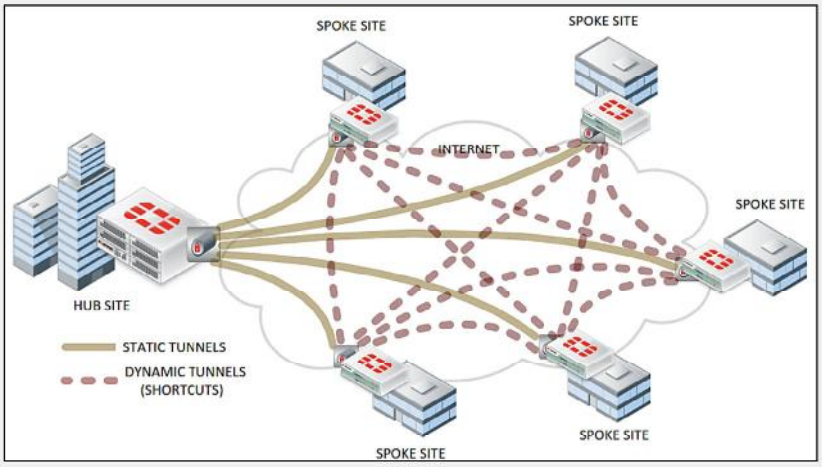
Auto-Discovery VPN

- Dynamically negotiates on-demand direct VPNs between spokes
  - Provides the benefits of a full mesh topology over a hub-and-spoke or partial mesh deployment
  - Dynamic routing is recommended to learn routes between hub and spokes and scale up easier
  - Static routing also works, but should be used for small deployments only



FORTINET Training Institute

14

Each VPN topology has its advantages and disadvantages.

Auto-discovery VPN (ADVPN) is a FortiGate feature that achieves the benefits of a full-mesh topology with the easier configuration and scalability benefits of hub-and-spoke and partial-mesh topologies.

First, you add the VPN configurations for building either a hub-and-spoke or a partial-mesh topology, to the FortiGate devices. Then, you enable ADVPN on the VPNs. ADVPN dynamically negotiates tunnels between spokes (without having them preconfigured) to get the benefits of a full-mesh topology.

You can use dynamic routing and static routing to deploy ADVPN. A dynamic routing protocol, such as BGP, is usually deployed in large networks because it enables you to exchange routing information between spokes and hub easier, and as a result, to scale up. You can also use static routing to deploy ADVPN, but it is recommended to do so in small networks that are not likely to grow considerably.

Whether you use dynamic routing or not, after a shortcut is negotiated, FortiGate automatically adds routes through the shortcut to redirect spoke-to-spoke traffic through it.

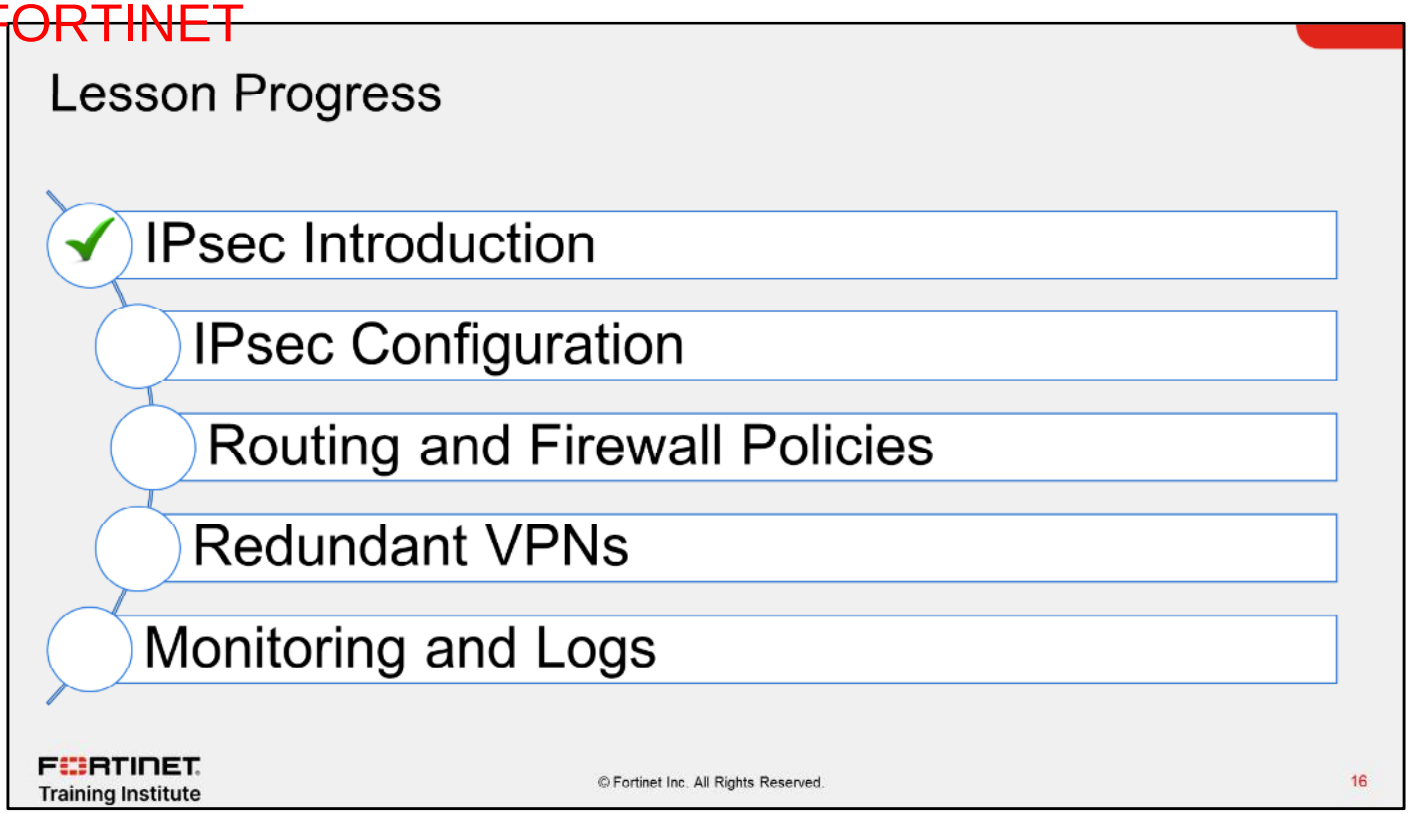
Knowledge Check

1. Which IPsec protocol is not supported by FortiGate?
   - A. IKEv2
   - ✓ B. AH

2. Which VPN topology is the most fault tolerant?
   - ✓ A. Full mesh
   - B. Hub-and-spoke

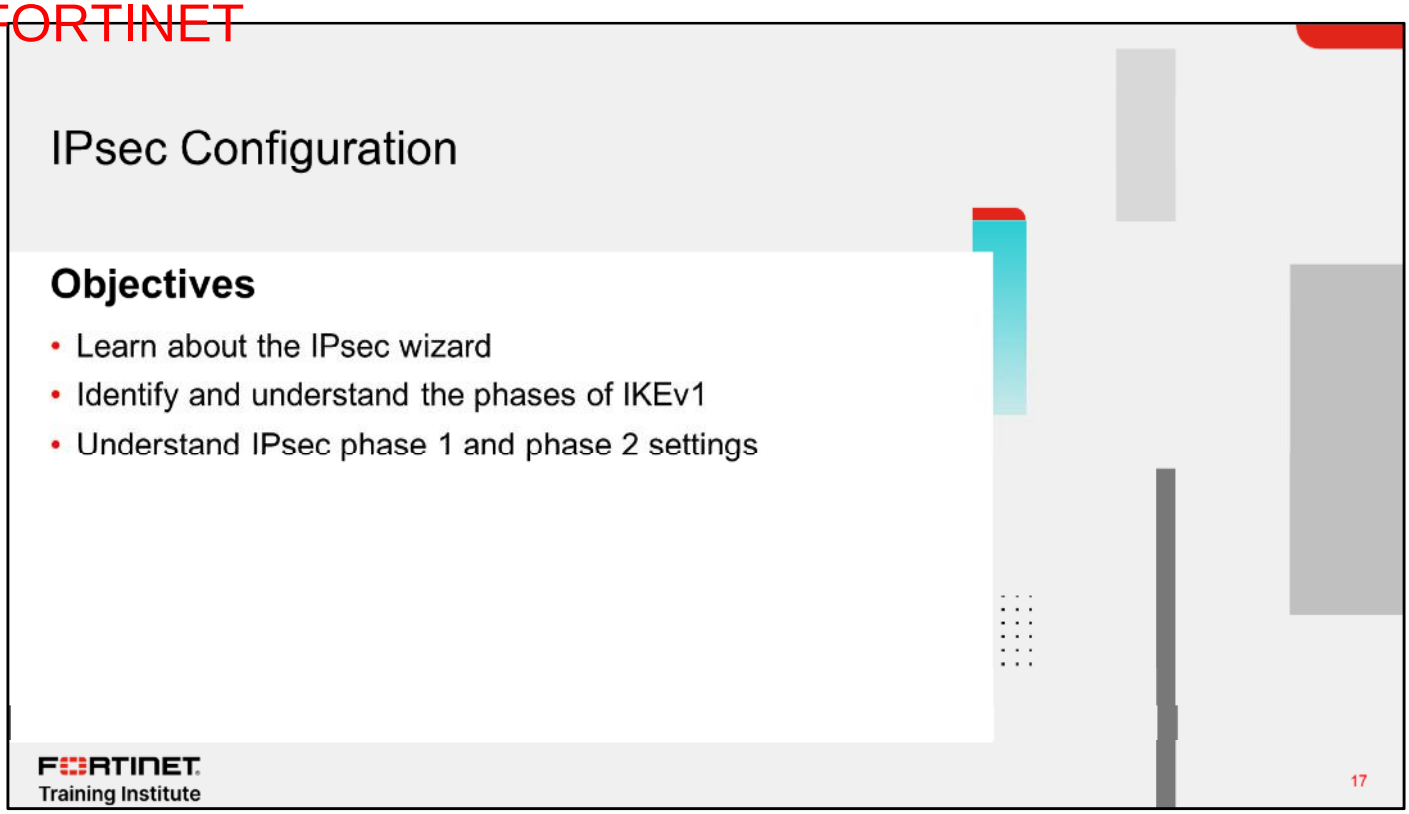
FORTINET Training Institute

15



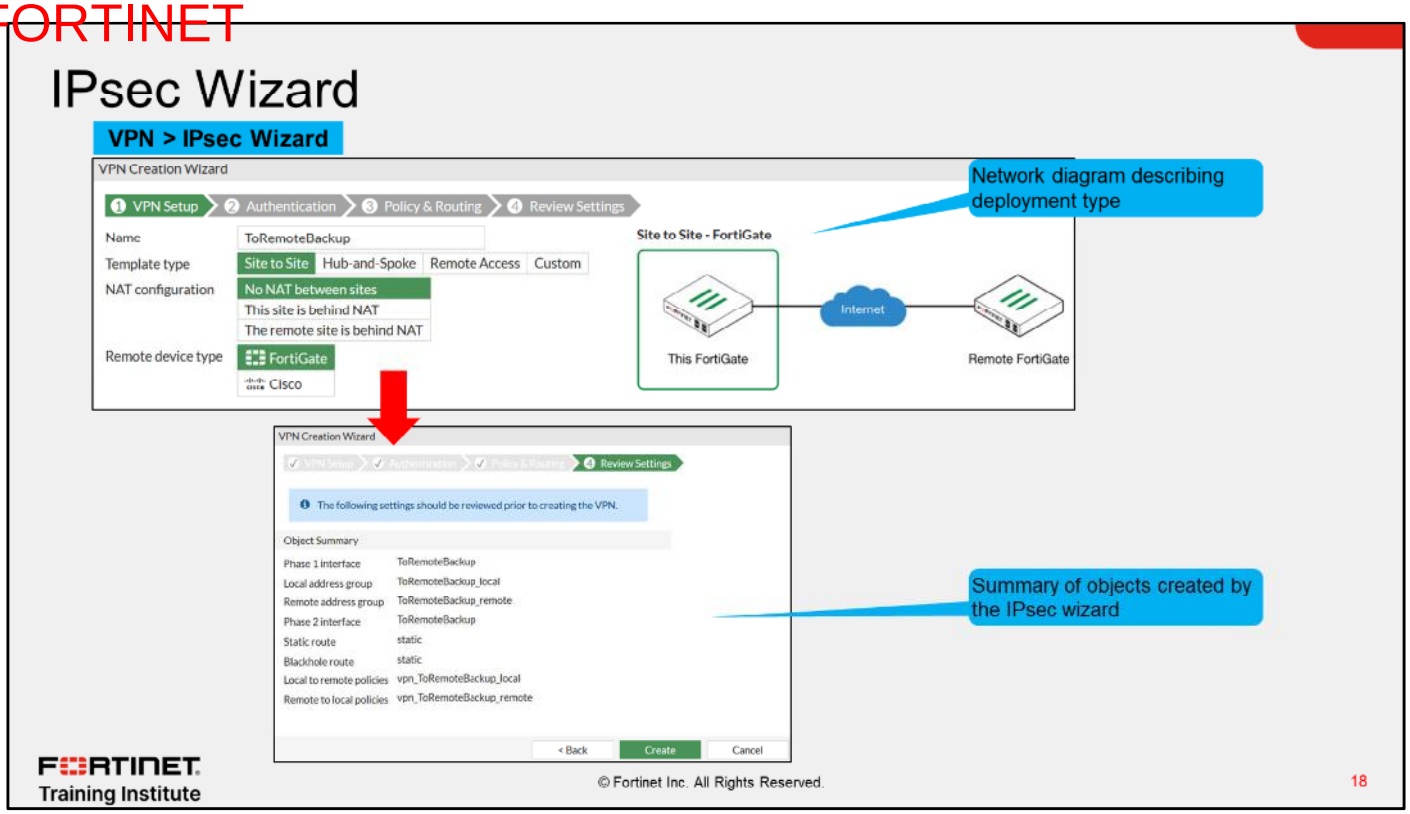
Good job! You have now been introduced to IPsec.

Now, you will learn about IPsec configuration.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in IPsec configuration, you will be able to successfully determine the settings required for your IPsec VPN deployment.



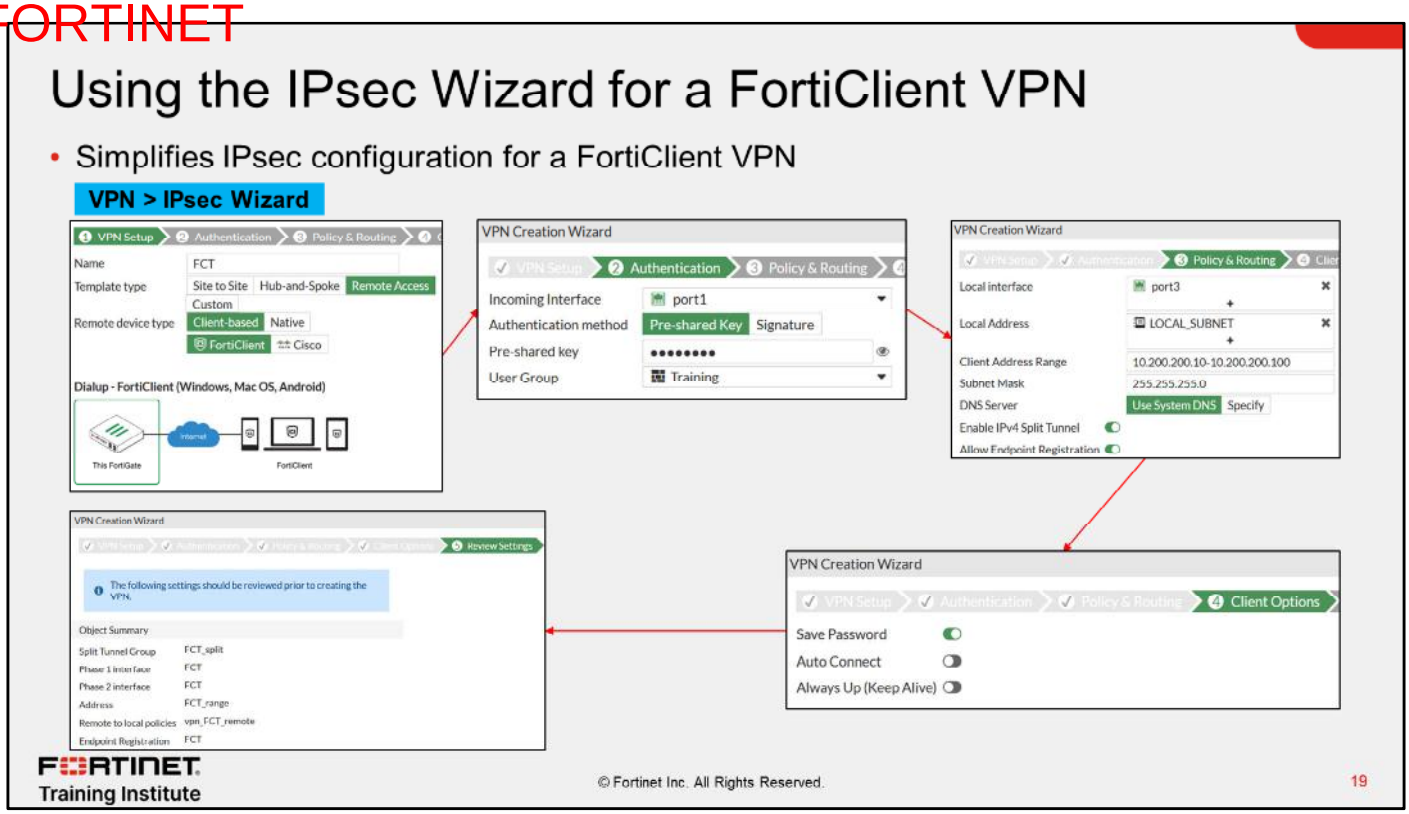
When you create an IPsec tunnel on the GUI, FortiGate redirects you to the **IPsec Wizard**. The wizard simplifies the creation of the new VPN by walking you through a four to five-step process. The first step is to select a template type. If you want to manually configure your VPN, you can select **Custom** as **Template type**, upon which FortiGate takes you directly to the phase 1 and phase 2 settings of the new VPN.

If you want the wizard to configure the VPN for you, then select the template type (**Site to Site**, **Hub-and-Spoke**, or **Remote Access**) that best matches your VPN. After that, the wizard asks you for key information such as the remote gateway information, authentication method, interfaces involved, and subnets. Based on the input you provide, the wizard applies one of the preconfigured IPsec tunnel templates comprising IPsec phase 1 and 2 settings and other related firewall address objects, routing settings, and firewall policies needed for the new tunnel to work.

In addition, the wizard shows a network diagram that changes based on the input provided. The purpose of the diagram is for the administrator to have a visual understanding of the IPsec VPN deployment that the wizard configures based on the input received.

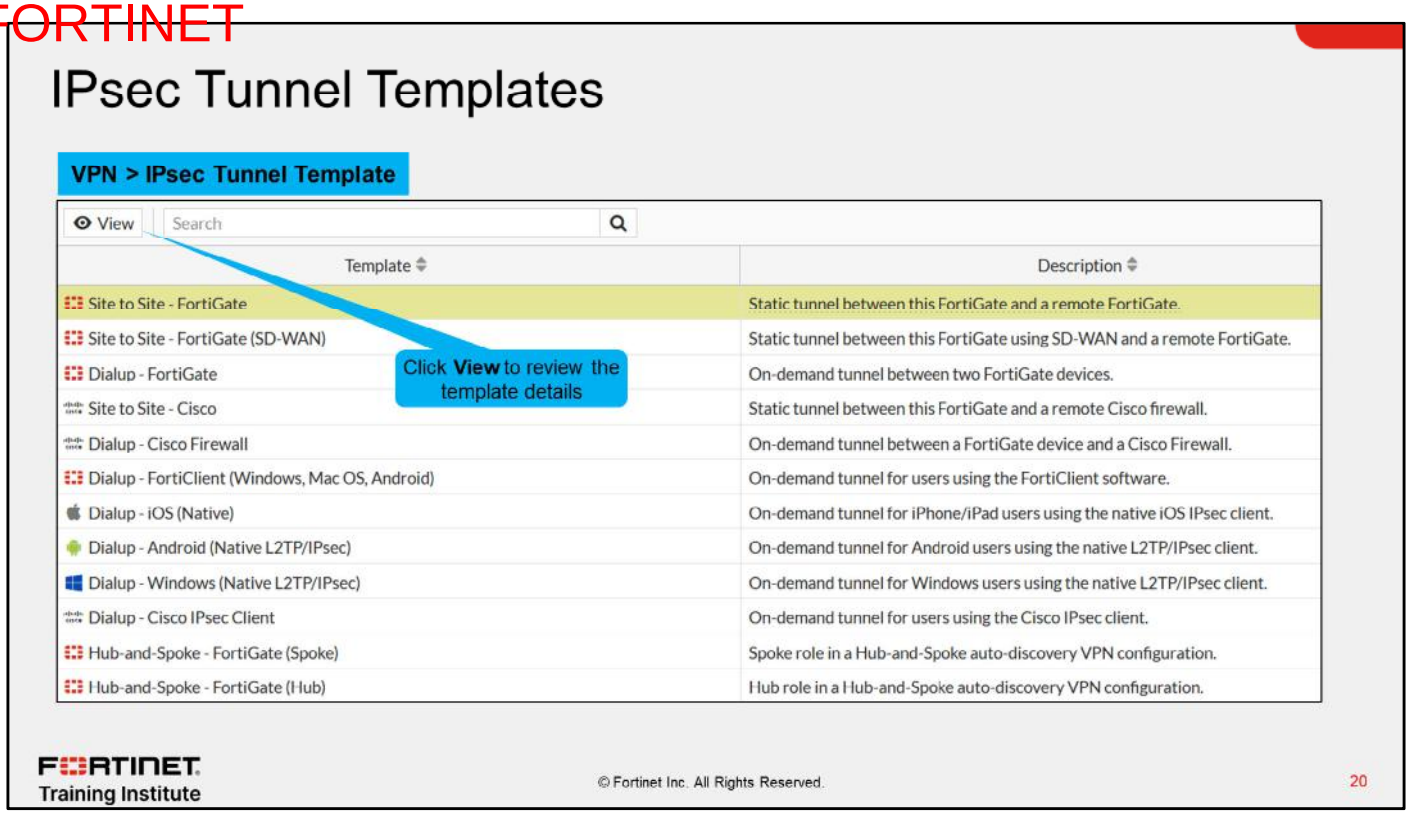
At the end of the wizard, the wizard provides a summary of the configuration changes made in the system, and that the administrator can review if needed.

If you are new to FortiGate, or don't have much experience with IPsec VPNs, using the IPsec wizard is recommended. You can later adjust the configuration applied by the wizard to match your specific needs.

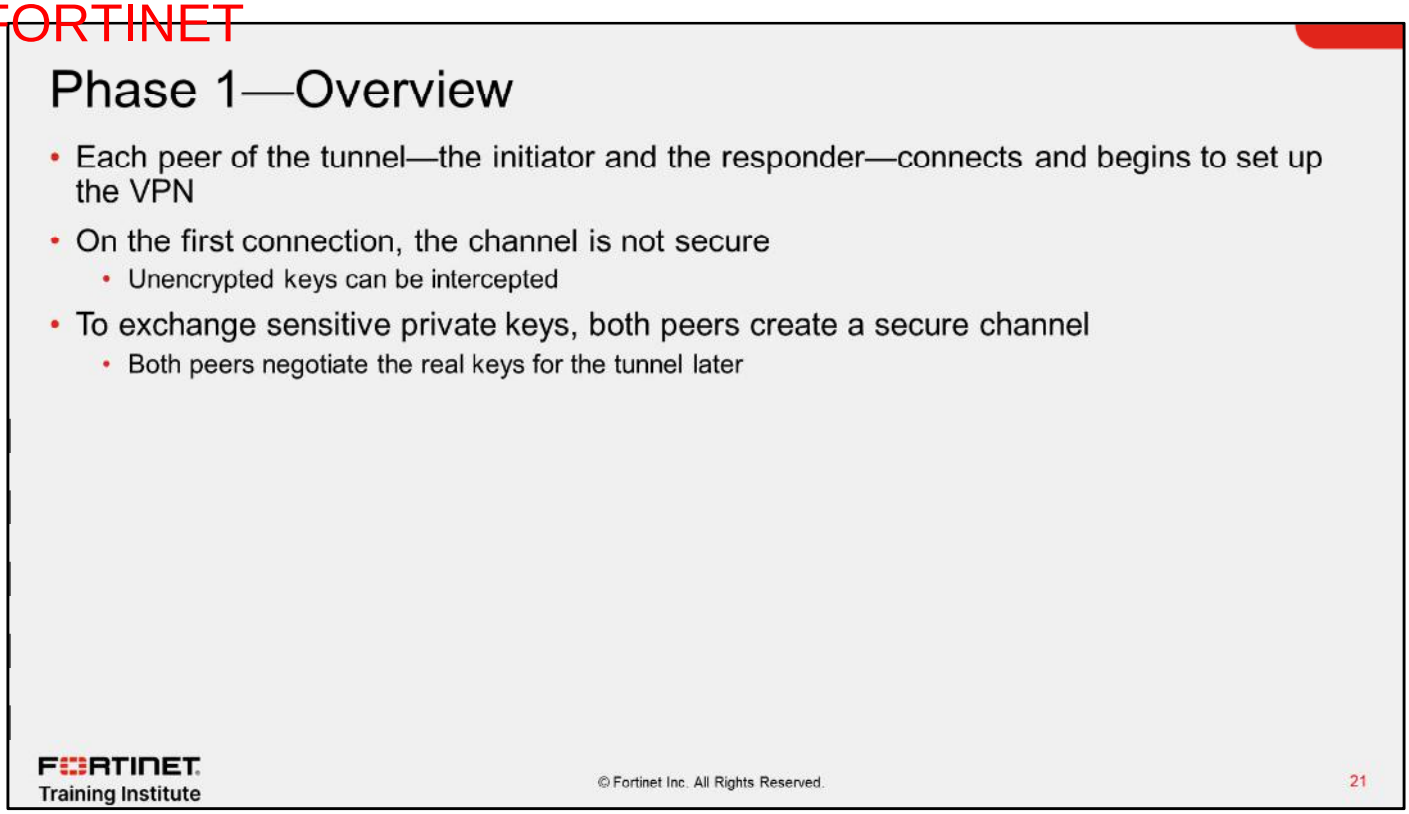


A common use of the IPsec wizard is for configuring a remote access VPN for FortiClient users. The wizard enables IKE mode config, XAuth, and other appropriate settings for FortiClient users. You will learn more about IKE mode config and XAuth in this lesson.

