnel200
 tunnel protection ipsec profile DMVPN-IPSEC-PROFILE-INET
!
crypto ipsec security-association replay window-size 1024
!
crypto isakmp nat keepalive 20
```

IPsec Tunnel Protection

Verification of Encryption on DMVPN Tunnels

When the DMVPN tunnels have been configured for IPsec protection, verify the status. The command **show dmvpn detail** provides the relevant IPsec information. Example 20-10 demonstrates the command on R31. The output lists the status of the DMVPN tunnel, the underlay IP addresses, and packet counts. Examining the packet counts can help to verify that network traffic is being transmitted out of a DMVPN tunnel or received on a DMVPN tunnel.

The command **show crypto ipsec sa** provides additional information that is not included in the output of the command **show dmvpn detail**, such as the path MTU, tunnel mode and replay detection.

```
R31-Spoke# show dmvpn detail
! Output omitted for brevity
# Ent Peer NBMA Addr  Peer Tunnel Add State UpDn Tm Attrb Target Network
----- --------------- --------------- ----- -------- ----- ----------------
    1 100.64.12.1      192.168.200.12     UP  00:03:39    S  192.168.200.12/32


Crypto Session Details:
--------------------------------------------------------------------------------
Interface: Tunnel200
Session: [0xE7192900]
 Session ID: 1
 IKEv2 SA: local 100.64.31.1/500 remote 100.64.12.1/500 Active
 Capabilities:(none) connid:1 lifetime:23:56:20
 Crypto Session Status: UP-ACTIVE
 fvrf: INET01, Phase1_id: 100.64.12.1
 IPSEC FLOW: permit 47 host 100.64.31.1 host 100.64.12.1
       Active SAs: 2, origin: crypto map
       Inbound: #pkts dec'ed 22 drop 0 life (KB/Sec) 4280994/3380
       Outbound: #pkts enc'ed 20 drop 0 life (KB/Sec) 4280994/3380
 Outbound SPI : 0x35CF62F4, transform : esp-256-aes esp-sha-hmac
   Socket State: Open

Pending DMVPN Sessions:
```

IKEv2 Protection

IKEv2 was developed, in part, to protect routers from various IKE intrusion methods. Primarily, it limits the number of packets required to process IKE establishment. During high CPU utilization, a session that has started may not complete because other sessions are consuming limited CPU resources. Problems can occur when the number of expected sessions is different from the number of sessions that can be established. Limiting the number of sessions that can be in negotiation minimizes the CPU resources needed so that the expected number of established sessions can be obtained.

The command **crypto ikev2 limit** {**max-in-negotiation-sa** *limit* | **max-sa** *limit*} [**outgoing**] limits the number of sessions being established or that are allowed to be established:

- The **max-sa** keyword limits the total count of SAs that a router can establish under normal conditions. You set the value to double the number of ongoing sessions in order to achieve renegotiation.
- To limit the number of SAs being negotiated at one time, you can use the **max-in-negotiation-sa** keyword.
- To protect IKE from half-open sessions, a cookie can be used to validate that sessions are valid IKEv2 sessions and not denial-of-service intrusions. The command **crypto ikev2 cookie-challenge** *challenge-number* defines the threshold of half-open SAs before issuing an IKEv2 cookie challenge.

IKEv2 Protection (Cont.)

In Example 20-12, R41 limits the number of SAs to 10, limits the number in negotiation to 6, and sets an IKEv2 cookie challenge for sessions above 4. R41 has 1 static session to the hub router (R11) and is limited to 9 additional sessions that all use the IKEv2 cookie challenge.

The command **show crypto ikev2 stats** displays the SA restrictions and shows that the four sessions are currently established to the four DMVPN hub routers.

```
R41-Spoke(config)# crypto ikev2 limit max-sa 10
R41-Spoke(config)# crypto ikev2 limit max-in-negotation-sa 6 outgoing
R41-Spoke(config)# crypto ikev2 limit max-in-negotation-sa 6
R41-Spoke(config)# crypto ikev2 cookie-challenge 4
R41-Spoke(config)# end
```

```
R41-Spoke# show crypto ikev2 stats
--------------------------------------------------------------------------------
                Crypto IKEv2 SA Statistics
--------------------------------------------------------------------------------
System Resource Limit:    0     Max  IKEv2 SAs: 10  Max in nego(in/out): 6/6
Total incoming IKEv2 SA Count: 0           active:    0    negotiating: 0
Total outgoing IKEv2 SA Count: 4           active:    4    negotiating: 0
Incoming IKEv2 Requests: 1      accepted:    1        rejected:     0
Outgoing IKEv2 Requests: 4      accepted:    4        rejected:     0
Rejected IKEv2 Requests: 0     rsrc low:     0        SA limit:     0
IKEv2 packets dropped at dispatch: 0
Incoming IKEV2 Cookie Challenged Requests: 0
               accepted: 0      rejected: 0           rejected no cookie: 0
Total Deleted sessions of Cert Revoked Peers: 0
conformed 0000 bps, exceeded 0000 bps, violated 0000 bps
```

Prepare for the Exam

Prepare for the Exam

Key Topics for Chapter 20

Description	
Data security terms	IPsec transform set
Security associations	Encrypting the tunnel interface
ESP modes	IPsec packet replay protection
IKEv2 keyring	Verification of encryption on DMVPN tunnels
IKEv2 profile	IKEv2 protection

Prepare for the Exam
Key Terms for Chapter 20

Key Terms
Authentication Header (AH) protocol
Encapsulating Security Payload (ESP)
Data confidentiality
Data integrity
Data availability
Origin authentication
Replay detection
Periodic rekey
Security association (SA)

Prepare for the Exam

Command Reference for Chapter 20

Task	Command Syntax
Configure an IKEv2 keyring	**crypto ikev2 keyring** *keyring-name* **peer** *peer-name* **address** *network subnet-mask* **pre-shared-key** *secure-key*
Configure an IKEv2 profile	**crypto ikev2 profile** *ike-profile-name* **match identity remote address** *ip-address* **match fvrf** {*vrf-name* | *any*} **authentication local pre-share** **authentication remote pre-share** **keyring local** *keyring-name*
Configure an IPsec transform set	**crypto ipsec transform-set** *transform-set-name [esp-encryption-name] [esp-authentication-name] [ah-authentication-name]* **mode** {**transport** | **tunnel**}

Prepare for the Exam

Command Reference for Chapter 20 (Cont.)

Task	Command Syntax
Configure an IPsec profile	**crypto ipsec profile** *profile-name* **set transform-set transform-set-name** **set ikev2-profile** *ike-profile-name*
Encrypt the DMVPN tunnel interface	**tunnel protection ipsec profile profile-name** [**shared**]
Modify the default IPsec replay window size	**crypto ipsec security-association replay window-size** *window-size*
Enable IPsec NAT keepalives	**crypto isakmp nat keepalive** *seconds*
Display the IKEv2 profile	**show crypto ikev2 profile**
Display the IPsec profile	**show crypto ipsec profile**

cisco