The images on this slide show the four-step process used by the IPsec wizard for assisting the administrator on the FortiClient VPN configuration.

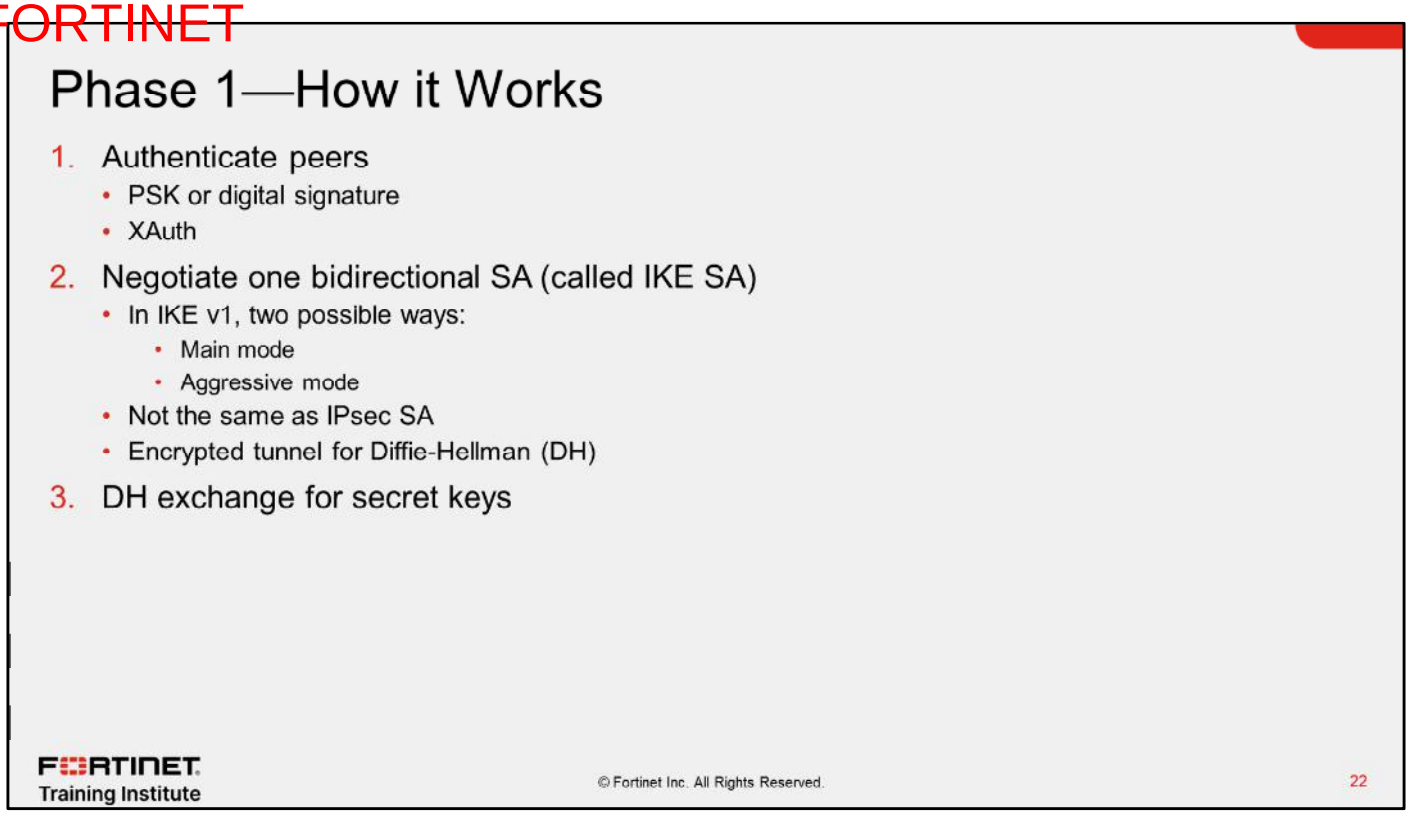


The IPsec wizard uses one of the templates shown on this slide when applying the configuration for the new IPsec tunnel. You can review the settings of a template by selecting the template, and then clicking **View**. You cannot change the template settings.



Phase 1 takes place when each peer of the tunnel—the initiator and the responder—connects and begins to set up the VPN. The initiator is the peer that starts the phase 1 negotiation, while the responder is the peer that responds to the initiator request.

When the peers first connect, the channel is not secure. An attacker in the middle could intercept unencrypted keys. Neither peer has a strong guarantee of the other peer's identity, so how can they exchange sensitive private keys? They can't. First, both peers create a secure tunnel. Then, they use this secure tunnel to negotiate the real keys for the tunnel later.



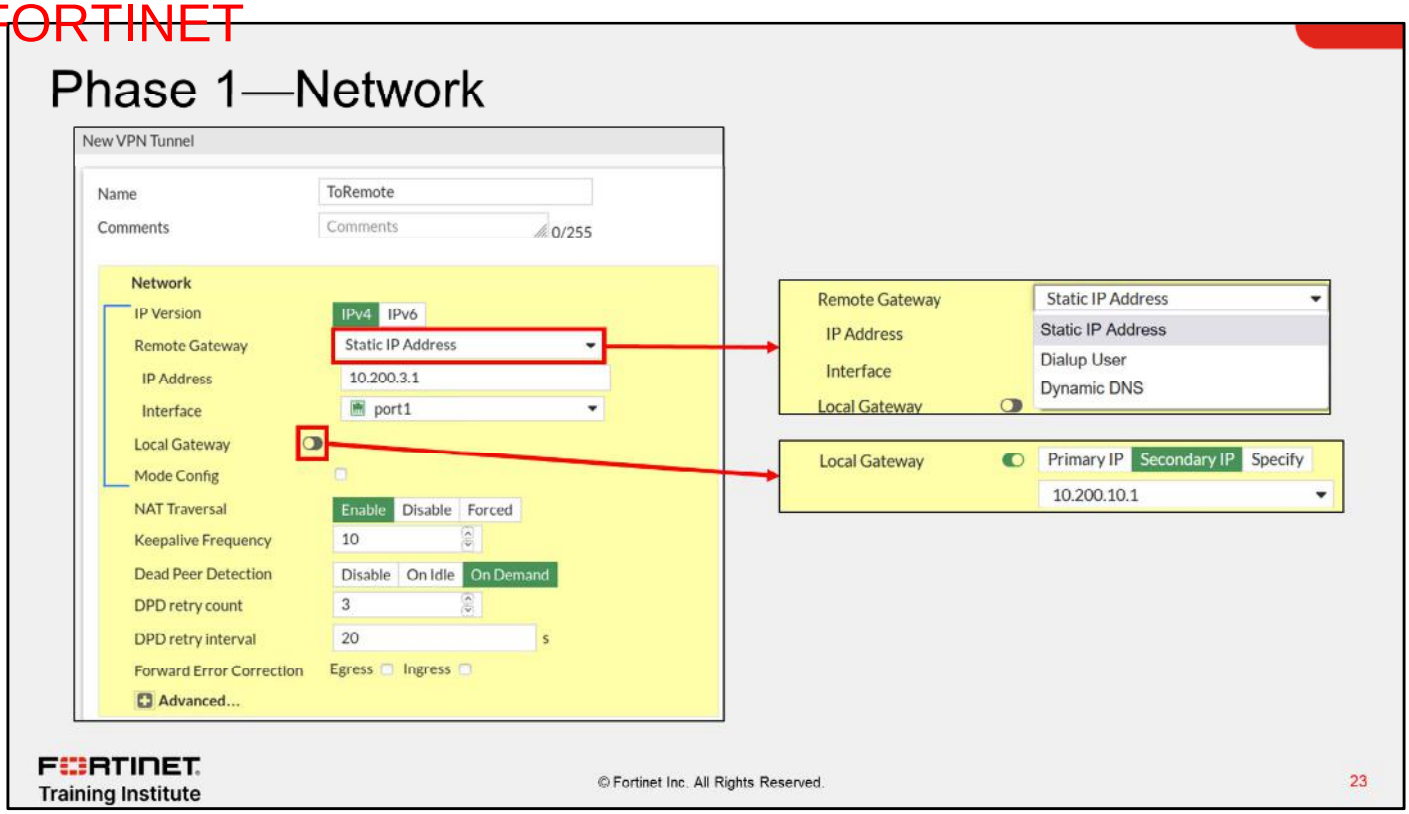
Now you'll examine how phase 1 works.

The purpose of phase 1 is to authenticate peers and set up a secure channel for negotiating the phase 2 SAs (or IPsec SAs) that are later used to encrypt and decrypt traffic between the peers. To establish this secure channel, the peers negotiate a phase 1 SA. This SA is called the IKE SA and is bidirectional.

To authenticate each other, the peers use two methods: pre-shared key or digital signature. You can also enable an additional authentication method, XAuth, to enhance authentication.

In IKEv1, there are two possible modes in which the IKE SA negotiation can take place: main, and aggressive mode. Settings on both ends must agree; otherwise, phase 1 negotiation fails and both IPsec peers are not able to establish a secure channel.

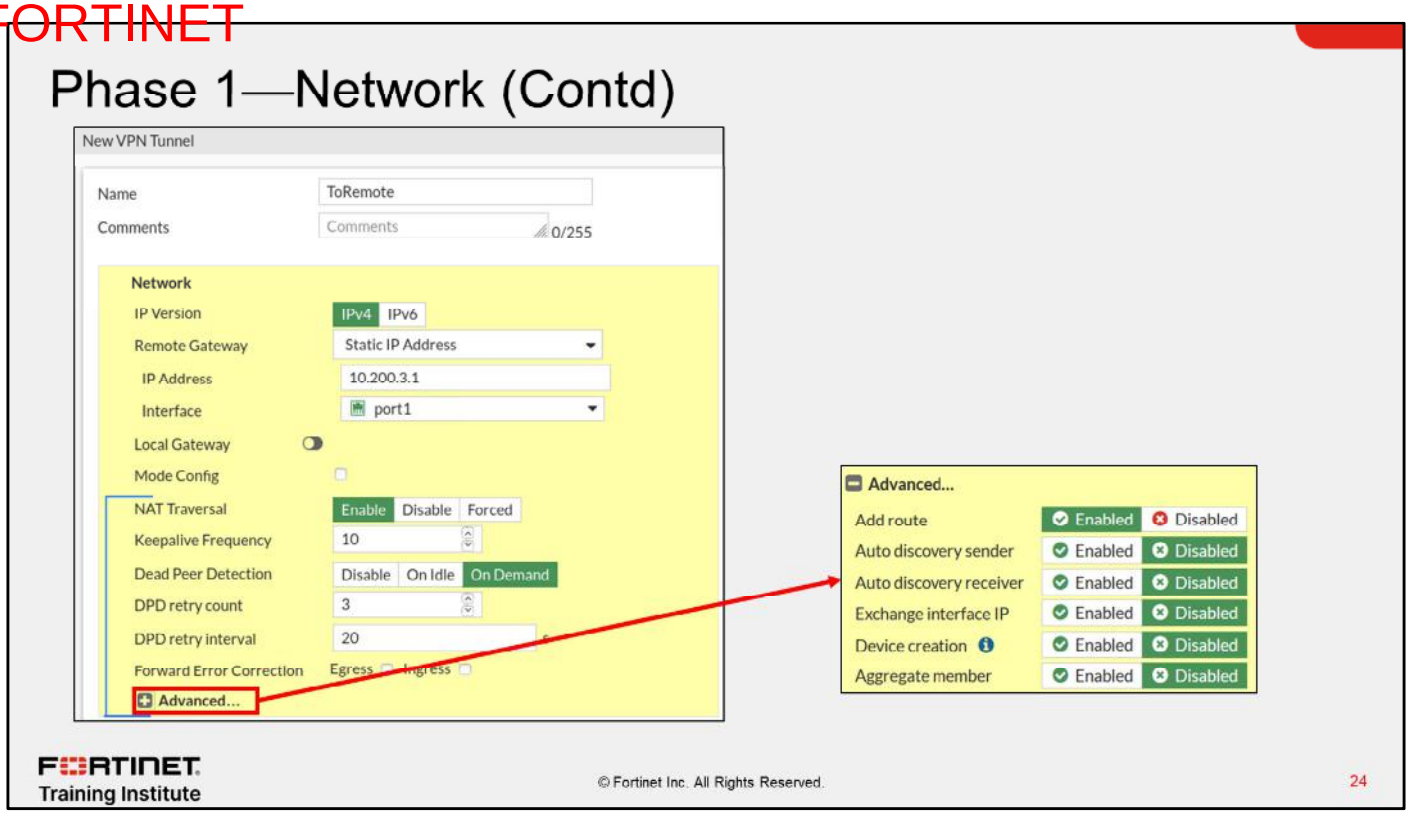
At the end of phase 1, the negotiated IKE SA is used to negotiate the DH keys that are used in phase 2. DH uses the public key (that both ends know) plus a mathematical factor called a nonce, in order to generate a common private key. With DH, even if an attacker can listen to the messages containing the public keys, they cannot determine the secret key.



Phase 1 configuration is broken down on the GUI into four sections: **Network**, **Authentication**, **Phase 1 Proposal**, and **XAUTH**. You will learn about the settings available on each section. You will learn about some of these settings in more detail on separate slides.

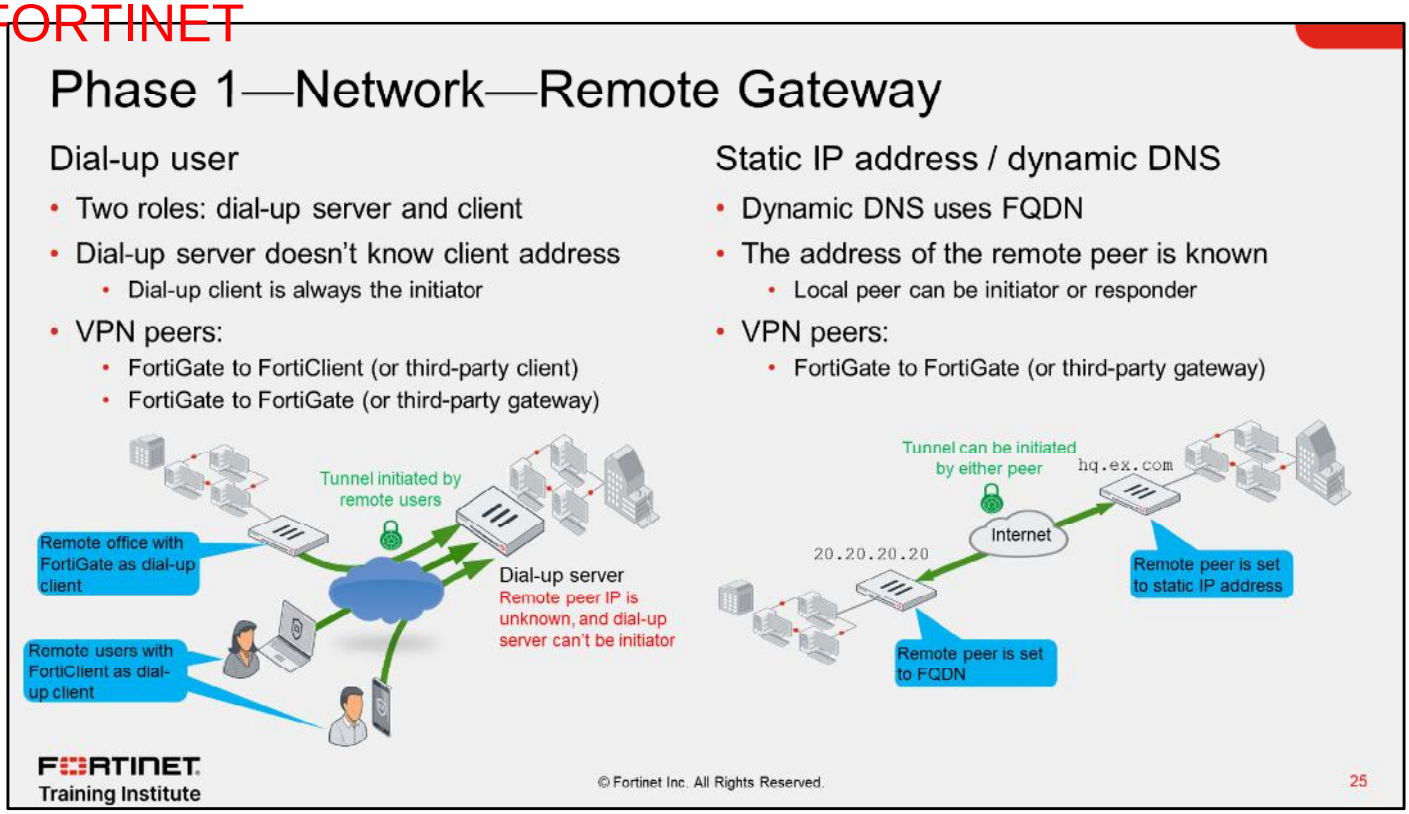
The section shown on this slide corresponds to the **Network** settings. The section includes the settings related to the connectivity of the IPsec tunnel:

- **IP Version**: select the IP version to use for the IPsec tunnel. Note that this defines only the IP version of the outer layer of the tunnel (after encapsulation). The packets being encapsulated (protected traffic) can be IPv4 or IPv6, and their IP version is defined in the phase 2 selectors.
- **Remote Gateway**: defines the type of the remote gateway. There are three types: **Static IP Address**, **Dialup User**, and **Dynamic DNS**. You will learn more about these types later in this lesson.
- **IP Address**: the IP address of the remote gateway. This field appears only when you select **Static IP Address** as **Remote Gateway**.
- **Interface**: refers to the interface where the IPsec tunnel terminates on the local FortiGate. Usually, this is the interface connected to the internet or the WAN. You need to make sure there is an active route to the remote gateway through this interface, otherwise the tunnel won't come up.
- **Local Gateway**: enable this setting when the interface where the tunnel terminates has multiple addresses assigned, and you want to specify which address to use for the tunnel. When you enable this setting, you see three options: **Primary IP**, **Secondary IP**, and **Specify**. Select **Specify** if you want to use an IP address different from the primary or secondary IP address.
- **Mode Config**: Enables automatic configuration through IKE. FortiGate acts as an *IKE mode config client* when you enable **Mode Config** and you set **Remote Gateway** to either **Static IP address** or **Dynamic DNS**. If you set **Remote Gateway** to **Dialup User**, FortiGate acts as an *IKE mode config server,* and more configuration options appear on the GUI. You will learn more about **Mode Config** in this lesson.



The following are the other options available on the GUI in the **Network** section:

- **NAT Traversal**: The option controls the behavior for NAT traversal. You will learn more about NAT traversal later in this lesson.
- **Keepalive Frequency**: When you enable NAT traversal, FortiGate sends keepalive probes at the configured frequency.
- **Dead Peer Detection**: Use dead peer detection (DPD) to detect dead tunnels. There are three DPD modes. **On Demand** is the default mode. You will learn more about DPD later in this lesson.
- **Forward Error Correction**: Forward error correction (FEC) is a technique that you can use to reduce the number of retransmissions in IPsec tunnels established over noisy links, at the expense of using more bandwidth. You can enable FEC on egress and ingress, and it is only supported when you disable IPsec hardware offloading. You will learn more about IPsec hardware offloading later in this lesson.
- **Advanced**:
  - **Add route**: Disable this setting if you are using a dynamic routing protocol over IPsec and do not want FortiGate to automatically add static routes.
  - **Auto discovery sender**: Enable this setting on a hub if you want the hub to facilitate ADVPN shortcut negotiation for spokes. When enabled, the hub sends a shortcut offer to the spoke to indicate that it can establish a shortcut to the remote spoke.
  - **Auto discovery receiver**: Enable this setting on a spoke if you want the spoke to negotiate an ADVPN shortcut.
  - **Exchange interface IP**: Enable this setting to allow the exchange of IPsec interface IP addresses. This allows a point-to-multipoint connection between the hub and spokes..
  - **Device creation**: Enable this setting to instruct FortiOS to create an interface for every dial-up client. To increase performance, disable this setting in dial-up servers with many dial-up clients.
  - **Aggregate member**: FortiGate allows you to aggregate multiple IPsec tunnels into a single interface. Enable this option if you want the tunnel to become an aggregate member.



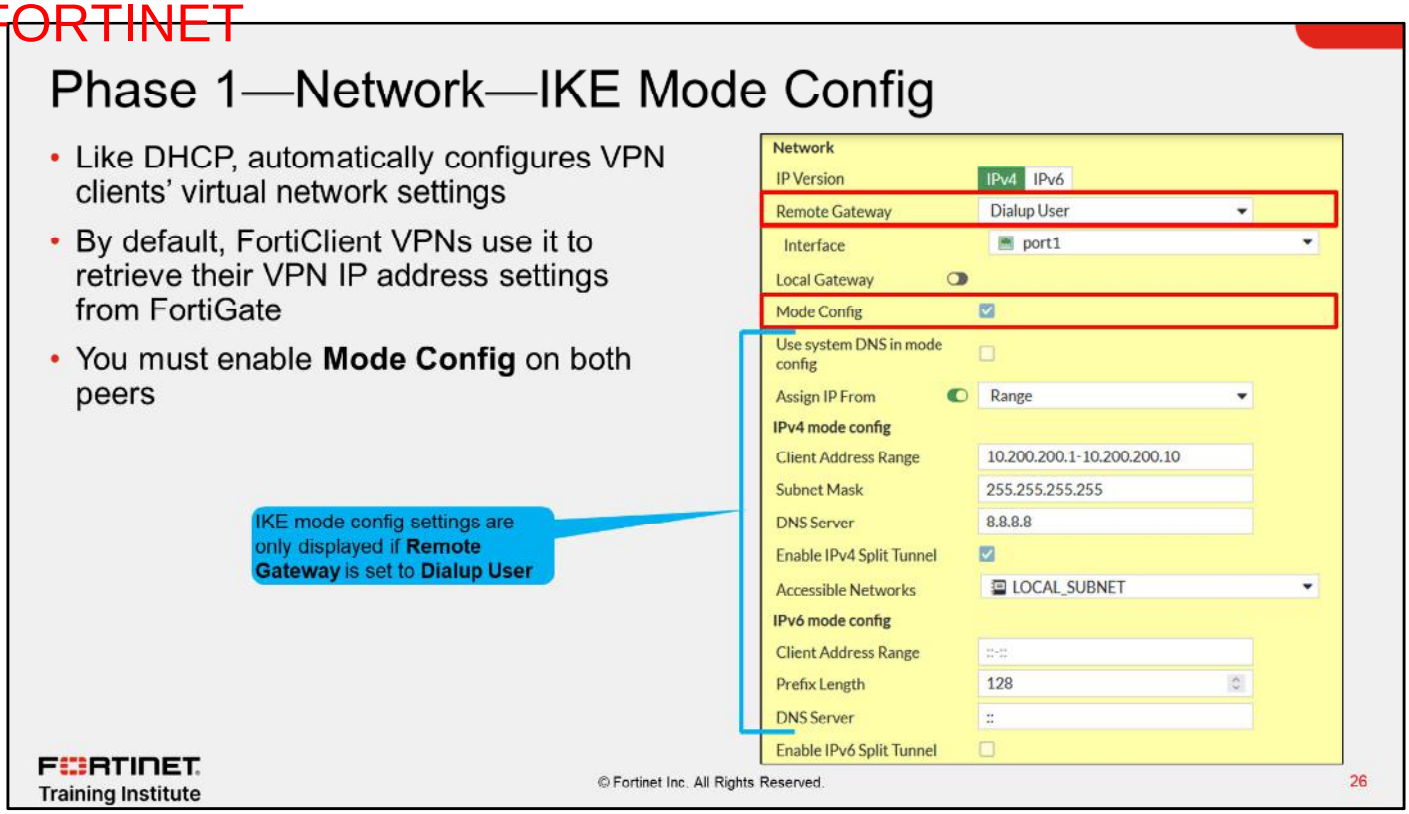
You have three options when configuring the remote gateway type of your VPN: **Dialup User**, **Static IP Address**, and **Dynamic DNS**.

Use **Dialup User** when the remote peer IP address is unknown. The remote peer whose IP address is unknown acts as the dial-up client, and this is often the case for branch offices and mobile VPN clients that use dynamic IP addresses, and no dynamic DNS. The dial-up client must know the IP address or FQDN of the remote gateway, which acts as the dial-up server. Because the dial-up server doesn't know the remote peer address, only the dial-up client can initiate the VPN tunnel.

Usually, dial-up clients are remote and mobile employees with FortiClient on their computer or handheld devices. You can also have a FortiGate device acting as a dial-up client for a remote office. One dial-up server configuration on FortiGate can be used for multiple IPsec tunnels from many remote offices or users.

Use **Static IP Address** or **Dynamic DNS** when you know the remote peer address. If you select **Static IP Address**, then you need to provide an IP address. If you select **Dynamic DNS**, then you need to provide a fully qualified domain name (FQDN), and make sure FortiGate can resolve that FQDN. When both peers know the remote peer address, that is, the remote gateway on both peers is set to **Static IP Address** or **Dynamic DNS**, then any peer can initiate the VPN tunnel.

Note that in a dial-up setup, the dial-up client is just a VPN peer with the remote gateway set to **static IP address** or **dynamic DNS**. When setting your VPN, you can combine different types of remote gateways. For obvious reasons, a tunnel in which both peers has the remote gateway set to **Dialup user** won't work.



IKE **Mode Config** is similar to DHCP because a server assigns network settings such as IP address, netmask, and DNS servers, to clients. This assignment takes place over IKE messages.

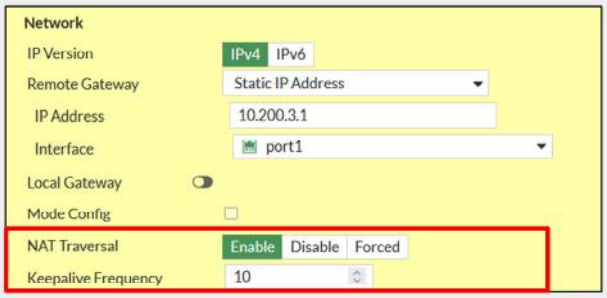
When you enable **Mode Config** on a FortiGate device acting as dial-up server, it pushes network settings to dial-up clients. The dial-up clients are usually FortiClient peers, but they can also be FortiGate peers.

For IKE mode config to work, you must enable the feature on both peers. On FortiClient, **Mode Config** is enabled by default, but on FortiGate, you must manually enable it.

Note that the IKE **Mode Config** settings, are displayed on the GUI only when you set **Remote Gateway** to **Dialup User**. On the FortiGate device acting as dial-up client, you can select **Mode Config** on the GUI, but the additional settings are not displayed.

Phase 1—Network—NAT Traversal (NAT-T)

- ESP can't support NAT because it has no port numbers
- If **NAT Traversal** is set to **Enable**, it detects whether NAT devices exist on the path
  - If yes, both ESP and IKE use UDP port 4500
  - Recommended if the initiator or responder is behind NAT
- If **NAT Traversal** is set to **Forced**:
  - ESP and IKE always use UDP port 4500, even when there are no NAT devices on the path
- Keepalive probes are sent frequently to keep the connection across the routers active



FORTINET Training Institute

27

The ESP protocol usually has problems crossing devices that are performing NAT. One of the reasons is that ESP does not use port numbers, like TCP and UDP do, to differentiate one tunnel from another.

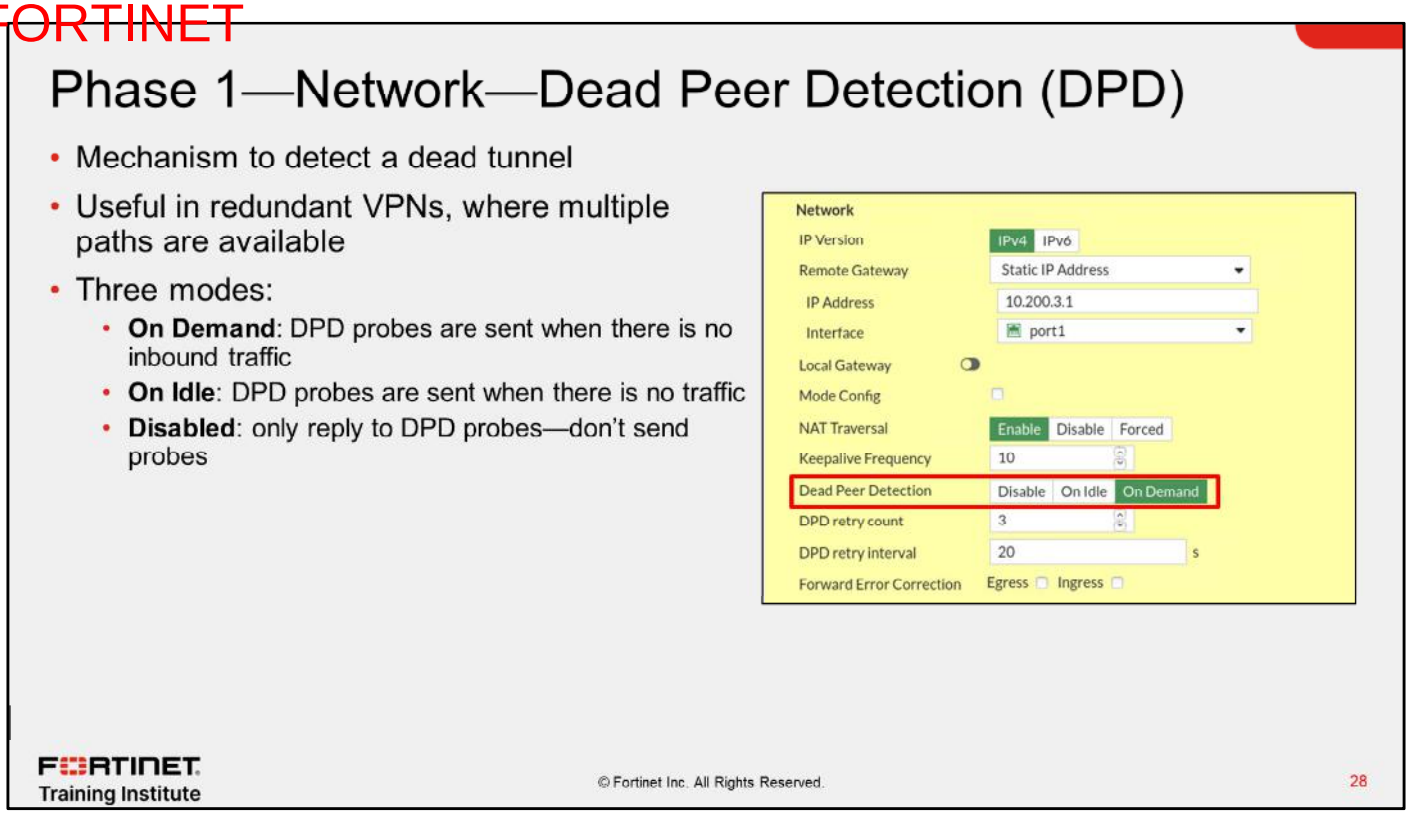
To solve this, NAT transversal (NAT-T) was added to the IPsec specifications. When NAT-T is enabled on both ends, peers can detect any NAT device along the path. If NAT is found, then the following occurs on both peers:

- IKE negotiation switches to using UDP port 4500.
- ESP packets are encapsulated in UDP port 4500.

So, if you have two FortiGate devices that are behind, for example, an ISP modem that performs NAT, you will probably need to enable this setting.

When you set the **NAT Traversal** setting to **Forced**, UDP port 4500 is always used, even when there is no NAT device along the path.

When you enable NAT-T, the **Keepalive Frequency** option shows the interval (in seconds) at which FortiGate sends keepalive probes. You need NAT-T when there is one or more routers along the path performing NAT. The purpose of the keepalive probes is to keep the IPsec connection active across those routers along the path.



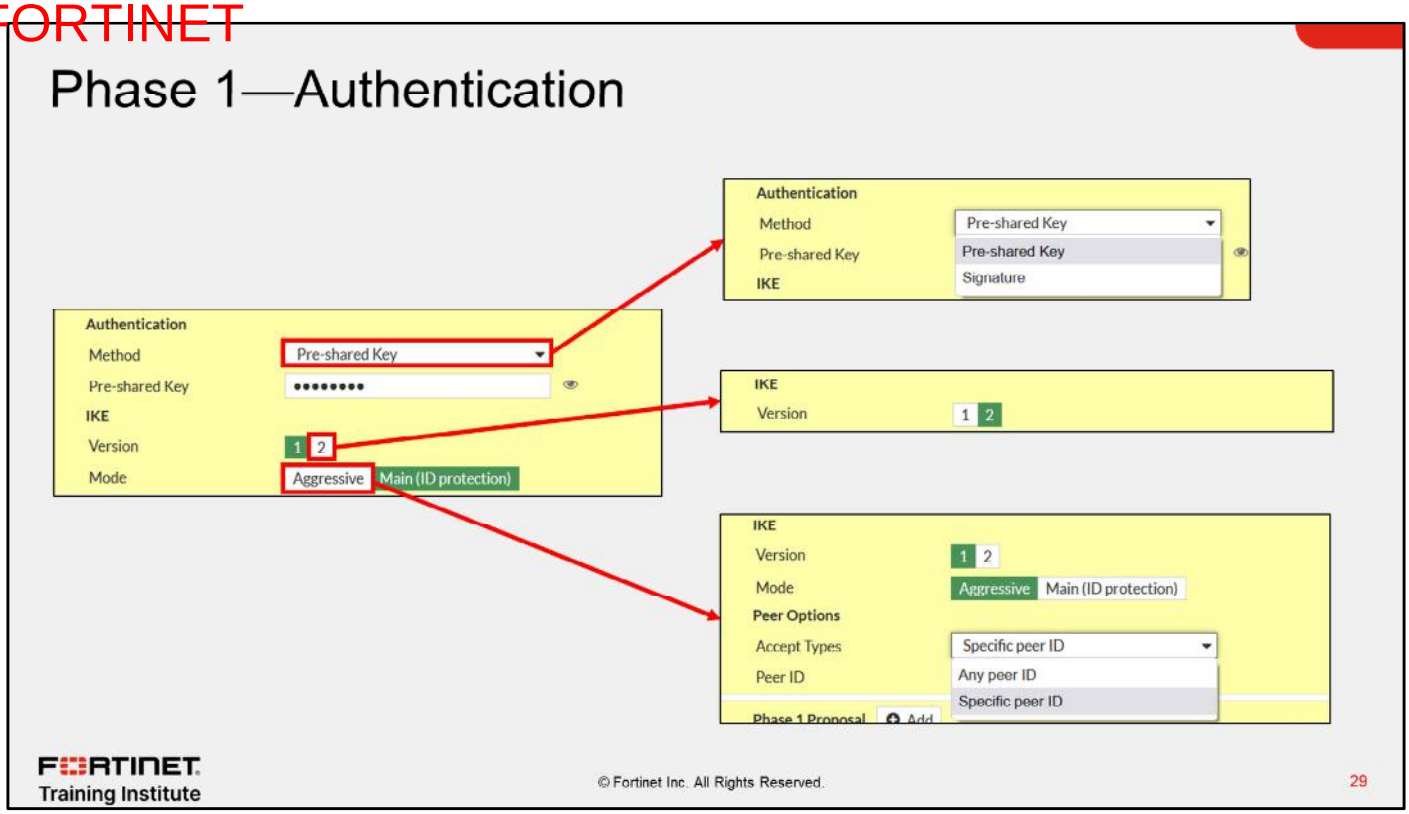
After the peers negotiate the IPsec SAs of a tunnel and, therefore, the tunnel is considered up, the peers usually don't negotiate another IPsec SA until it expires. In most cases, the IPsec SA expires every few hours. This means that if there is a network disruption along the path of the tunnel before the IPsec SA expires, the peers will continue to send traffic through the tunnel even though the communication between the sites is disrupted.

When you enable DPD, DPD probes are sent to detect a failed (or dead) tunnel and bring it down before its IPsec SAs expire. This failure detection mechanism is very useful when you have redundant paths to the same destination, and you want to fail over to a backup connection when the primary connection fails to keep the connectivity between the sites up.

FortiGate supports three DPD modes:

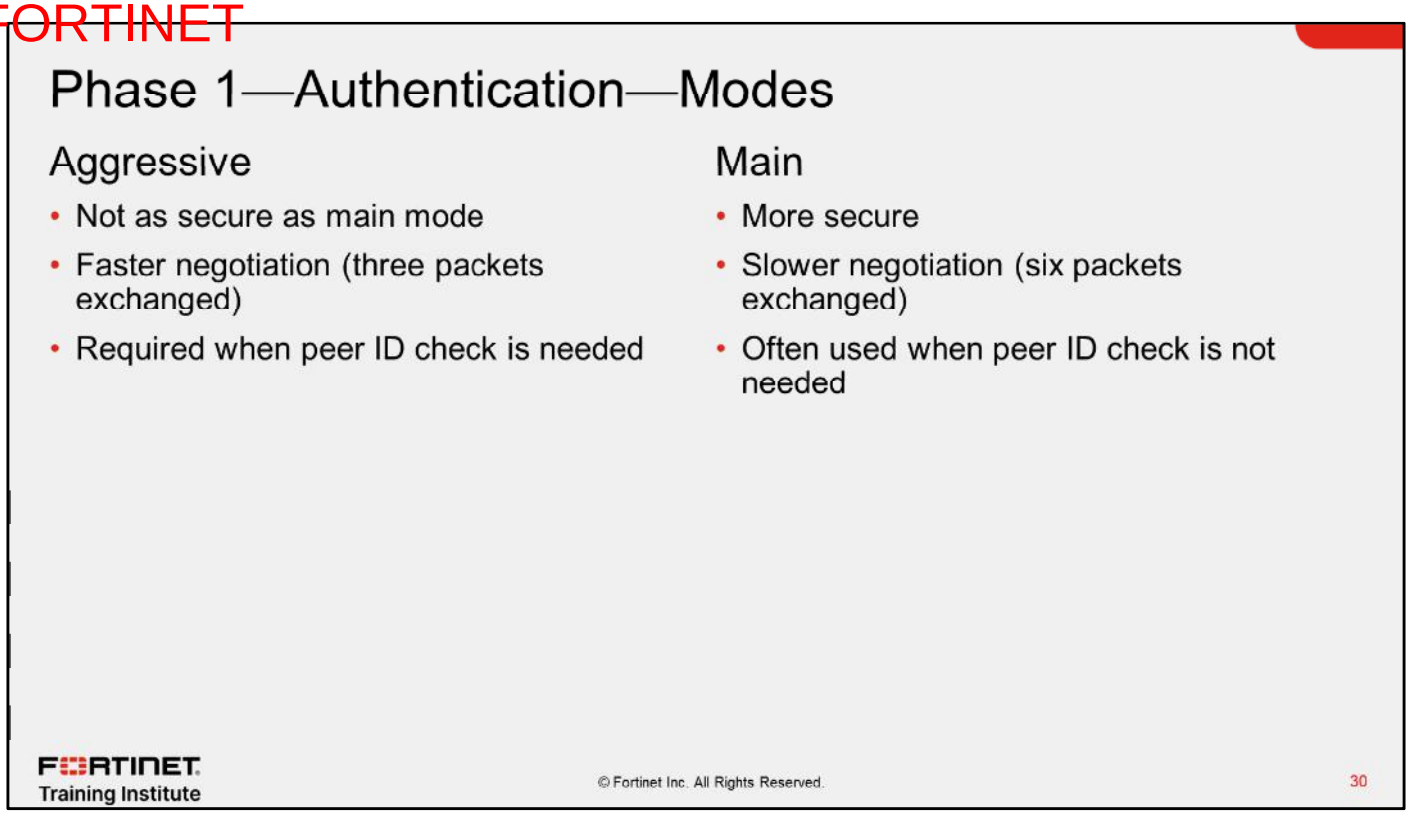
- **On Demand**: FortiGate sends DPD probes if there is only outbound traffic through the tunnel, but no inbound. Because network applications are usually bidirectional, observing only traffic on the outbound direction could be an indication of a network failure.
- **On Idle**: FortiGate sends DPD probes when no traffic is observed in the tunnel. An idle tunnel does not necessarily mean the tunnel is dead. Avoid this mode if you have many tunnels, because the overhead introduced by DPD can be very resource intensive.
- **Disabled**: FortiGate replies only to DPD probes received. FortiGate never sends DPD probes to the remote peer and therefore cannot detect a dead tunnel.

The default DPD mode is **On Demand**. In terms of scalability, **On Demand** is a better option than **On Idle**.



Now, you will learn about the **Authentication** section in phase 1 configuration:

- **Method**: FortiGate supports two authentication methods: **Pre-shared Key** and **Signature**. When you select **Pre-shared Key**, you must configure both peers with the same pre-shared key. When you select **Signature**, phase 1 authentication is based on digital certificate signatures. Under this method, the digital signature on one peer is validated by the presence of the CA certificate installed on the other peer. That is, on the local peer, you need to install both the local peer's certificate and the CA certificate that issued the remote peer certificate.
- **Version**: allows you to select the IKE version to use. When selecting version **2**, aggressive and main modes disappear because they don't apply to IKEv2.
- **Mode**: refers to the IKEv1 mode. Two options are available: **Aggressive** and **Main (ID protection)**. You will learn more about these modes in this lesson.



IKE supports two different negotiation modes: main and aggressive. Which one should you use?

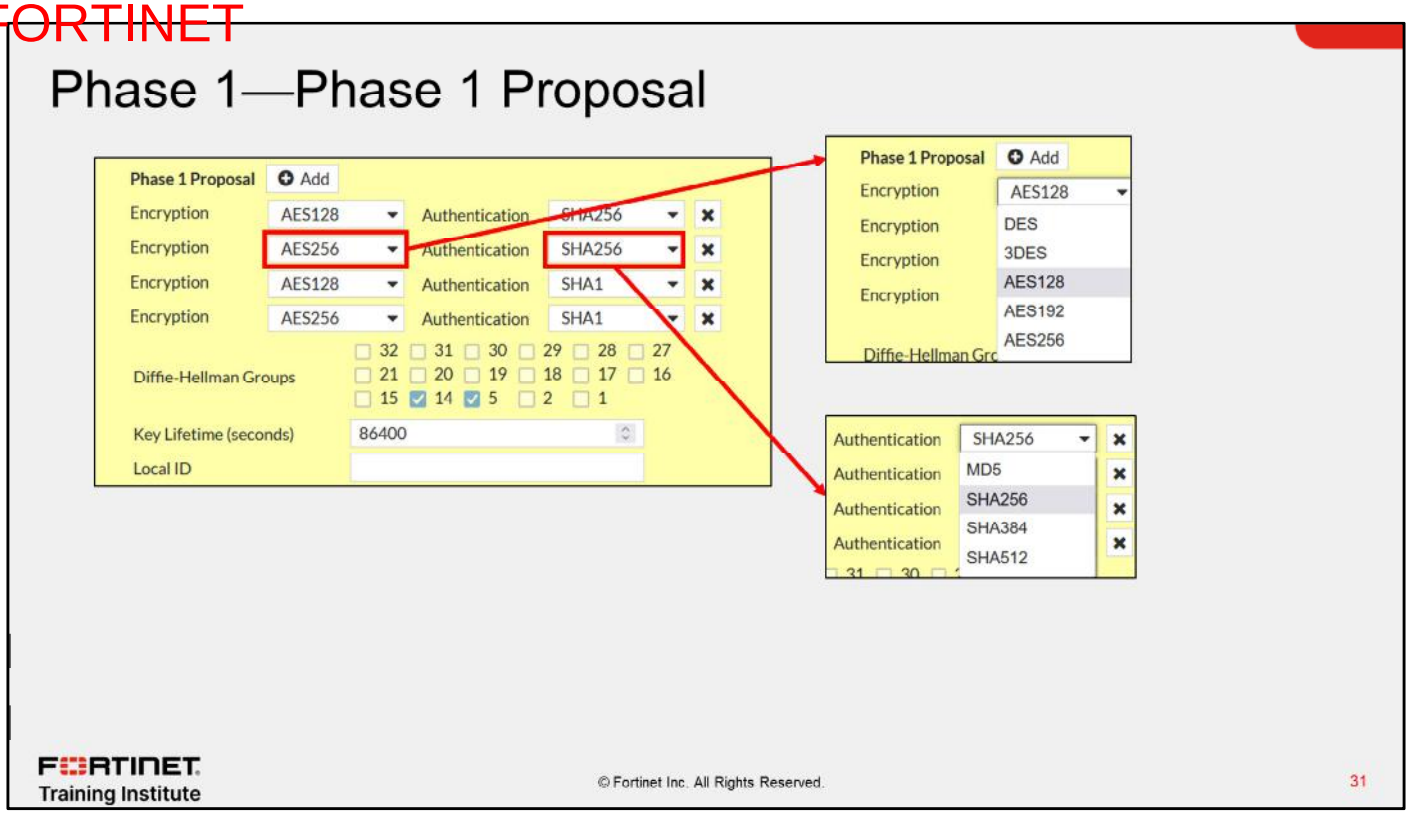
To answer that question, we can analyze three categories: security, performance, and deployment.

Security wise, main mode is considered more secure because the pre-shared key hash is exchanged encrypted, whereas in aggressive mode, the hash is exchanged unencrypted. Although the attacker would still have to guess the cleartext pre-shared key for the attack to be successful, the fact that the pre-shared key hash has been encrypted in main mode reduces considerably the chances of a successful attack.

In terms of performance, aggressive mode may be a better option. This is because the negotiation is completed after only three packets are exchanged, whereas in main mode, six packets are exchanged. For this reason, you may want to use aggressive mode when a great number of tunnels terminate on the same FortiGate device, and performance is a concern.

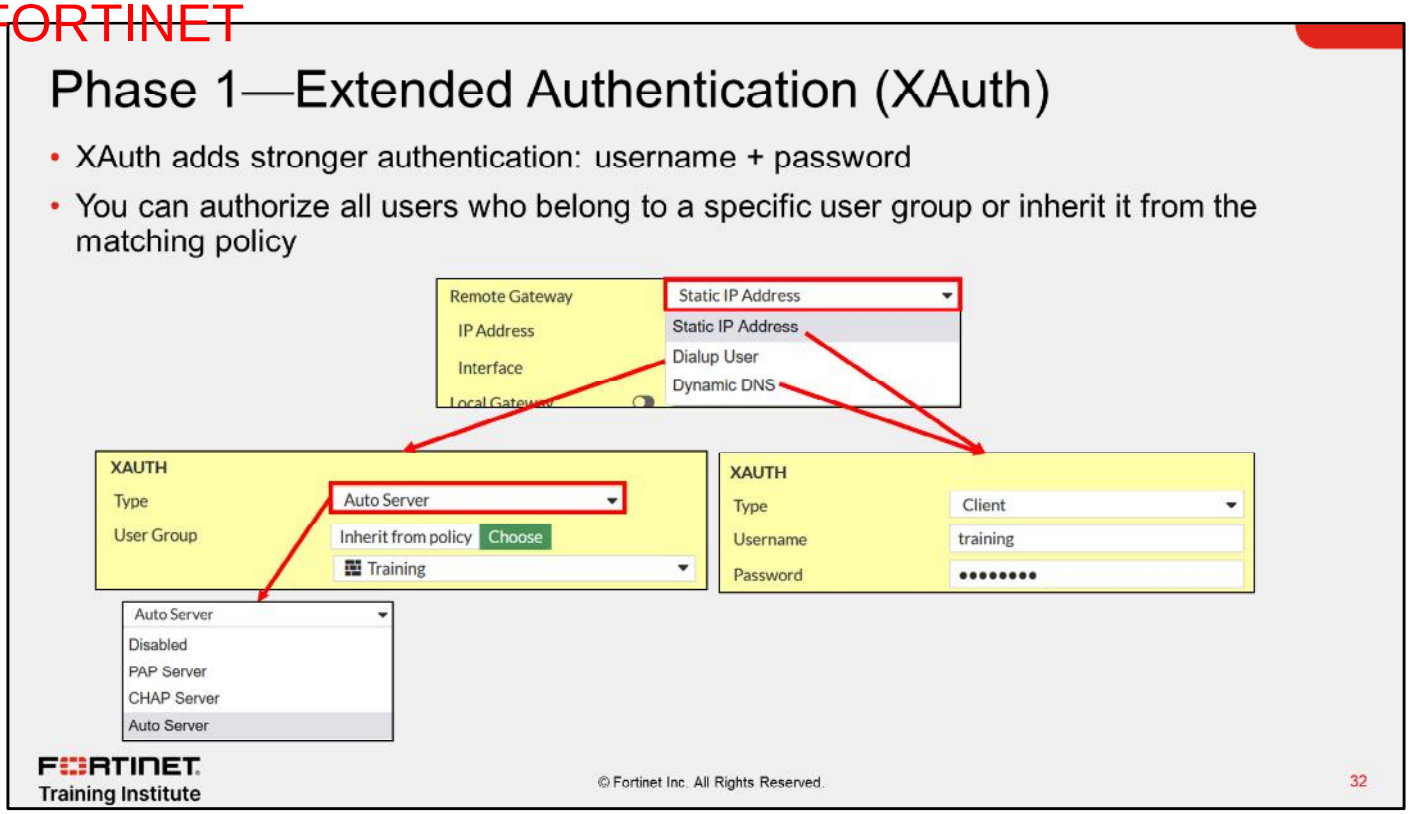
Another use case for aggressive mode, is when there is more than one dial-up tunnel terminating on the same FortiGate IP address, and the remote peer is authenticated using a peer ID because its IP address is dynamic. Because peer ID information is sent in the first packet in an aggressive mode negotiation, then FortiGate can match the remote peer with the correct dial-up tunnel. The latter is not possible in main mode because the peer ID information is sent in the last packet, and after the tunnel has been identified.

When both peers know each other's IP address or FQDN, you may want to use main mode to take advantage of its more secure negotiation. In this case, FortiGate can identify the remote peer by its IP address and, as a result, associate it with the correct IPsec tunnel.



Now, you will learn about the **Phase 1 Proposal** section of phase 1 configuration. This section allows you to enable the different proposals that FortiGate supports when negotiating the IKE SA (or phase 1 SA). You can combine different parameters to suit your security needs. You must at least configure one combination of encryption and authentication algorithms, or several.

- **Encryption**: select the algorithm to use for encrypting and decrypting the data.
- **Authentication**: select the authentication algorithm to use for verifying the integrity and authenticity of the data.
- **Diffie-Hellman Groups**: The Diffie-Hellman (DH) algorithm is used during IKE SA negotiation. The use of DH in phase 1 is mandatory and can't be disabled. You must select at least one DH group. The higher the DH group number, the more secure the phase 1 negotiation is. However, a higher DH group number also results in a longer compute time.
- **Key Lifetime**: defines the lifetime of the IKE SA. At the end of the lifetime, a new IKE SA is negotiated.
- **Local ID**: if the peer accepts a specific peer ID, type that same peer ID in this field.



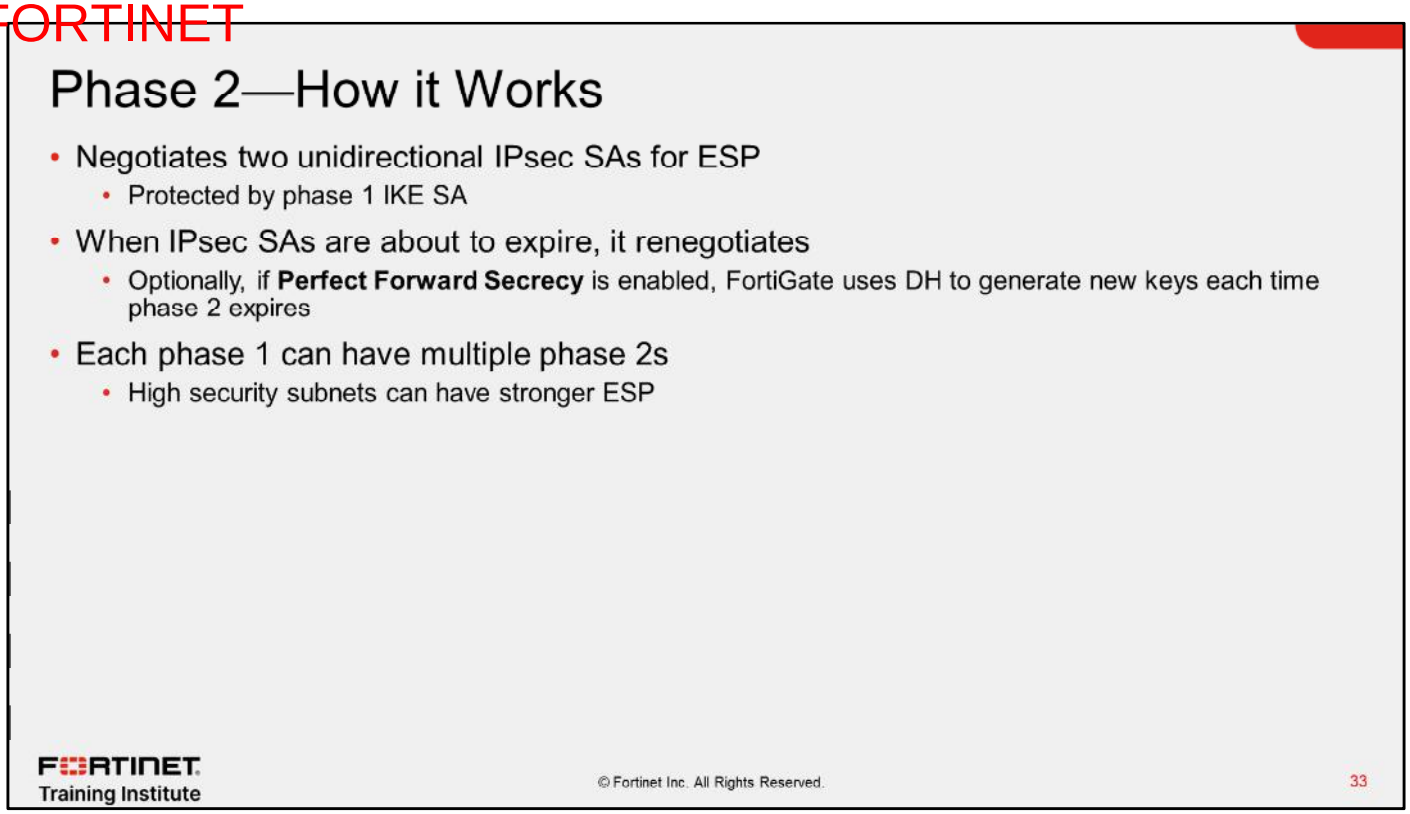
Phase 1 supports two types of authentication: pre-shared keys and digital signatures. The XAuth extension, sometimes called phase 1.5, forces remote users to authenticate additionally with their credentials (username and password). So, additional authentication packets are exchanged if you enable it. What is the benefit? Stronger authentication.

When you set **Remote Gateway** to **Dialup User**, FortiGate acts as the authentication server. The **XAUTH** section shows the authentication server type options: **PAP Server**, **CHAP Server**, and **Auto Server**. In the example shown on this slide, **Auto Server** is selected, which means that FortiGate automatically detects the authentication protocol used by the client.

After you select the authentication server type, you configure how user group matching is performed. There are two options: **Inherit from policy** and **Choose**. The latter is used in the example on this slide, and allows you to select one of the user groups available on FortiGate. Note that, when you select **Choose**, you must configure a separate dial-up VPN for every group of users that require a different network access policy.

The other way to authenticate VPN users with XAuth is by selecting **Inherit from policy**. When you select this option, FortiGate authenticates users based on their matching IPsec policy and, as a result, the configuration for controlling network access is simpler. That is, you control network access by configuring multiple policies for different user groups, instead of configuring multiple tunnels for different user groups. The **Inherit from policy** option follows a similar authentication approach used for SSL VPN remote users. You will learn more about SSL VPN in another lesson.

When **Remote Gateway** is set to **Static IP Address** or **Dynamic DNS**, FortiGate acts as the client, and the **XAUTH** section shows the **Client** option as **Type**. You can then set the credentials that FortiGate uses to authenticate against the remote peer through XAuth.

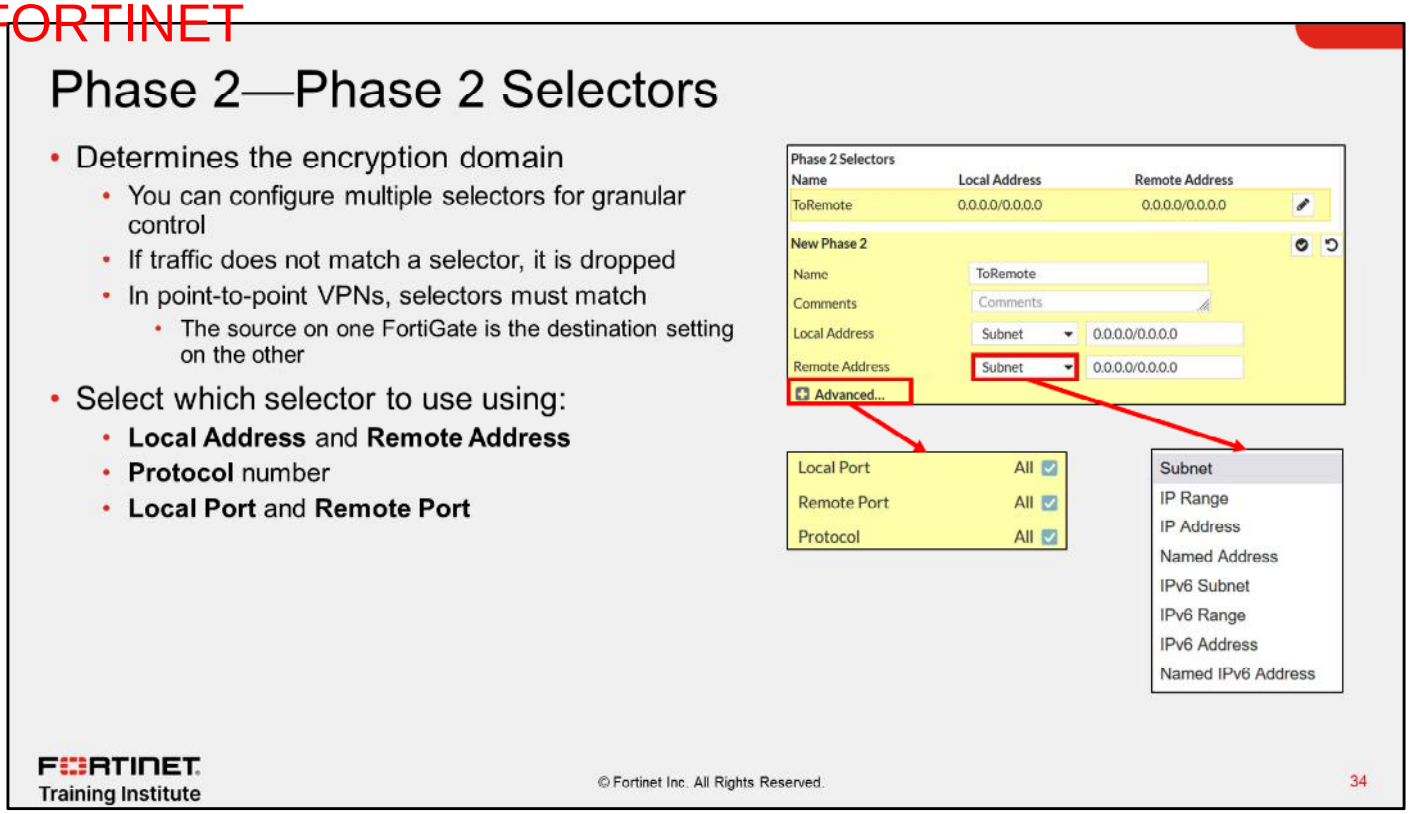


After phase 1 has established a secure channel to exchange data, phase 2 begins.

Phase 2 negotiates security parameters for two IPsec SAs over the secure channel established during phase 1. ESP uses IPsec SAs to encrypt and decrypt the traffic exchanged between sites.

Phase 2 does not end when ESP begins. Phase 2 periodically renegotiates IPsec SAs to maintain security. If you enable **Perfect Forward Secrecy**, each time phase 2 expires, FortiGate uses DH to recalculate new secret keys. In this way, new keys are not derived from older keys, making it much harder for an attacker to crack the tunnel.

Each phase 1 can have multiple phase 2s. When would this happen? For example, you may want to use different encryption keys for each subnet whose traffic is crossing the tunnel. How does FortiGate select which phase 2 to use? By checking which phase 2 selector (or quick mode selector) matches the traffic.



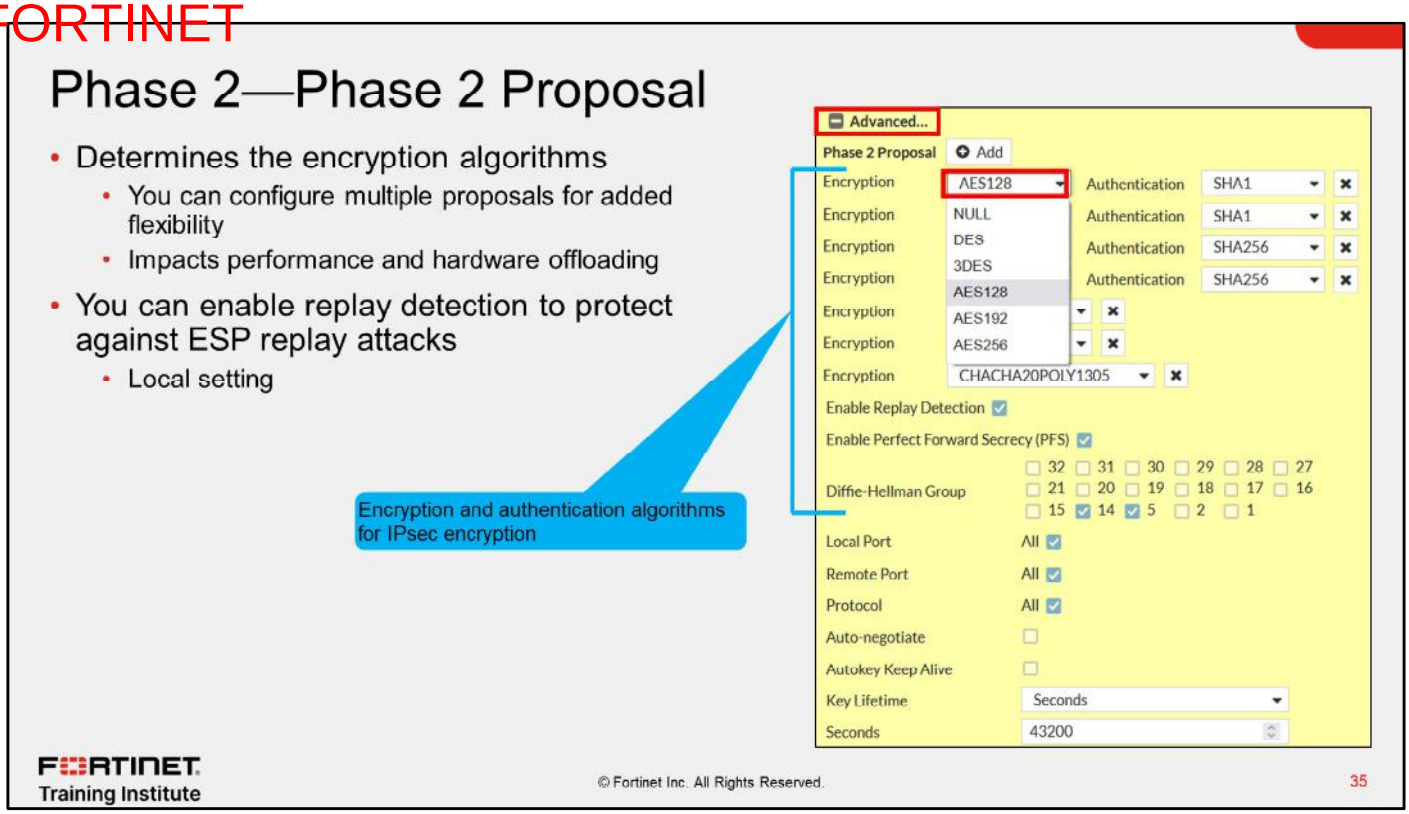
In phase 2, you must define the encryption domain (or interesting traffic) of your IPsec tunnel. The encryption domain refers to the traffic that you want to protect with IPsec, and it is determined by your phase 2 selector configuration.

You can configure multiple selectors to have more granular control over traffic. When you configure a phase 2 selector, you specify the encryption domain by indicating the following network parameters:

- **Local Address** and **Remote Address**: as seen in the example shown on this slide, you can define IPv4 or IPv6 addresses using different address scopes. When selecting **Named Address** or **Named IPv6 Address**, FortiGate allows you to select an IPv4 or IPv6 firewall address object, respectively, configured in the system.
- **Protocol**: is in the **Advanced** section, and is set to **All** by default.
- **Local Port** and **Remote Port**: are also shown in the **Advanced** section, and are set to **All** by default. This applies only to port-based traffic such as TCP or UDP. You will learn more about the **Advanced** section later in this lesson.

Note that after the traffic is accepted by a firewall policy, traffic is dropped before entering the IPsec tunnel if the traffic does not match any of the phase 2 selectors configured. For this reason, usually, it's more intuitive to filter traffic with firewall policies. So, if you don't want to use phase 2 selector filtering, you can just create one phase 2 selector with both the local and remote addresses set to any subnet, like in the example shown on this slide, and then use firewall policies to control which traffic is accepted on the IPsec tunnel.

In addition, the phase 2 selector network parameters on both peers must match if the tunnel is point-to-point, that is, when the remote gateway is *not* set to dial-up user.



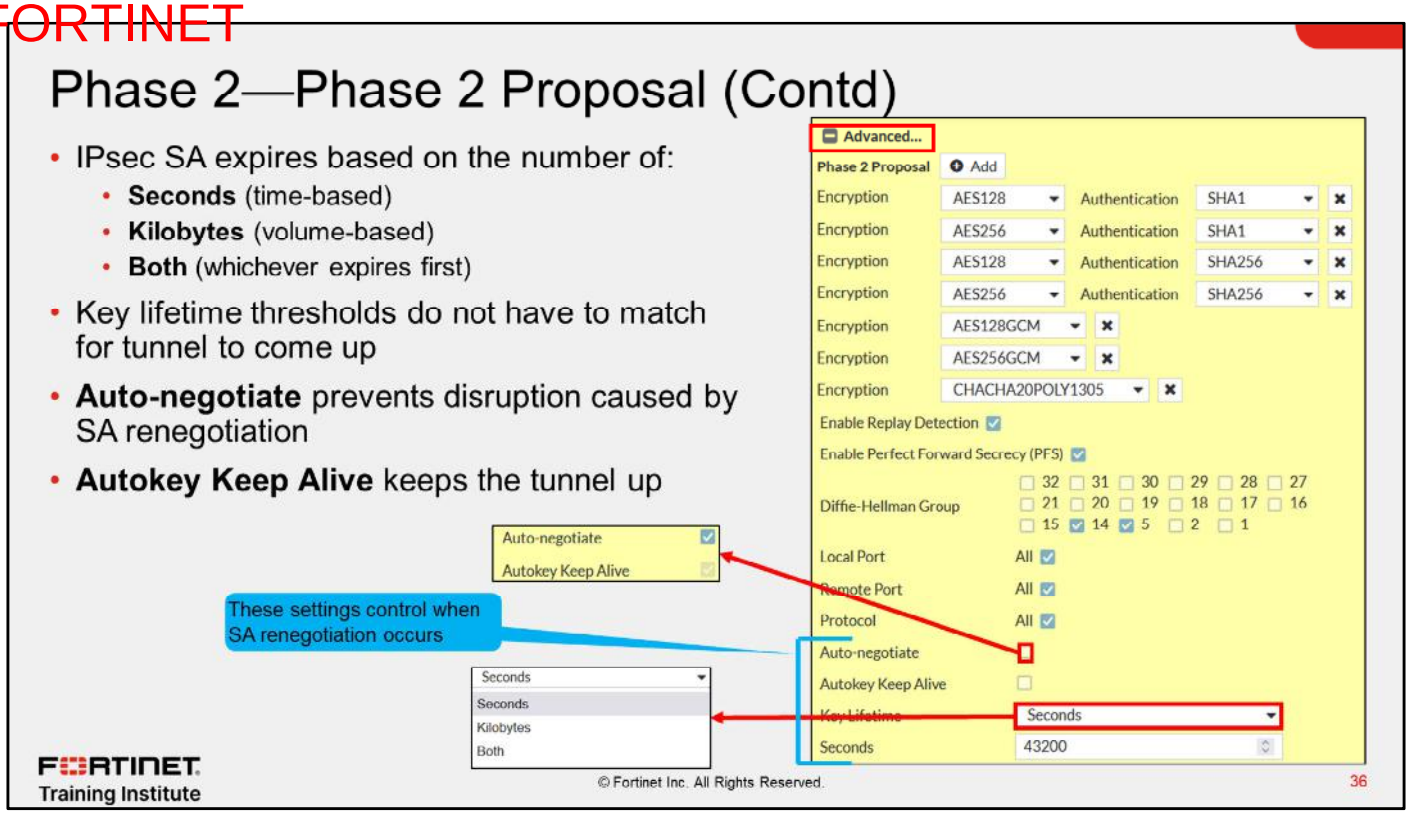
For every phase 2 selector, you need to configure one or more phase 2 proposals. A phase 2 proposal defines the algorithms supported by the peer for encrypting and decrypting the data over the tunnel. You can configure multiple proposals to offer more options to the remote peer when negotiating the IPsec SAs.

Like in phase 1, you need to select a combination of encryption and authentication algorithms. Some algorithms are considered more secure than others, so make sure to select the algorithms that conform with your security policy. However, note that the selection of the algorithms has a direct impact on FortiGate IPsec performance. For example, **3DES** is known to be a much more resource-intensive encryption algorithm than **DES** and **AES**, which means that your IPsec throughput could be negatively impacted if you select **3DES** as the encryption algorithm. Also, note that if you select **NULL** as the encryption algorithm, traffic is not encrypted.

In addition, some encryption algorithms, such as **CHACHA20POLY1305**, are not supported for hardware offload. That is, if you have a FortiGate device that contains network processor (NP) units, you can achieve higher IPsec performance if you select an algorithm that is supported for IPsec offload by your NP unit model, such as AES or DES. For a list of supported encryption algorithms for IPsec hardware offloading, refer to `https://docs.fortinet.com`.

When configuring the phase 2 proposal, you can select **Enable Replay Detection** to detect antireplay attacks on ESP packets. Note that this is a local setting and, therefore, it is not included as part of the proposals presented by the peer during phase 2 negotiation.

Also, if you enable **Perfect Forward Secrecy**, FortiGate uses DH to enhance security during the negotiation of IPsec SAs.

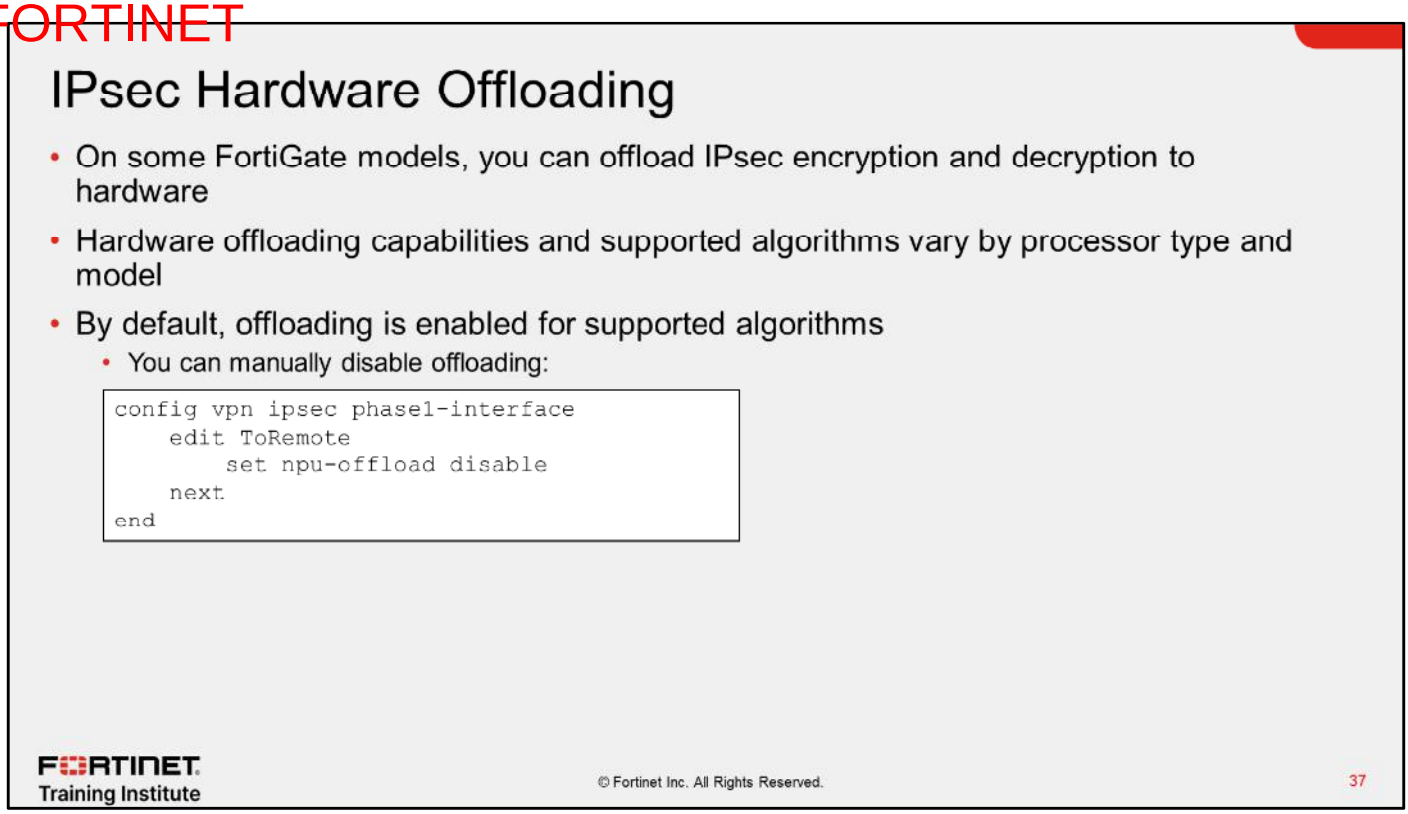


IPsec SAs are periodically renegotiated to improve security, but when does that happen? It depends on the key lifetime settings configured on the phase 2 proposal.

The expiration of an IPsec SA is determined by the lifetime type and threshold configured. By default, **Key Lifetime** is set to **Seconds** (time-based). This means that when the SA duration reaches the number of seconds set as **Seconds**, the SA is considered expired. You can also set the key lifetime to **Kilobytes** (volume-based), upon which the SA expires after the amount of traffic encrypted and decrypted using that SA reaches the threshold set. Alternatively, you can select **Both** as the key lifetime type, upon which FortiGate tracks both the duration of the SA and the amount of traffic. Then, when any of the two thresholds is reached, the SA is considered expired. Note that the key lifetime thresholds do not have to match for the tunnel to come up. When thresholds are different, the peers agree on using the lowest threshold value offered between the two.

When IPsec SAs expire, FortiGate needs to negotiate new SAs to continue sending and receiving traffic over the IPsec tunnel. Technically, FortiGate deletes the expired SAs from the respective phase 2 selectors, and installs new ones. If IPsec SA renegotiation takes too much time, then FortiGate might drop interesting traffic because of the absence of active SAs. To prevent this, you can enable **Auto-negotiate**. When you do this, FortiGate not only negotiates new SAs before the current SAs expire, but it also starts using the new SAs right away. The latter prevents traffic disruption by IPsec SA renegotiation.

Another benefit of enabling **Auto-negotiate** is that the tunnel comes up and stays up automatically, even when there is no interesting traffic. When you enable **Autokey Keep Alive** and keep **Auto-negotiate** disabled, the tunnel does not come up automatically unless there is interesting traffic. However, after the tunnel is up, it stays that way because FortiGate periodically sends keep alive packets over the tunnel. Note that when you enable **Auto-negotiate**, **Autokey Keep Alive** is implicitly enabled.



On some FortiGate models, you can offload the encryption and decryption of IPsec traffic to hardware. The algorithms that are supported depend on the NP unit model present on FortiGate. For a list of supported encryption algorithms for IPsec hardware offloading, refer to `https://docs.fortinet.com`.

By default, hardware offloading is enabled for the supported algorithms. This slide shows the commands you can use to disable hardware offloading per tunnel, if necessary.

Knowledge Check

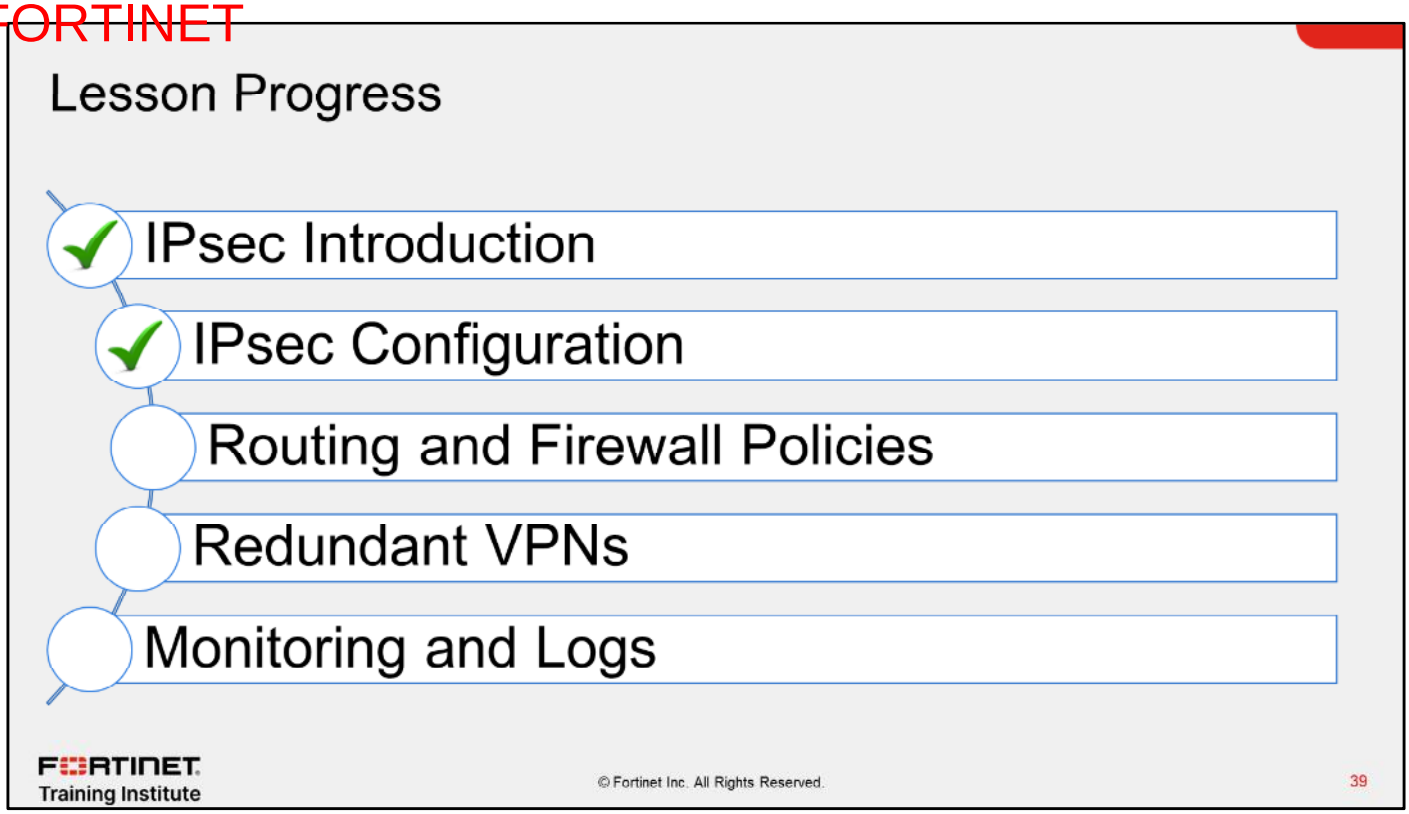
1. Which type of VPN peer can initiate a VPN tunnel?
   A. Dial-up server
   ✓B. Dial-up client

2. On which phase do you configure the algorithms used for traffic encryption?
   A. Phase 1
   ✓B. Phase 2

3. Which IKEv1 negotiation mode is faster?
   ✓A. Aggressive
   B. Main

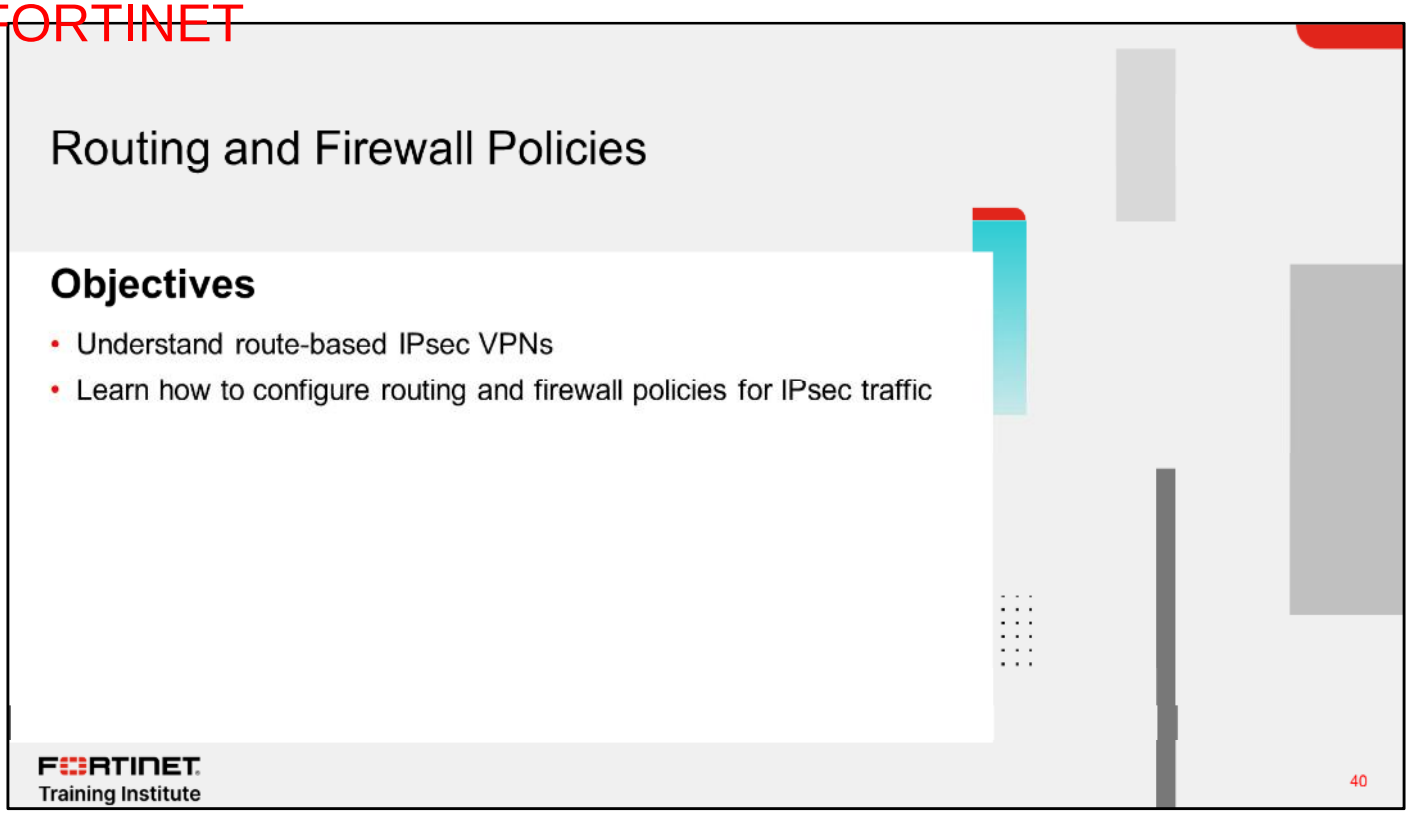
FORTINET Training Institute

38



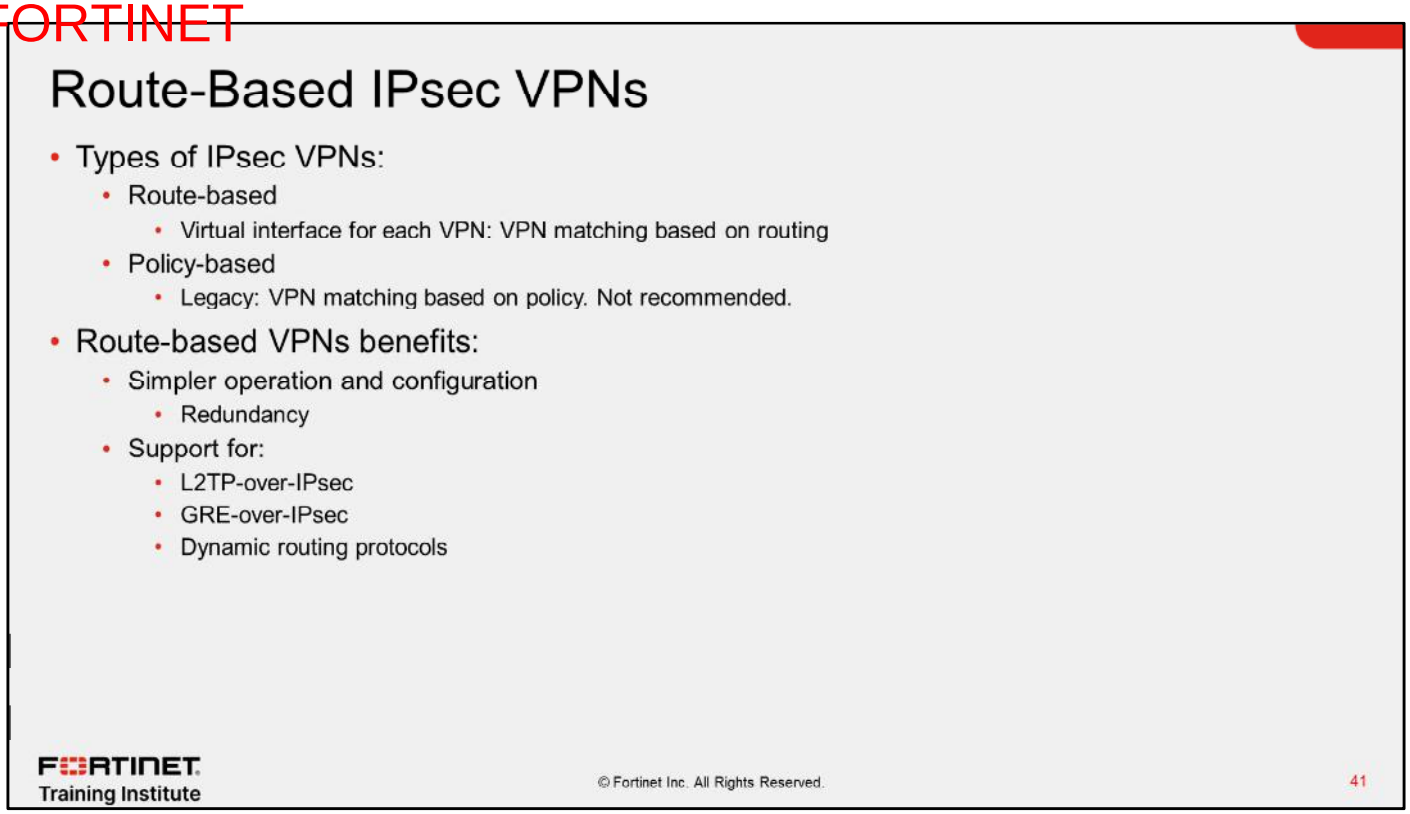
Good job! You now understand IPsec configuration.

Now, you will learn about routing and firewall policies for IPsec traffic.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in routing and firewall policies for IPsec VPNs, you will be able to set up appropriate routing and firewall policies for your IPsec VPN deployment.



FortiGate supports two types of IPsec VPNs: route-based and policy-based. Policy-based is a legacy IPsec VPN that is supported only for backward compatibility reasons, and its use *is not recommended* for new deployments. Unless otherwise stated, all IPsec VPN references in this lesson are for route-based IPsec VPNs.

In a route-based IPsec VPN, FortiGate automatically adds a virtual interface with the VPN name. This means that not only can you configure routing and firewall polices for IPsec traffic in the same way you do for non-IPsec traffic, but you also can leverage the presence of multiple connections to the same destination to achieve redundancy.

Another benefit of route-based IPsec VPNs is that you can deploy variations of IPsec VPNs such as L2TP-over-IPsec and GRE-over-IPsec. In addition, you can also enable dynamic routing protocols for scalability purposes and best path selection.

Routes for IPsec VPNs

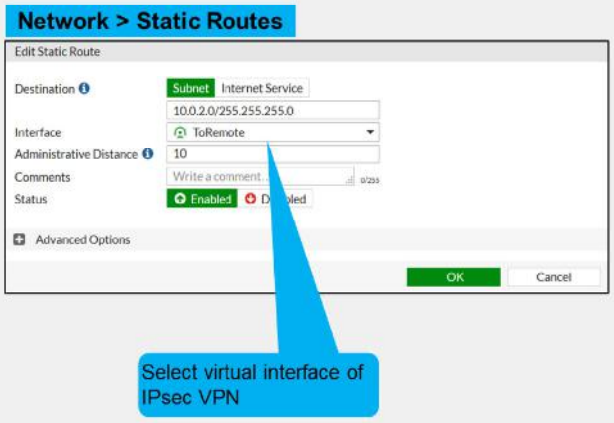
Dial-up user

```
config vpn ipsec phase1-interface
    edit "Dialup"
        set add-route enable | disable
    next
end
```

- `add-route` is enabled (default)
  - No need to configure static routes
  - Static routes are added after phase 2 is up
    - The destination is the local network presented by the dial-up client during phase 2 negotiation
    - The default route distance is 15
  - Static routes are deleted after phase 2 is down
- `add-route` is disabled
  - Useful when dynamic routing protocol is used
  - Dynamic routing protocol takes care of routing updates

Static IP address / dynamic DNS

- Static routes are needed



FORTINET Training Institute

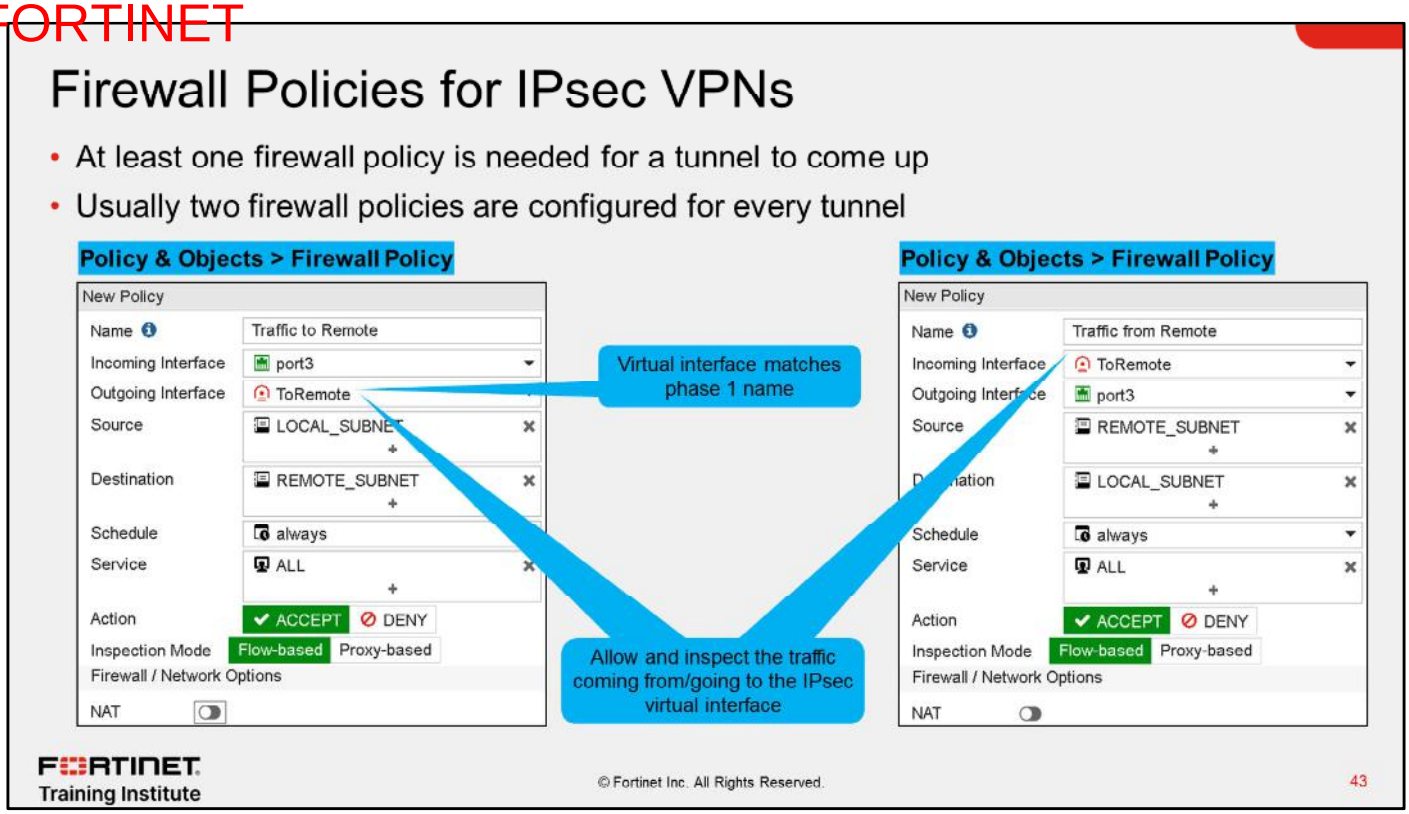
42

Although you can use dynamic routing protocols for IPsec VPNs, this lesson covers only the use of static routes.

The routing configuration needed for your IPsec VPN depends on the type of remote gateway configured. When you set the remote gateway to **Dialup User** and enable `add-route`, FortiGate automatically adds a static route for the local network presented by the remote peer during phase 2 negotiation. In addition, the route is added to the routing table only after phase 2 is up. If phase 2 goes down, the static route is removed from the routing table.

When you set the remote gateway to **Dialup User** and disable `add-route`, FortiGate does not add static routes automatically. In this case, a dynamic routing protocol is used between the remote peers to exchange routing information.

When the remote gateway is set to **Static IP Address** or **Dynamic DNS**, you must configure static routes. When you configure a static route, you select the virtual interface of the IPsec tunnel as the outgoing interface.



You must configure at least one firewall policy that accepts traffic on your IPsec tunnel. Otherwise, the tunnel will not come up.

When you configure firewall policies for non-IPsec traffic, the policy determines the direction of the traffic that initiates sessions. The same applies to IPsec traffic. For this reason, you usually want to configure at least two firewall policies for your IPsec VPN: one incoming policy and one outgoing policy. The incoming policy allows traffic initiated from the remote site, while the outgoing policy allows traffic to be initiated from the local network.

Note that the policies are configured with the virtual tunnel interface (or phase 1 name) as the incoming or outgoing interface.

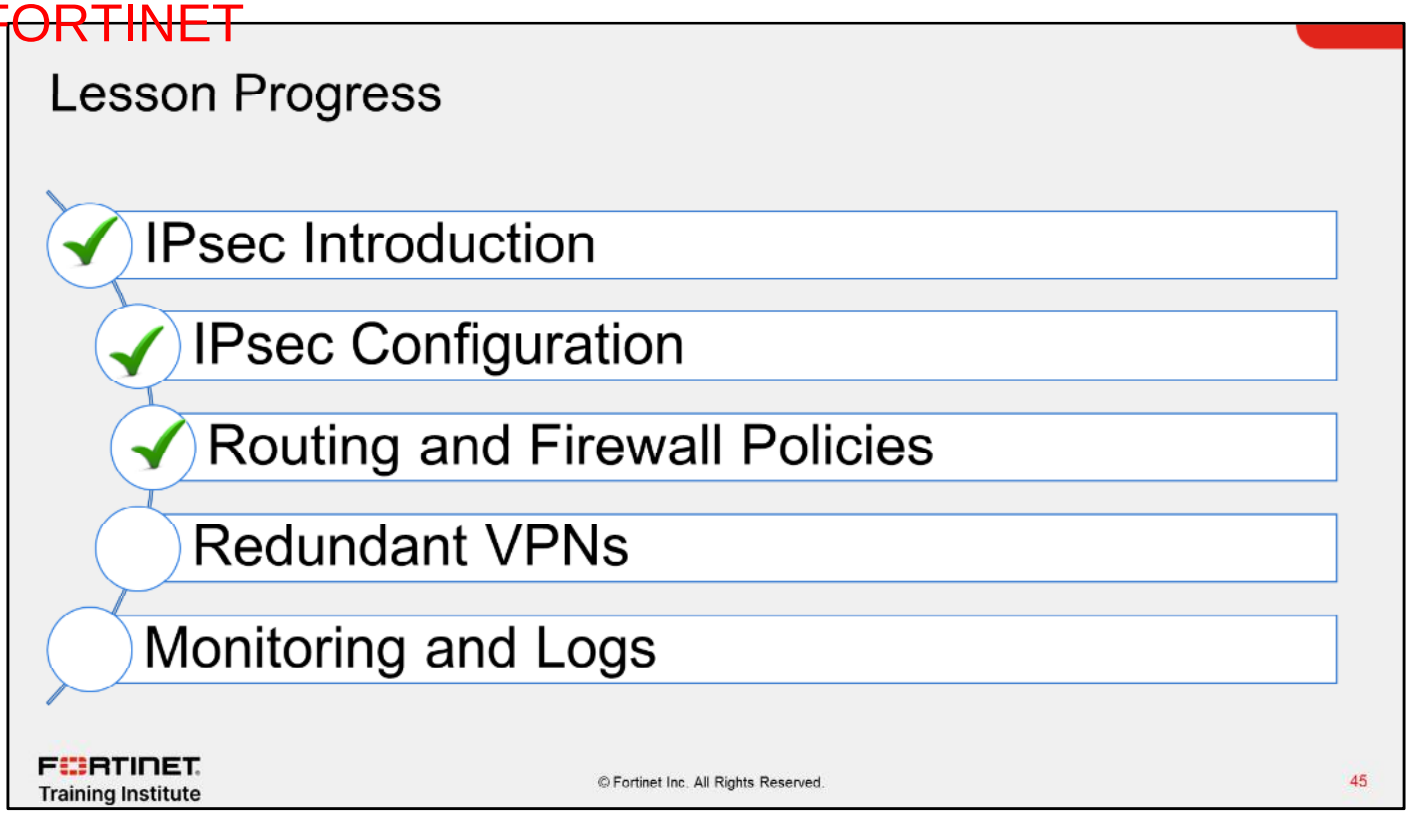
Knowledge Check

1. Which IPsec VPN type is legacy and not recommended for new deployments?
   - A. Route-based IPsec VPN
   - ✓ B. Policy-based IPsec VPN

2. What is a configuration requirement for an IPsec tunnel to come up?
   - ✓ A. A firewall policy accepting traffic on the IPsec tunnel
   - B. A route for IPsec traffic

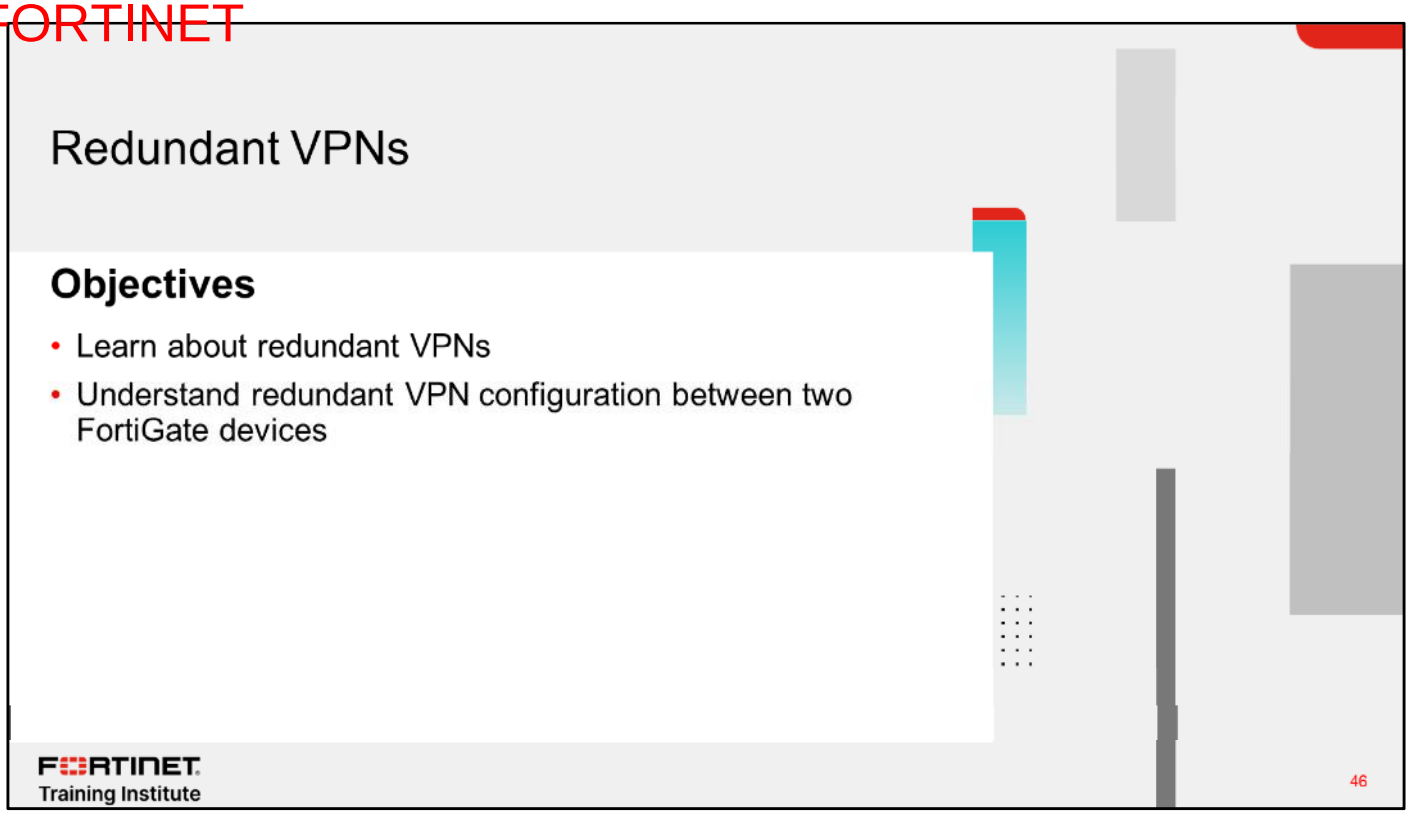
FORTINET Training Institute

44



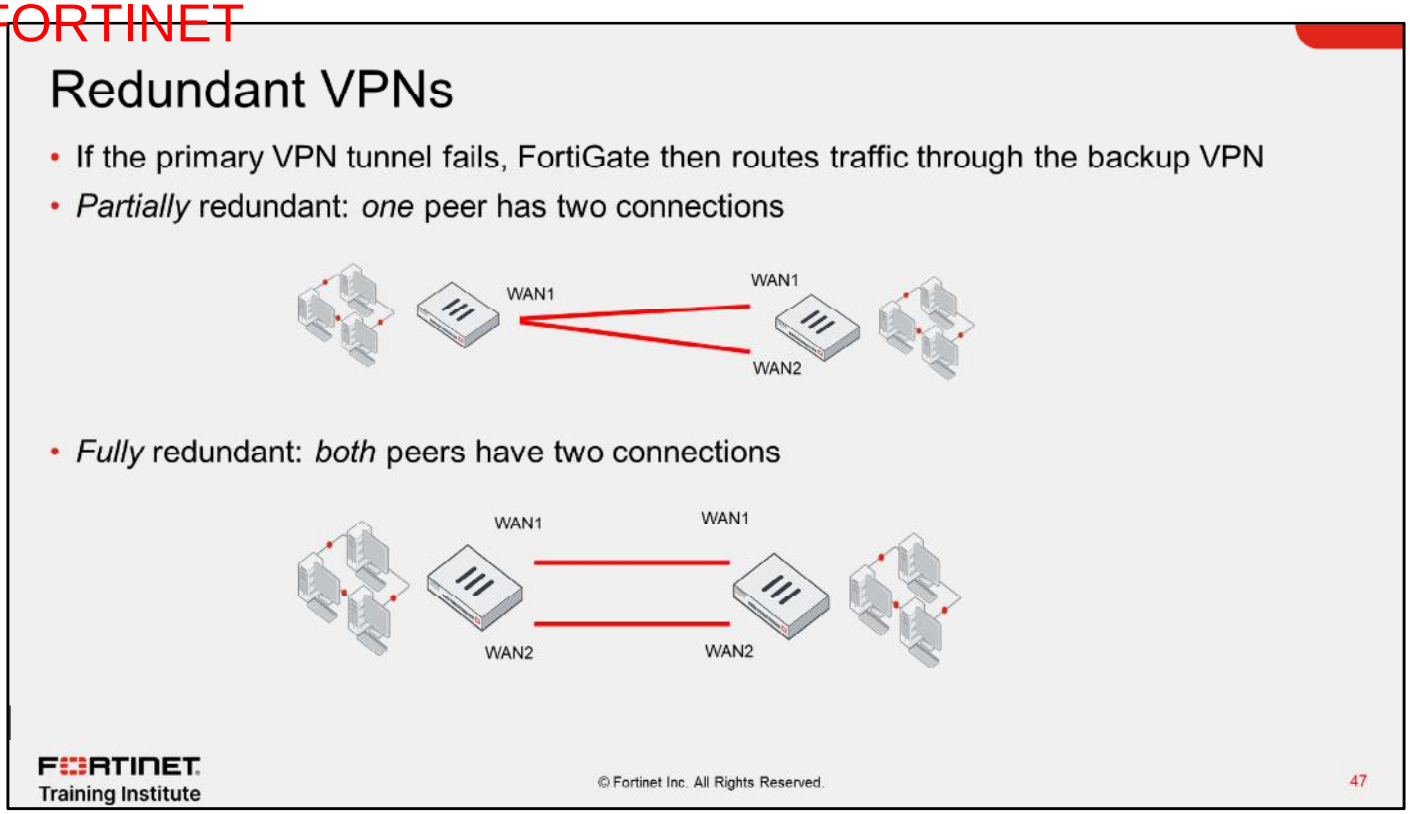
Good job! You now understand routing and firewall policies for IPsec traffic.

Now, you will learn about redundant VPNs.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in redundant VPNs, you will able to add redundancy to your IPsec VPN deployment.



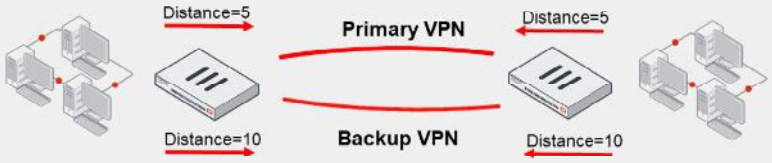
How can you make your IPsec VPN deployment more resilient? Provide a second ISP connection to your site and configure two IPsec VPNs. If the primary IPsec VPN fails, another tunnel can be used instead.

There are two types of redundant VPNs:

- Partially redundant: on one peer (usually the hub, where a backup ISP is available if the main ISP is down), each VPN terminates on *different* physical ports. That way, FortiGate can use an alternative VPN. On the other peer, each VPN terminates on the *same* physical port—so the spoke is not fault tolerant.
- Fully-redundant: both peers terminate their VPNs on different physical ports, so they are both fault tolerant.

Redundant VPN Configuration

- Add one phase 1 configuration for each tunnel. You should enable DPD on both ends.
- Add at least one phase 2 definition for each phase 1
- Add one static route for each path
  - Use distance or priority to select primary routes over backup routes
  - Alternatively, use dynamic routing
- Configure firewall policies for each IPsec interface



FORTINET Training Institute

48

So, how do you configure a partially or fully redundant VPN?

First, create one phase 1 for each path—one phase 1 for the primary VPN and one for the backup VPN. You should also enable DPD on both ends.

Second, create at least one phase 2 definition for each phase 1.

Third, you must add at least one static route for each VPN. Routes for the primary VPN must have a lower distance (or lower priority) than the backup. This causes FortiGate to use the primary VPN while it's available. If the primary VPN fails, then FortiGate automatically uses the backup route. Alternatively, you could use a dynamic routing protocol, such as OSPF or BGP.

Finally, configure firewall policies to allow traffic through both the primary and backup VPNs.

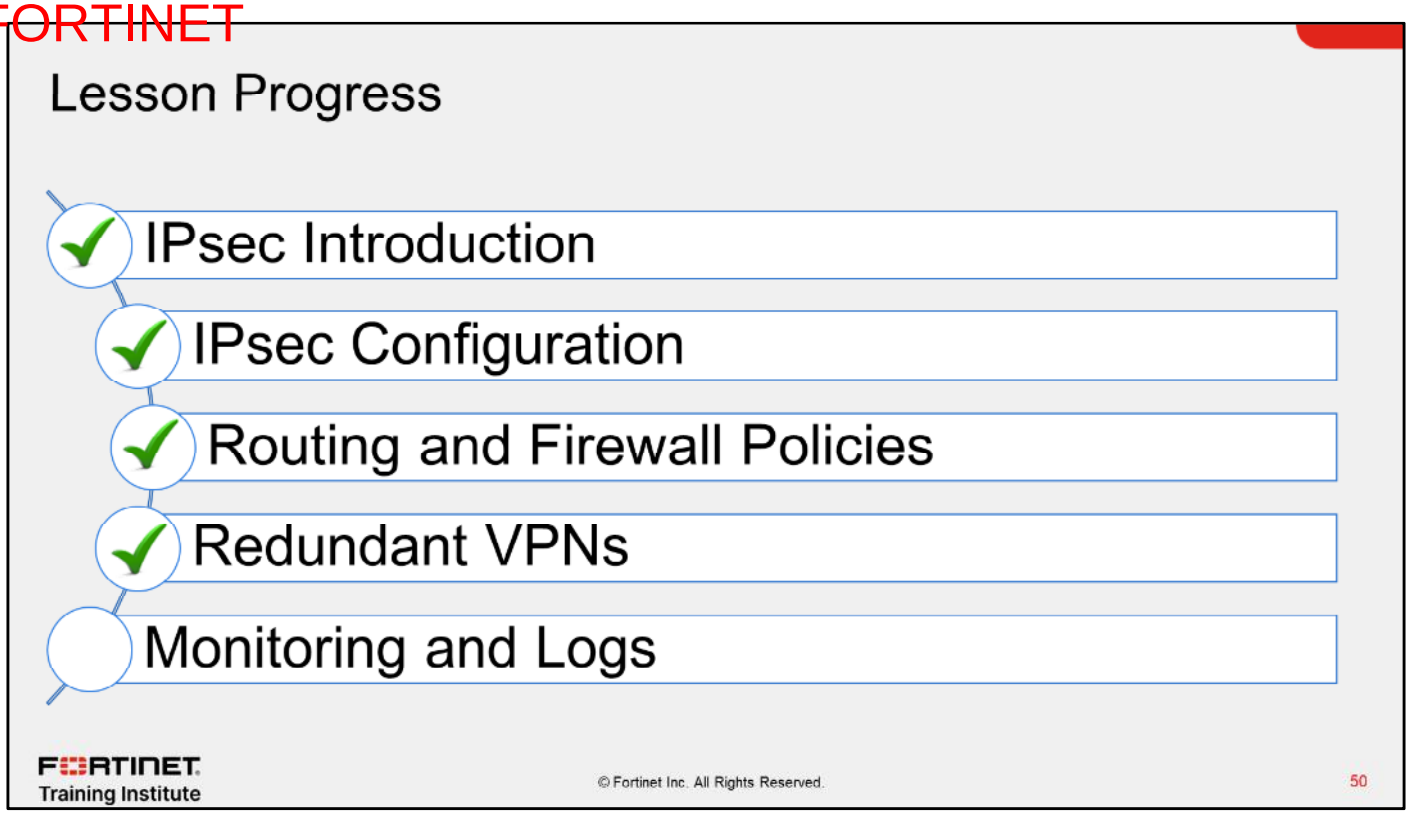
Knowledge Check

1. Which feature should be enabled in a redundant IPsec VPN deployment?
   - ✓ A. DPD
   - B. XAuth

2. Which setting determines whether a tunnel is used as primary or backup?
   - ✓ A. Routing
   - B. Firewall policies

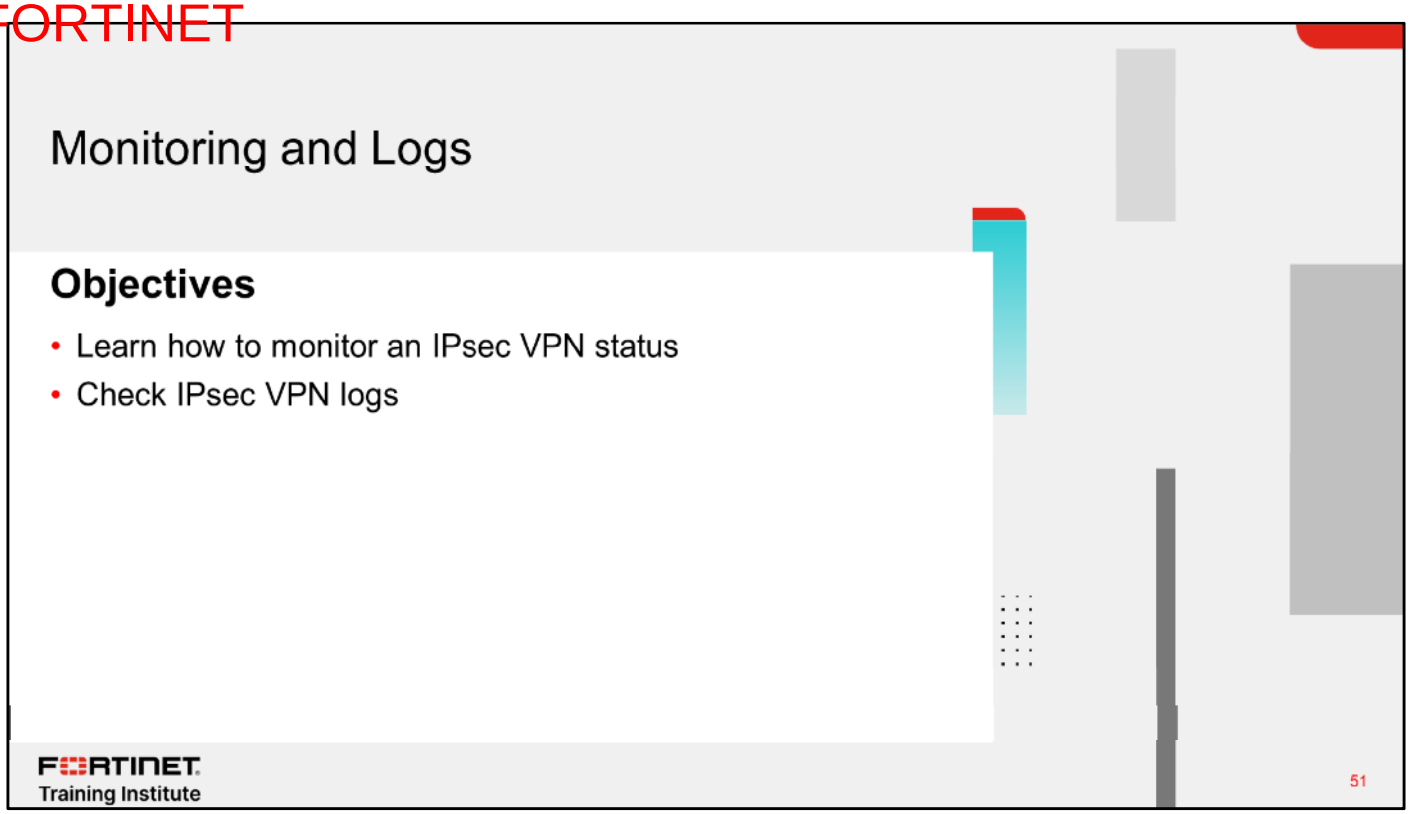
FORTINET Training Institute

49



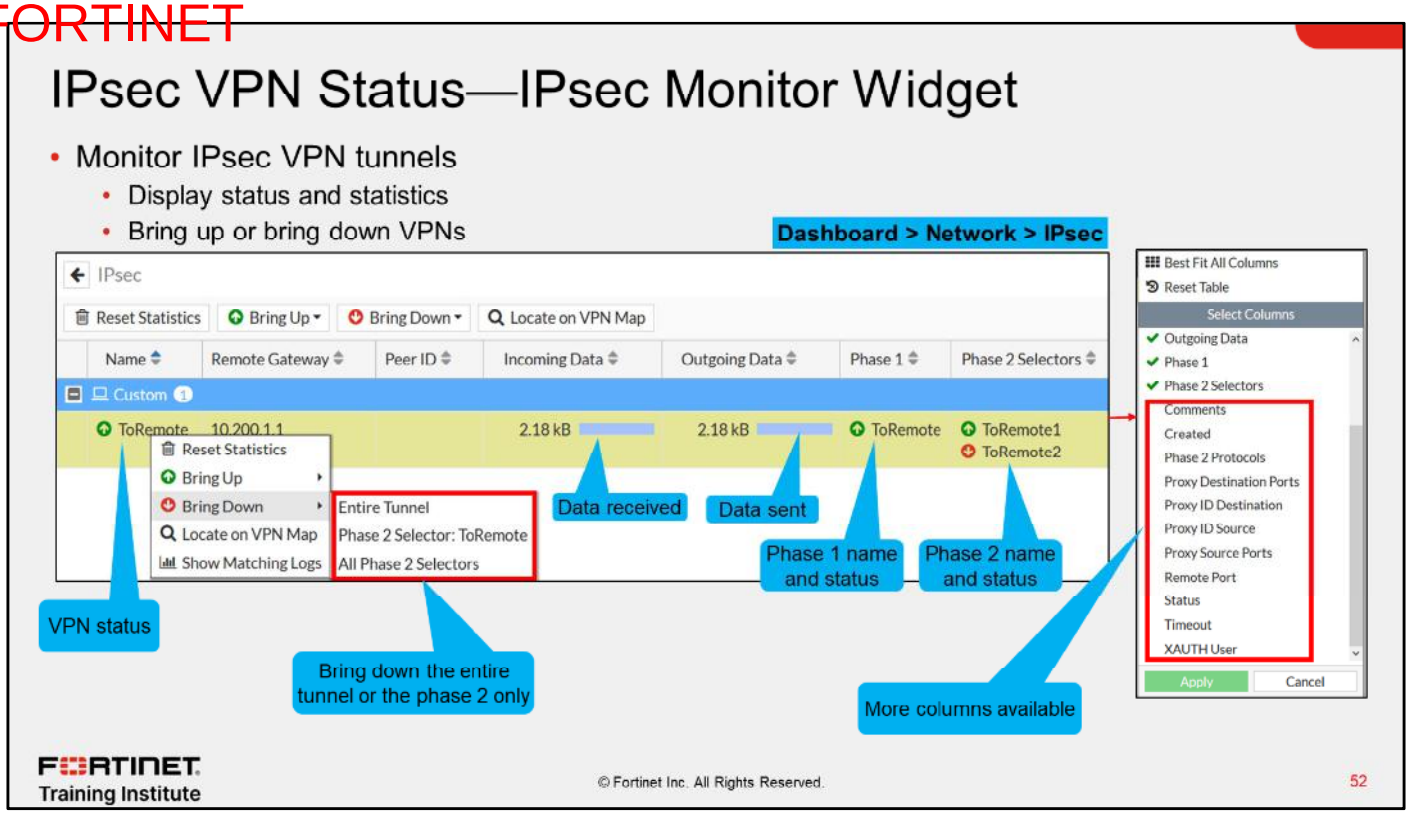
Good job! You now understand redundant VPNs.

Now, you will learn about monitoring IPsec VPNs and reviewing their logs.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in monitoring and logs, you will be able to monitor IPsec VPN and review past events.



On the GUI dashboard, you can use the IPsec widget to monitor the status of your IPsec VPNs. The widget shows the phase 1 and phase 2 status of an IPsec VPN.

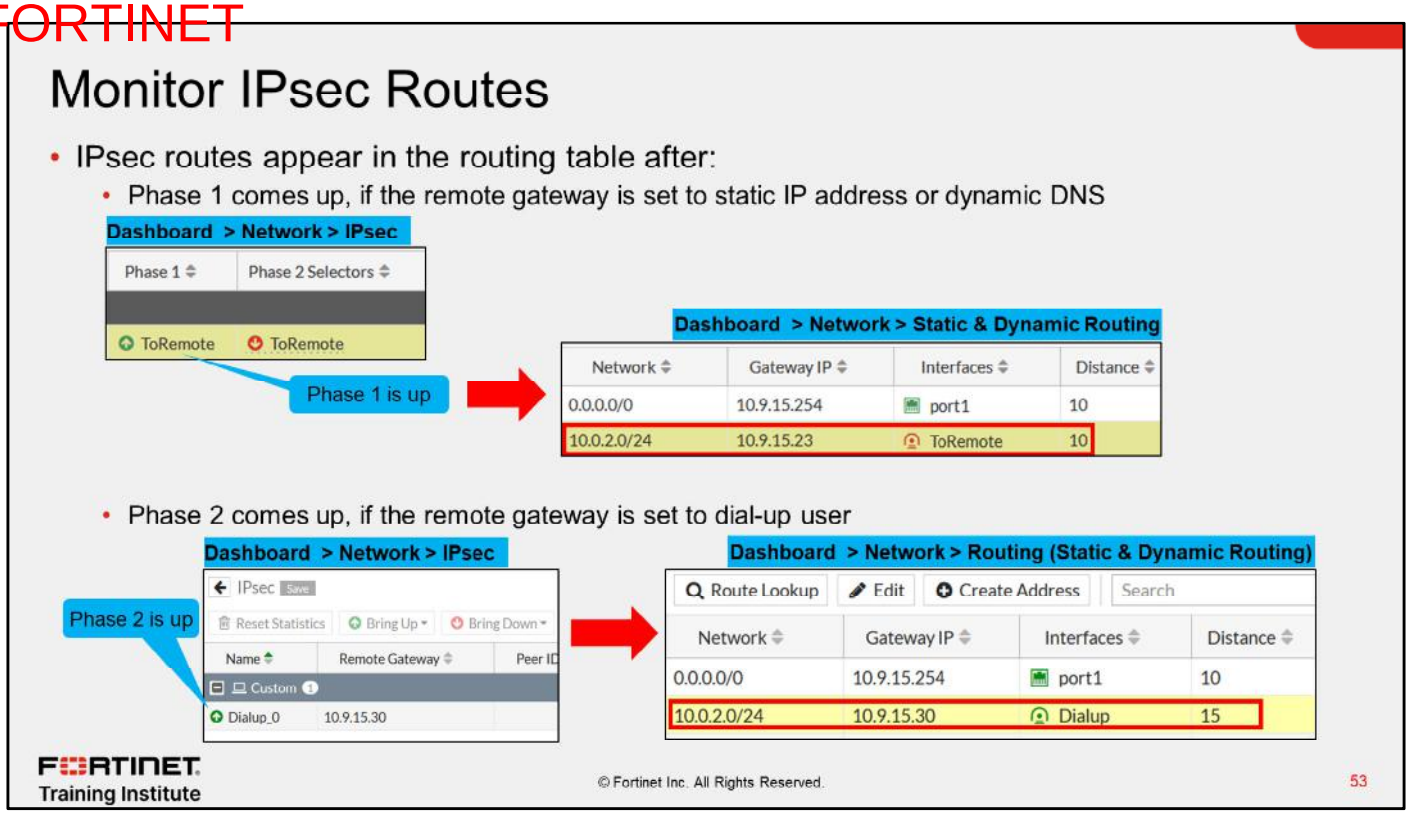
You can also bring up or bring down individual VPNs, and get additional details. When you bring up an IPsec VPN using the IPsec widget, you can choose between bringing up a particular phase 2 selector or all phase 2 selectors in that VPN. Because bringing up a phase 2 selector requires bringing up its phase 1 first, then bringing up a phase 2 selector results in its phase 1 also coming up.

To bring down the VPN, you can choose between bringing down a particular phase 2 selector, all selectors, or the entire tunnel. When you bring down the entire tunnel, you bring down all phase 2 selectors as well as the phase 1.

The **Name** column indicates the VPN status. The VPN is up when at least one of its phase 2 selectors is up. If all phase 2 selectors are down, the VPN status is also down. The **Phase 1** and **Phase 2 Selectors** columns indicate the status of phase 1 and phase 2 selectors, respectively.

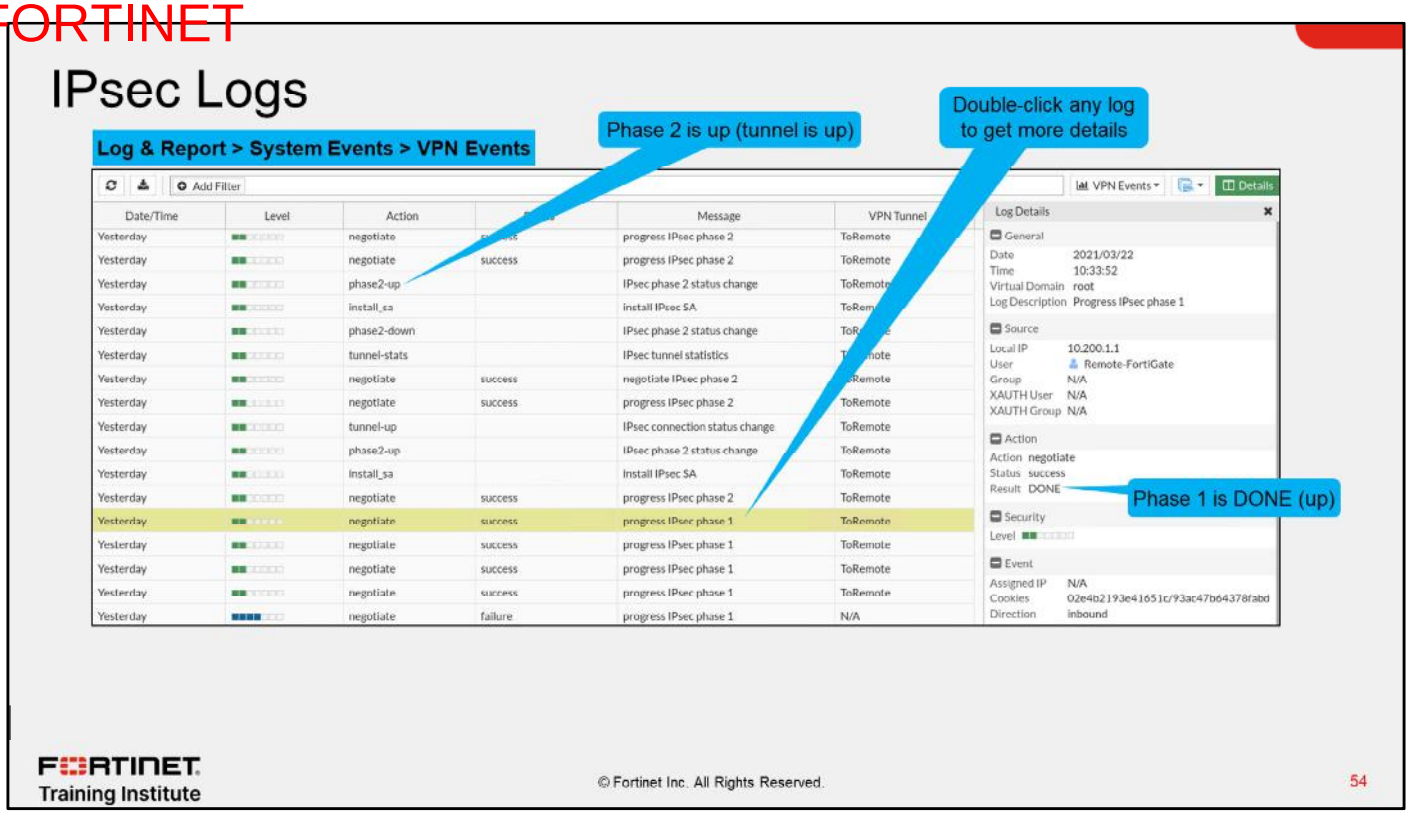
The IPsec widget also displays the amount of data sent and received through the tunnel. When you right-click any of the columns, a menu opens with a list of all the columns available. You can enable additional columns to get further details about the IPsec tunnels.

In the example shown on this slide, the **ToRemote** VPN is up because at least one of its phase 2 selectors (**ToRemote1**) is up.



If you set the remote gateway to **Static IP Address** or **Dynamic DNS**, the static routes for these tunnels become active in the routing table after phase 1 comes up. Phase 1 negotiation is started automatically because automatic negotiation is enabled on phase 1 by default. This behavior allows FortiGate to match interesting traffic to the right tunnel. Moreover, if phase 2 is not up, traffic matching the static route triggers a phase 2 negotiation, which eventually results in the tunnel (or phase 2) to come up.

When you set the remote gateway to **Dialup User**, by default, a static route for the destination network is added after phase 2 comes up. The distance set for the static route is 15. If phase 2 goes down, the route is removed from the routing table.



FortiGate logs IPsec VPN events by default. To view IPsec VPN event logs, click **Log & Report** > **System Events** > **VPN Events**.

The logs track the progress of phase 1 and phase 2 negotiations, and report on tunnel up and down events and DPD failures, among other events. For more information about IPsec logs, visit `https://docs.fortinet.com`.

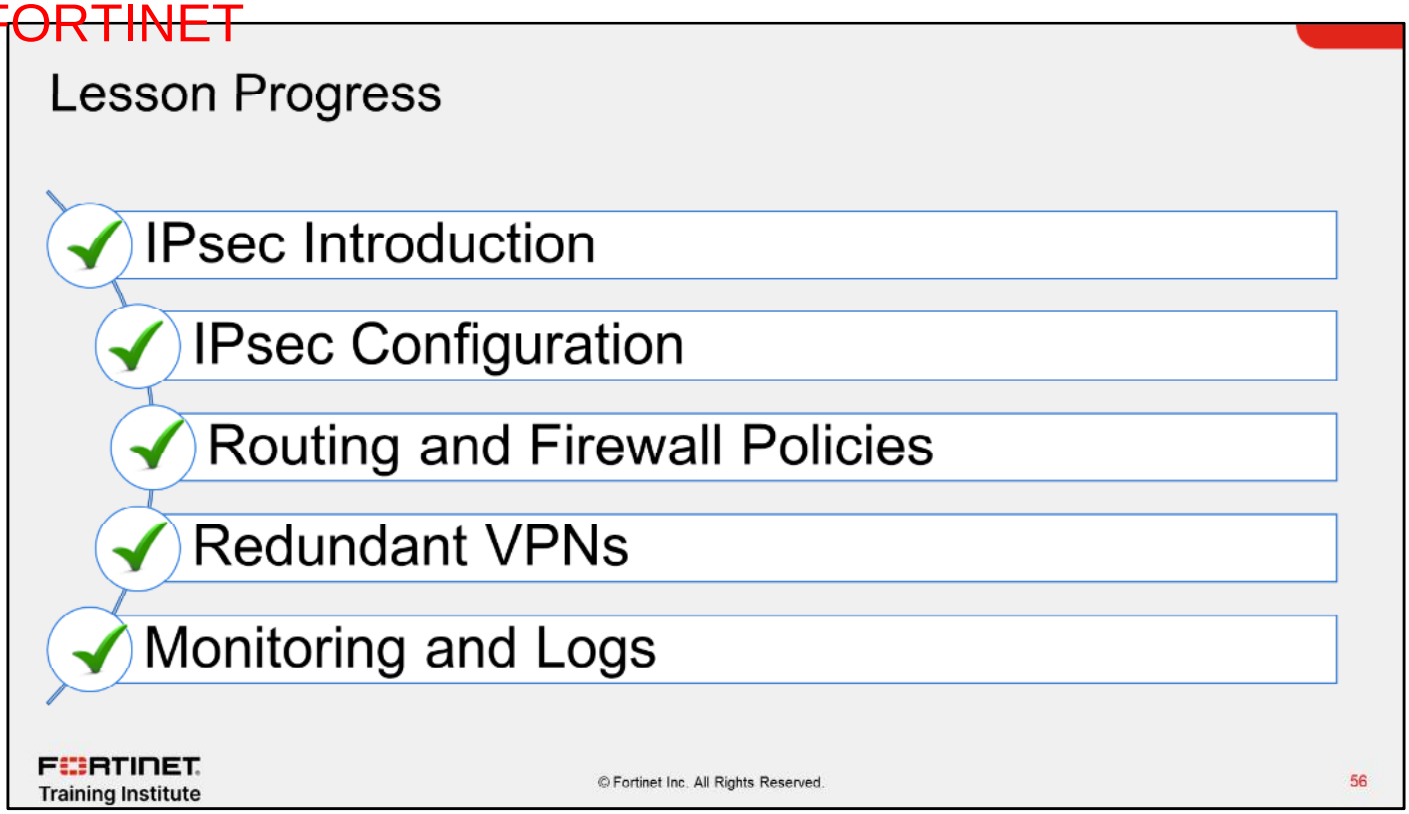
Knowledge Check

1. The IPsec monitor widget enables you to bring down the _____ of an IPsec VPN.
   - A. Phase 1
   - ✓ B. Entire tunnel

2. When the remote gateway is set to dial-up user, a static route to the remote network is added to the routing table after _____.
   - A. Phase 1 comes up
   - ✓ B. Phase 2 comes up

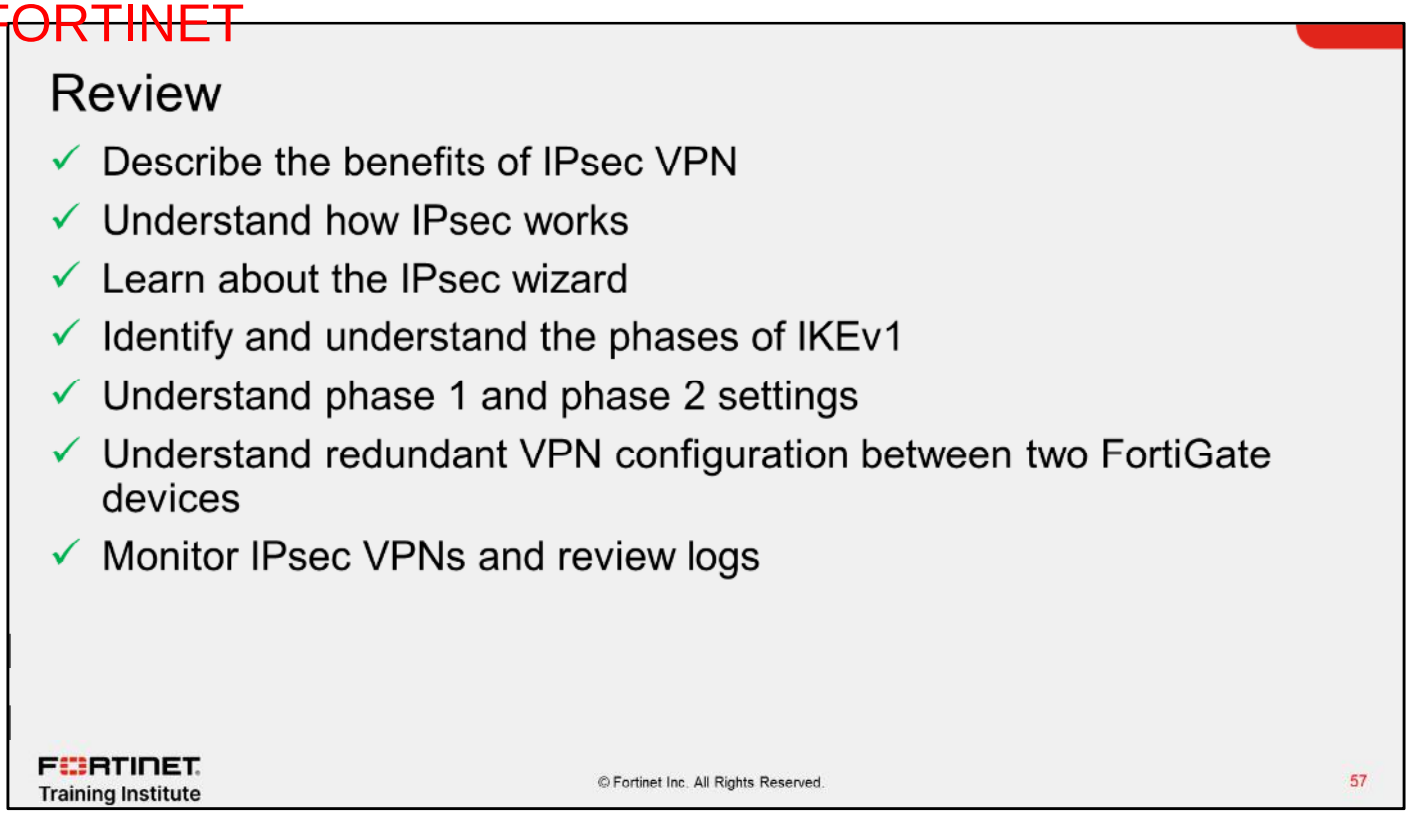
FORTINET Training Institute

55



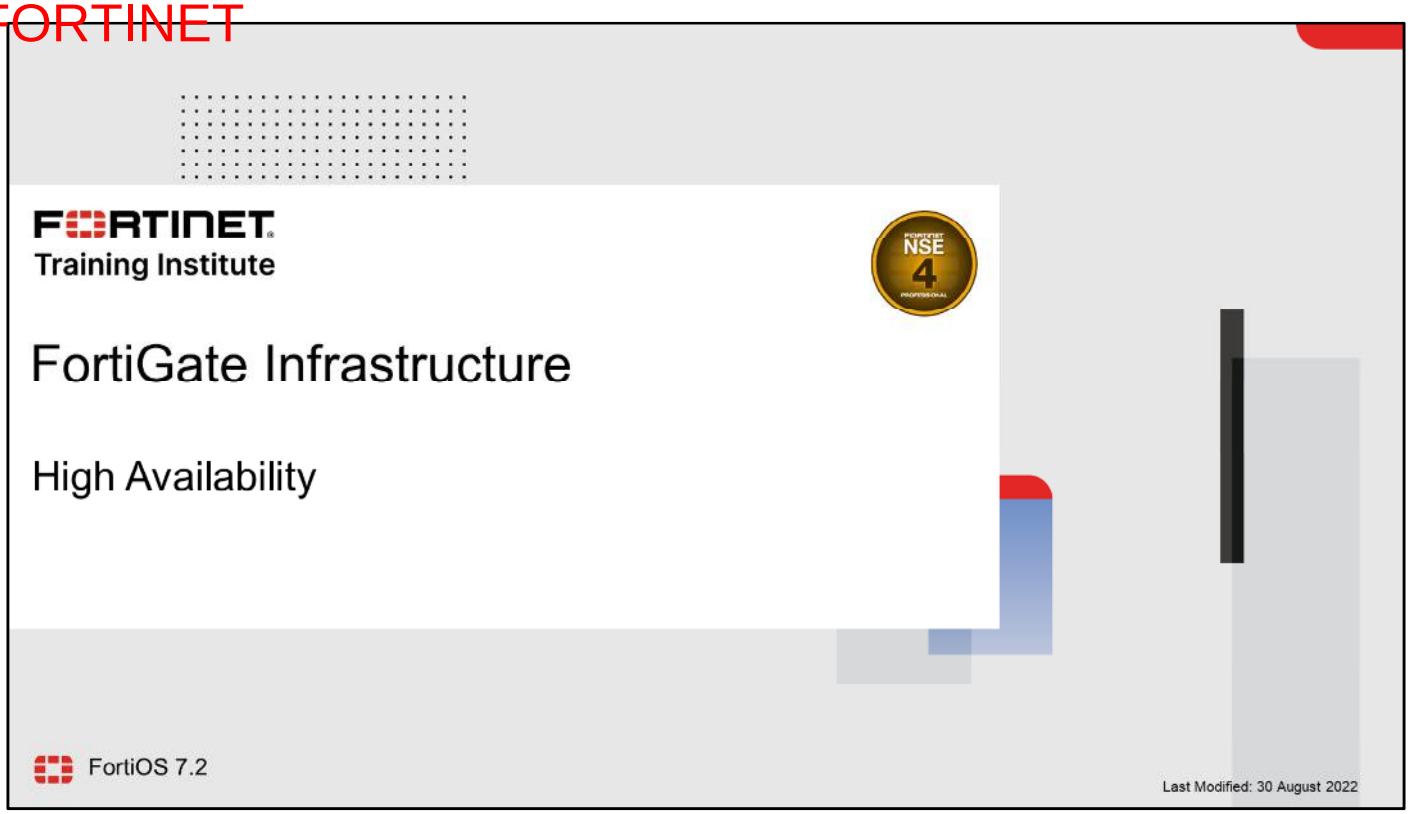
Congratulations! You have completed this lesson.

Now, you will review the objectives that you covered in this lesson.

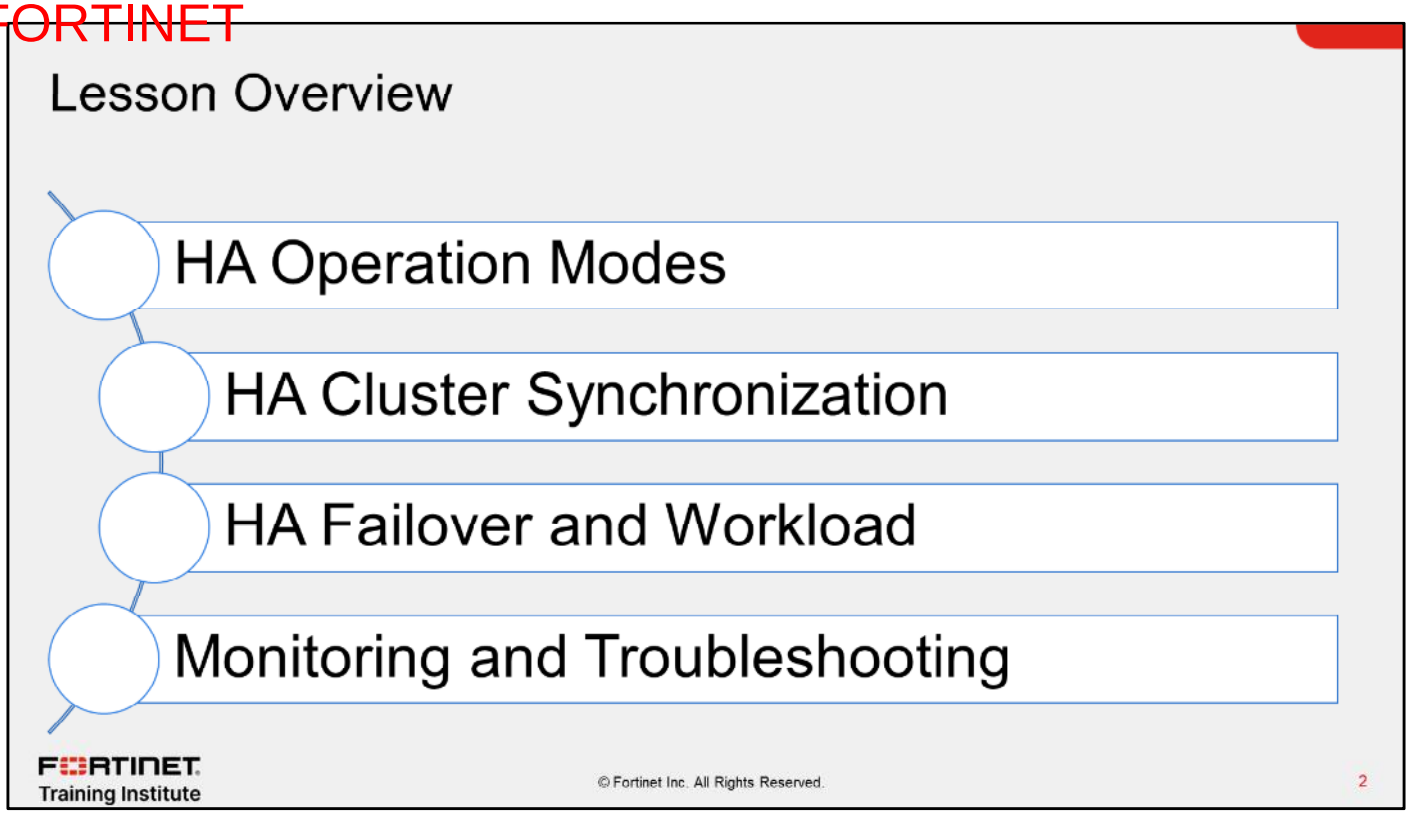


This slide shows the objectives that you covered in this lesson.

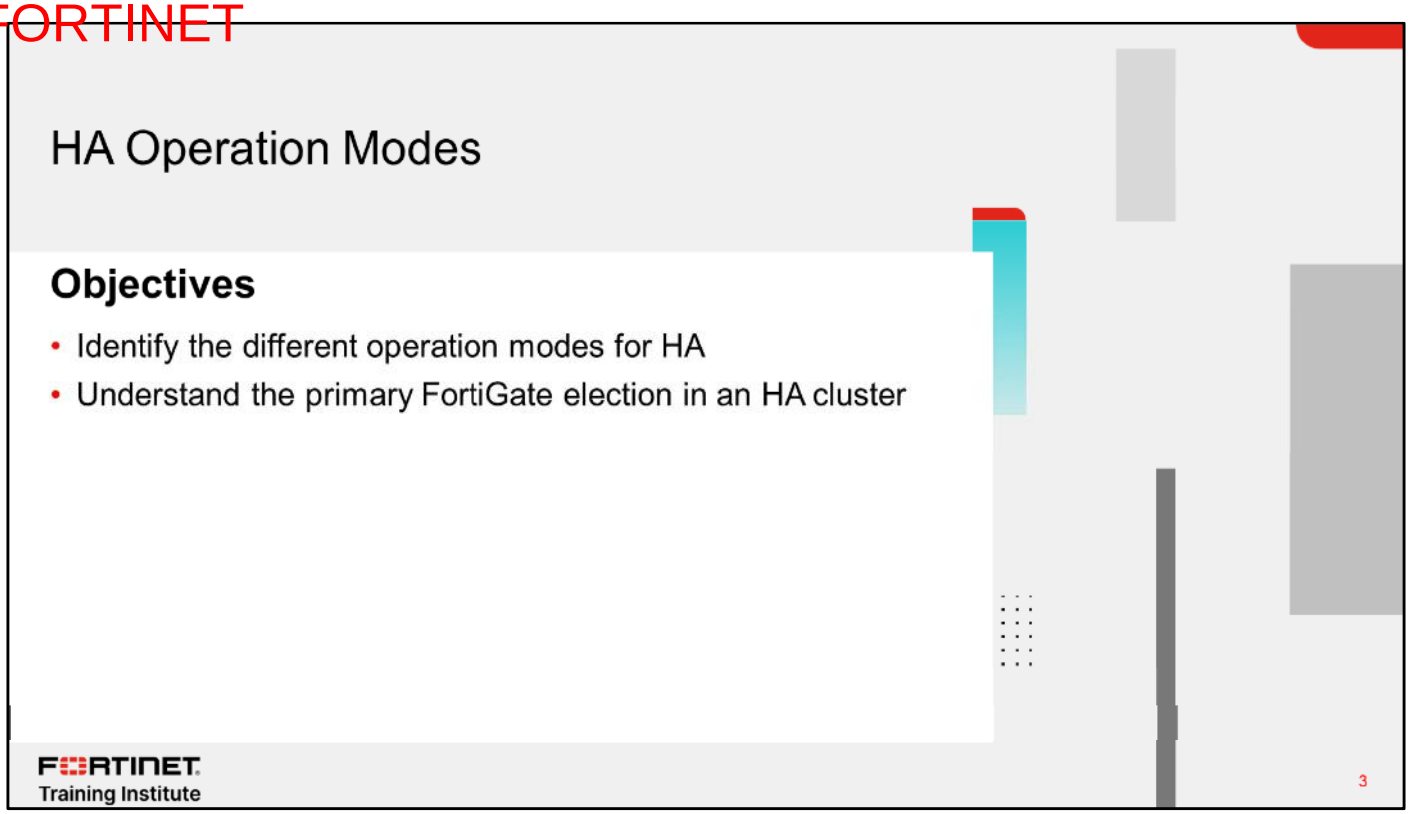
By mastering the objectives covered in this lesson, you learned how the IPsec protocol works, and how to configure and monitor IPsec VPNs on FortiGate.



In this lesson, you will learn about the fundamentals of FortiGate high availability (HA) and how to configure it. FortiGate HA provides a solution for enhanced reliability and increased performance.

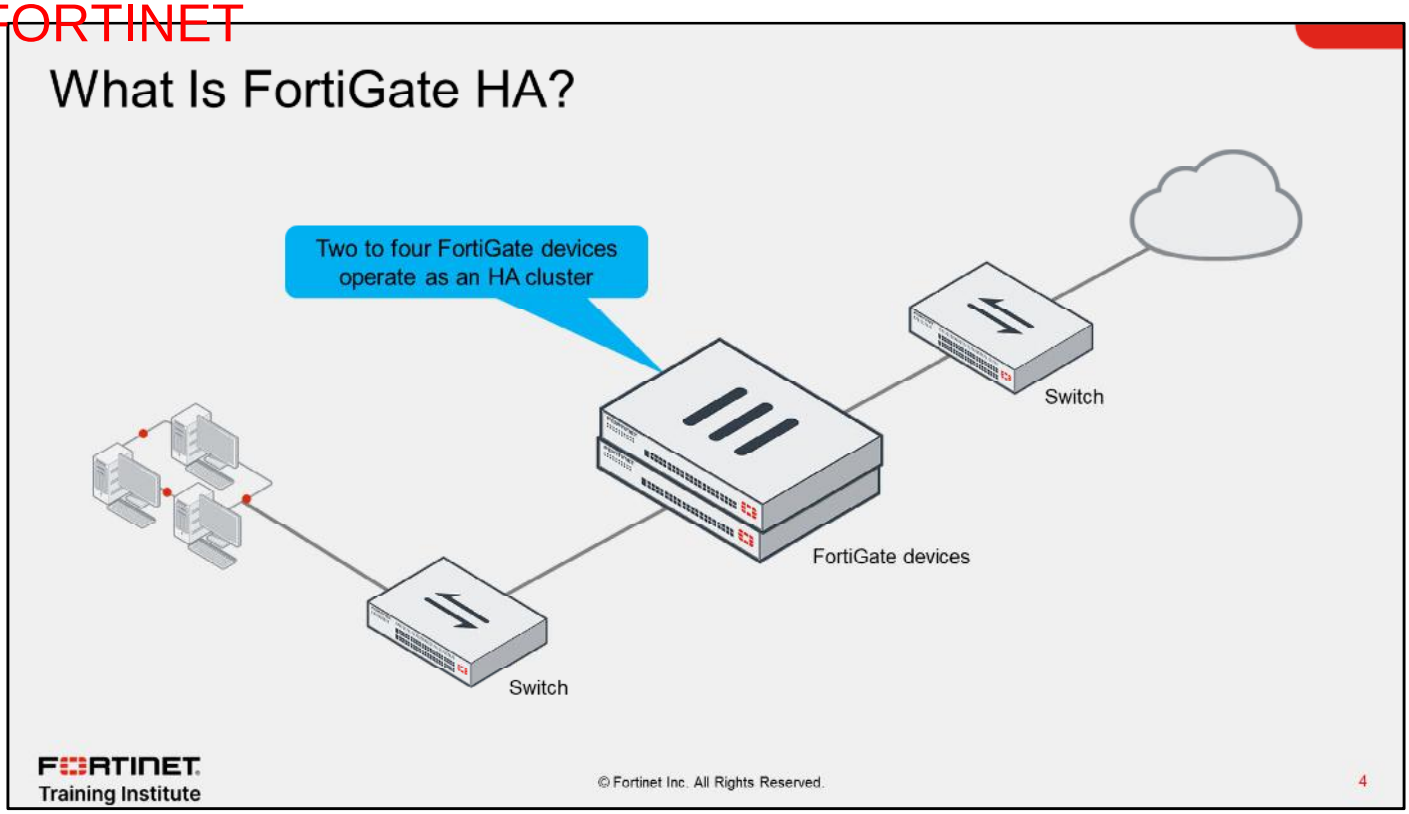


In this lesson, you will learn about the topics shown on this slide.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in HA operation modes and primary FortiGate election, you will be able to choose and implement the right HA operation mode in your network based on your requirements. You will be able to use FortiGate devices effectively in your network.

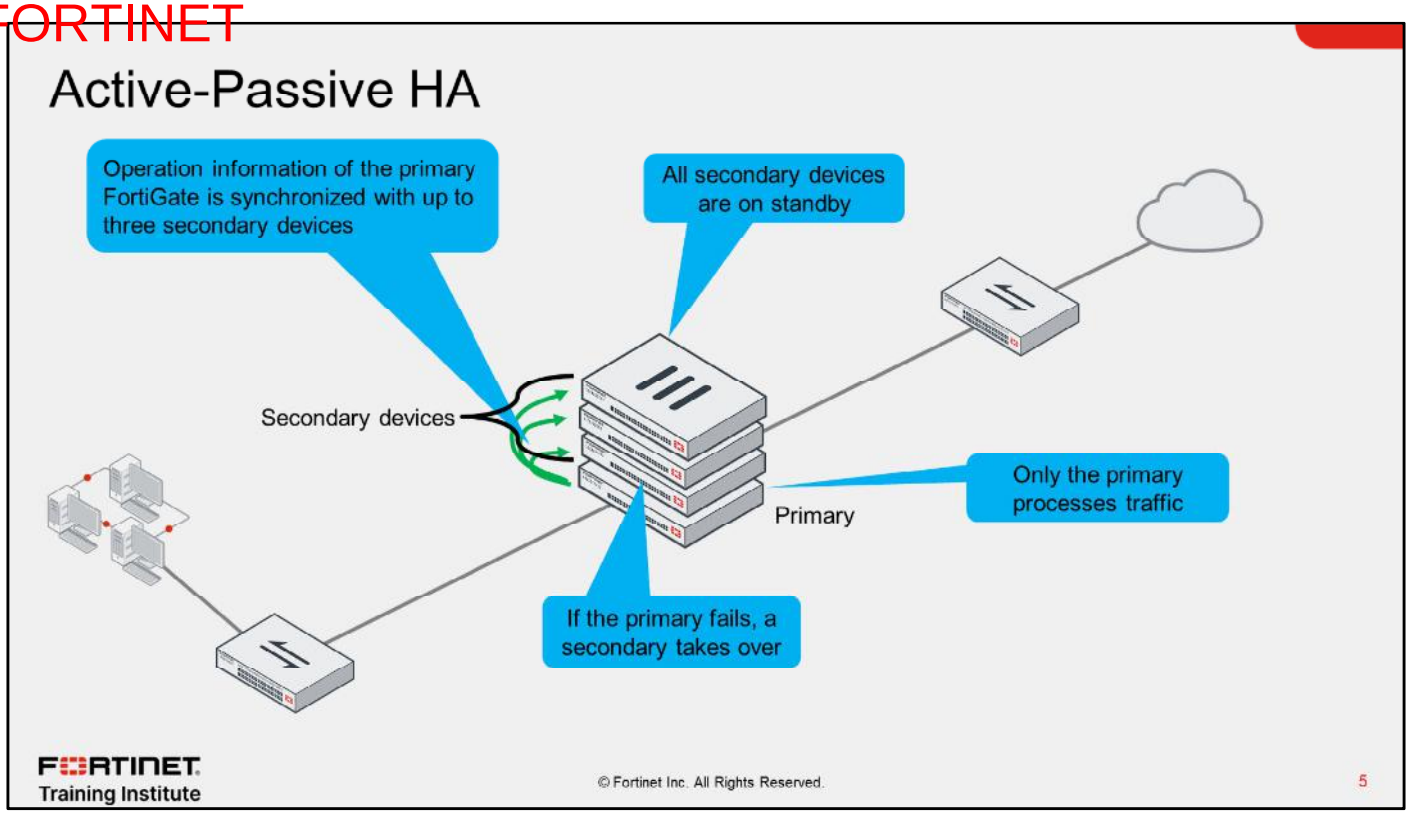


The idea of HA is simple. HA links and synchronizes two to four FortiGate devices to form a cluster for redundancy and performance purposes.

A cluster includes one device that acts as the primary FortiGate (also called the active FortiGate). The primary synchronizes its configuration, session information, FIB entries, FortiGuard definitions, and other operation-related information to the secondary devices, which are also known as standby devices.

The cluster shares one or more heartbeat interfaces among all devices—also known as members—for synchronizing data and monitoring the health of each member.

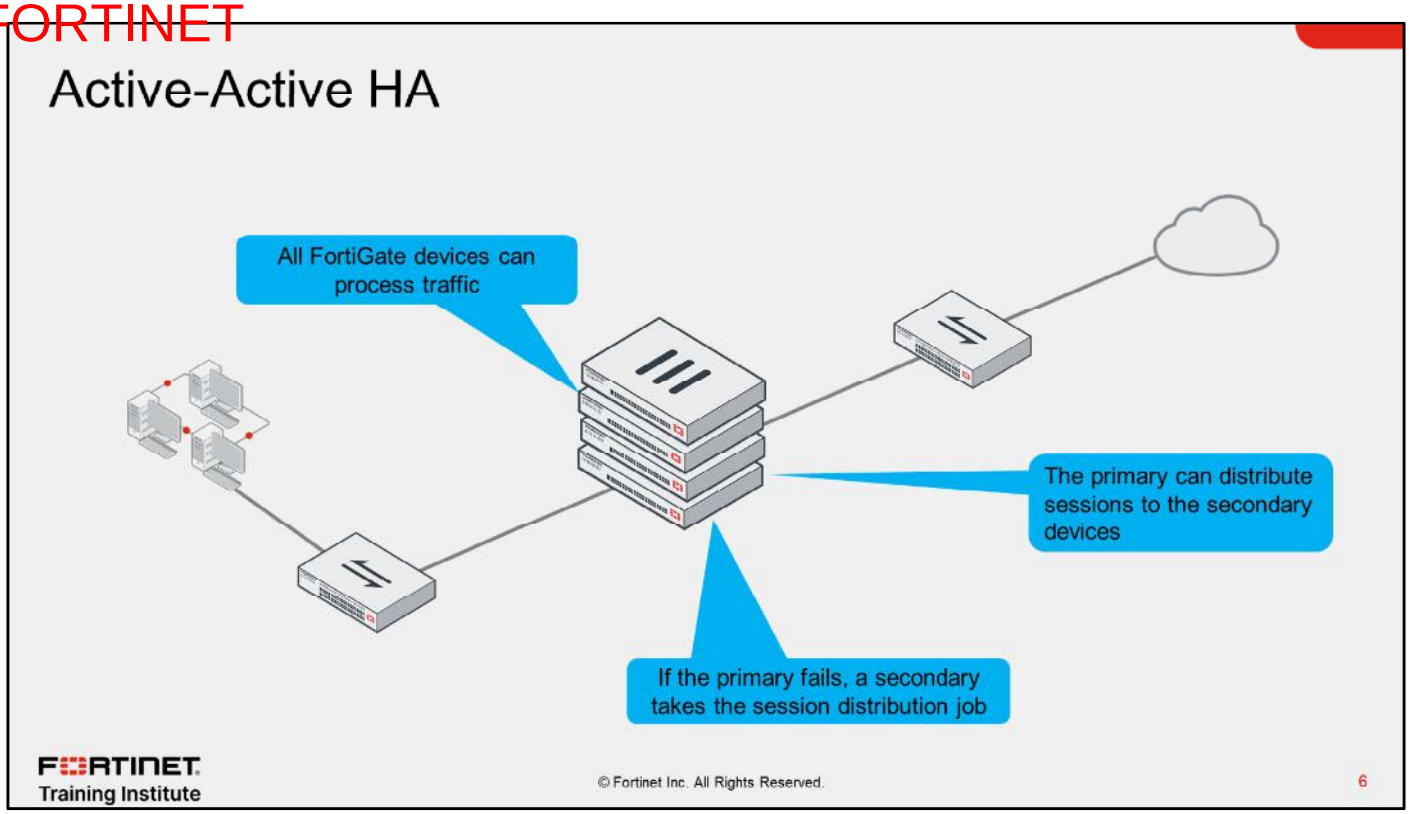
There are currently two HA operation modes available: active-active (A-A) and active-passive (A-P). Now, you will examine the differences.



First, take a look at active-passive mode. In either of the two HA operation modes, the operation information (configuration, sessions, FIB entries, and so on) of the primary FortiGate is synchronized with secondary devices.

In active-passive mode, the primary FortiGate is the only FortiGate that actively processes traffic. Secondary FortiGate devices remain in passive mode, monitoring the status of the primary device.

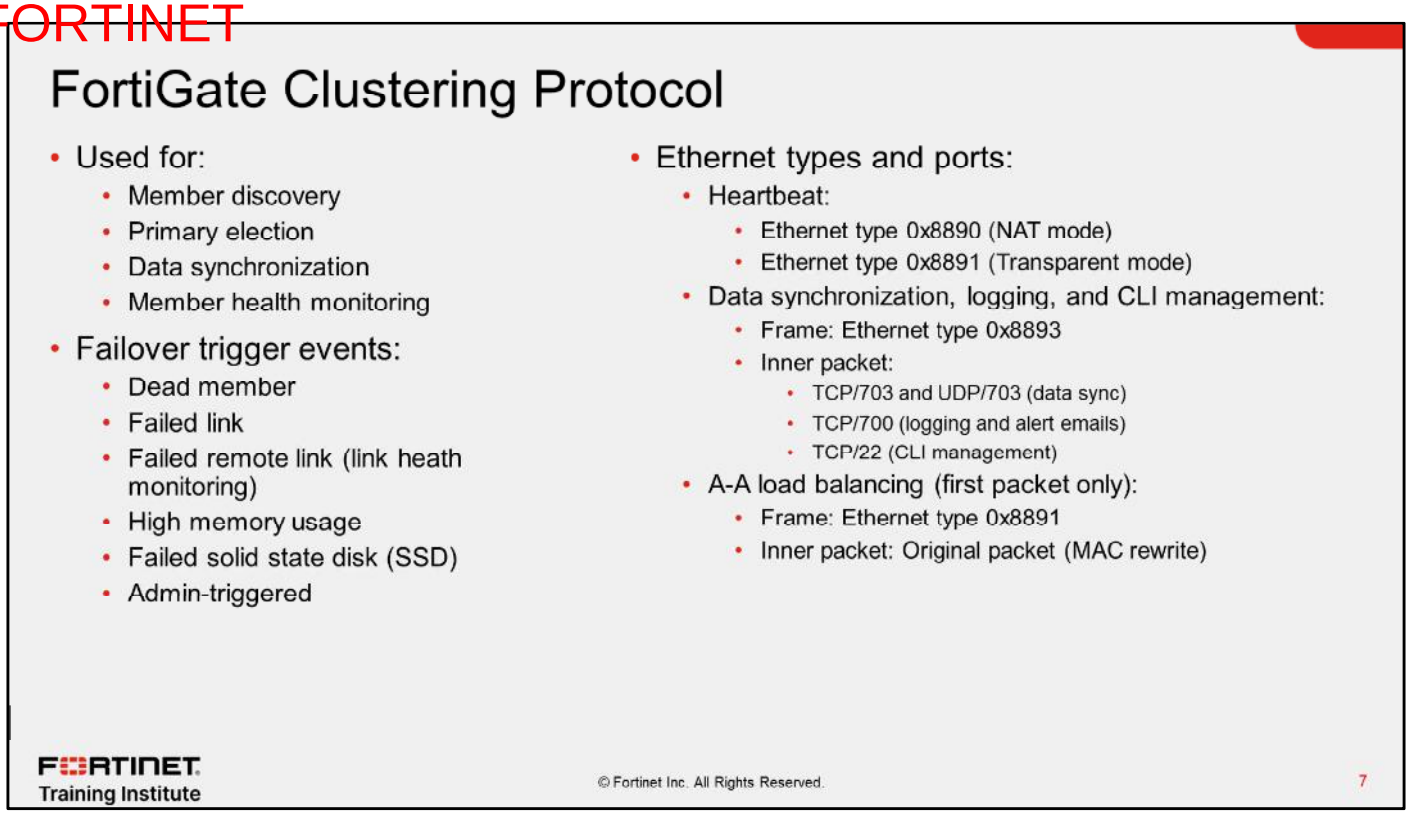
If a problem is detected on the primary FortiGate, one of the secondary devices takes over the primary role. This event is called an *HA failover*.



The other HA mode is active-active.

Like active-passive HA, in active-active HA, the operation-related data of the primary FortiGate is synchronized to the secondary FortiGate devices. Also, if a problem is detected on the primary device, one of the secondary devices takes over the role of the primary, to process the traffic.

However, one of the main differences from active-passive mode is that in active-active mode, all cluster members can process traffic. That is, based on the HA settings and traffic type, the primary FortiGate can distribute sessions to the secondary devices.



FortiGate HA uses the Fortinet-proprietary FortiGate Clustering Protocol (FGCP) to discover members, elect the primary FortiGate, synchronize data among members, and monitor the health of members.

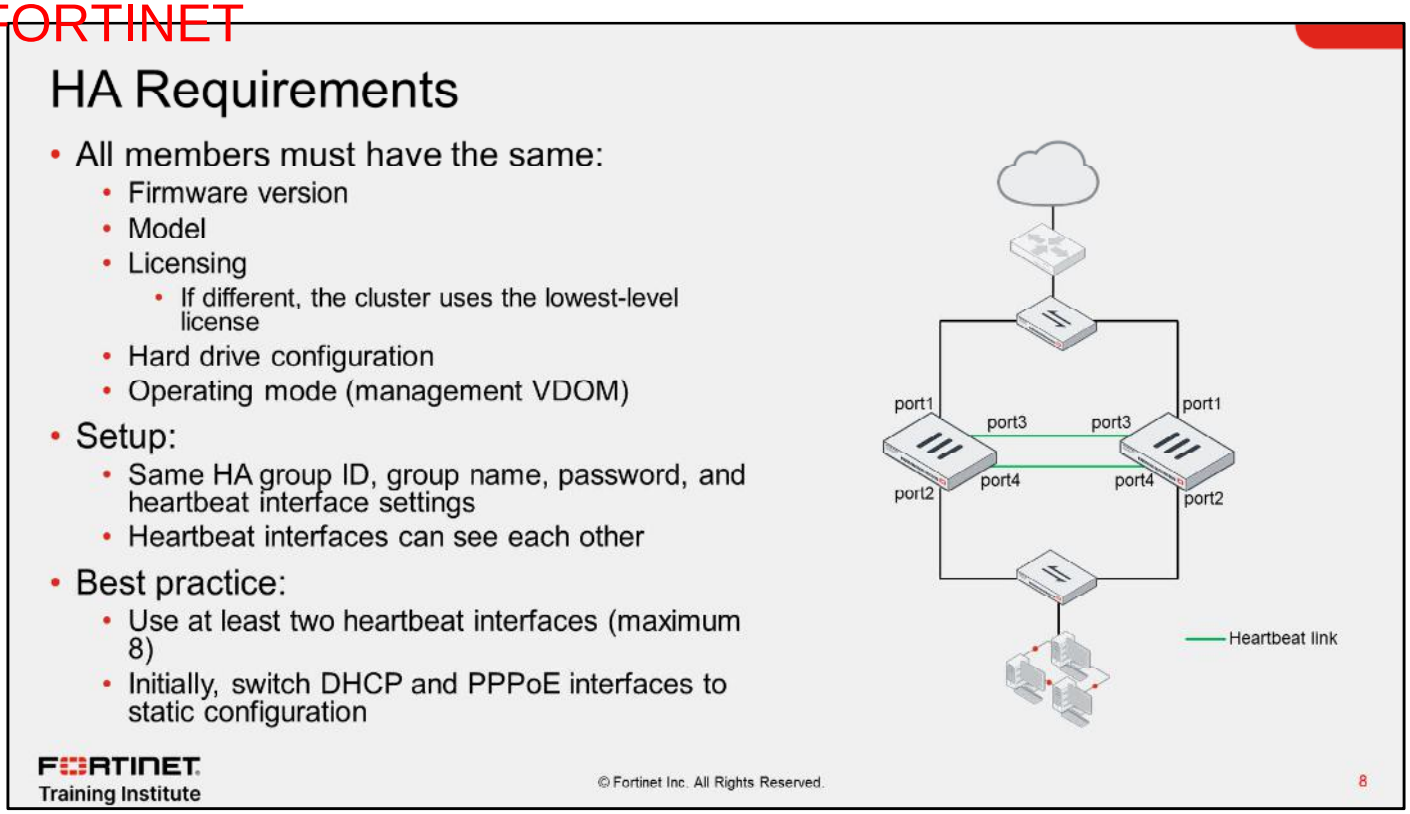
To discover and monitor members, the members broadcast heartbeat packets over all configured heartbeat interfaces. If the cluster operates in NAT mode, the heartbeat frames are type 8890. In transparent mode, the heartbeat frames are type 8891. If the cluster operates in active-active mode, the first packet of a session distributed to the secondary is encapsulated in Ethernet frames type 8891.

The members also exchange frames type 8893 for data synchronization, local CLI management, and logging purposes. For data synchronization, the inner packet can be TCP port 703 or UDP port 703, depending on the type of data to synchronize. The primary also relays logs and alert emails from secondary devices over TCP port 700. For local HA management using the CLI, the inner packet is SSH.

You can configure the cluster to perform HA failover based on the following events:

- Dead member: The primary FortiGate is unresponsive.
- Failed link: The link of one or more monitored interfaces on the primary FortiGate goes down.
- Failed remote link: FortiGate uses the link health monitor feature to monitor the health of one or more interfaces. The primary fails if the accumulated penalty of all failed interfaces reaches the set threshold.
- High memory usage: The primary fails if its memory utilization reaches the configured threshold.
- Failed SSD: FortiOS detects a failure in an SSD. Only available for devices with SSDs.
- Admin-triggered: The administrator issues a manual failover.

For any of the failover trigger events, the result is that the cluster promotes one of the secondary devices to the new primary FortiGate role.



To successfully form an HA cluster, you must ensure that the members have the same:

- Firmware version
- Model: the same hardware model or VM model
- Licensing: includes the FortiGuard license, VDOM license, FortiClient license, and so on
- Hard drive configuration: the same number and size of drives and partitions
- Operating mode: the operating mode—NAT mode or transparent mode—of the management VDOM

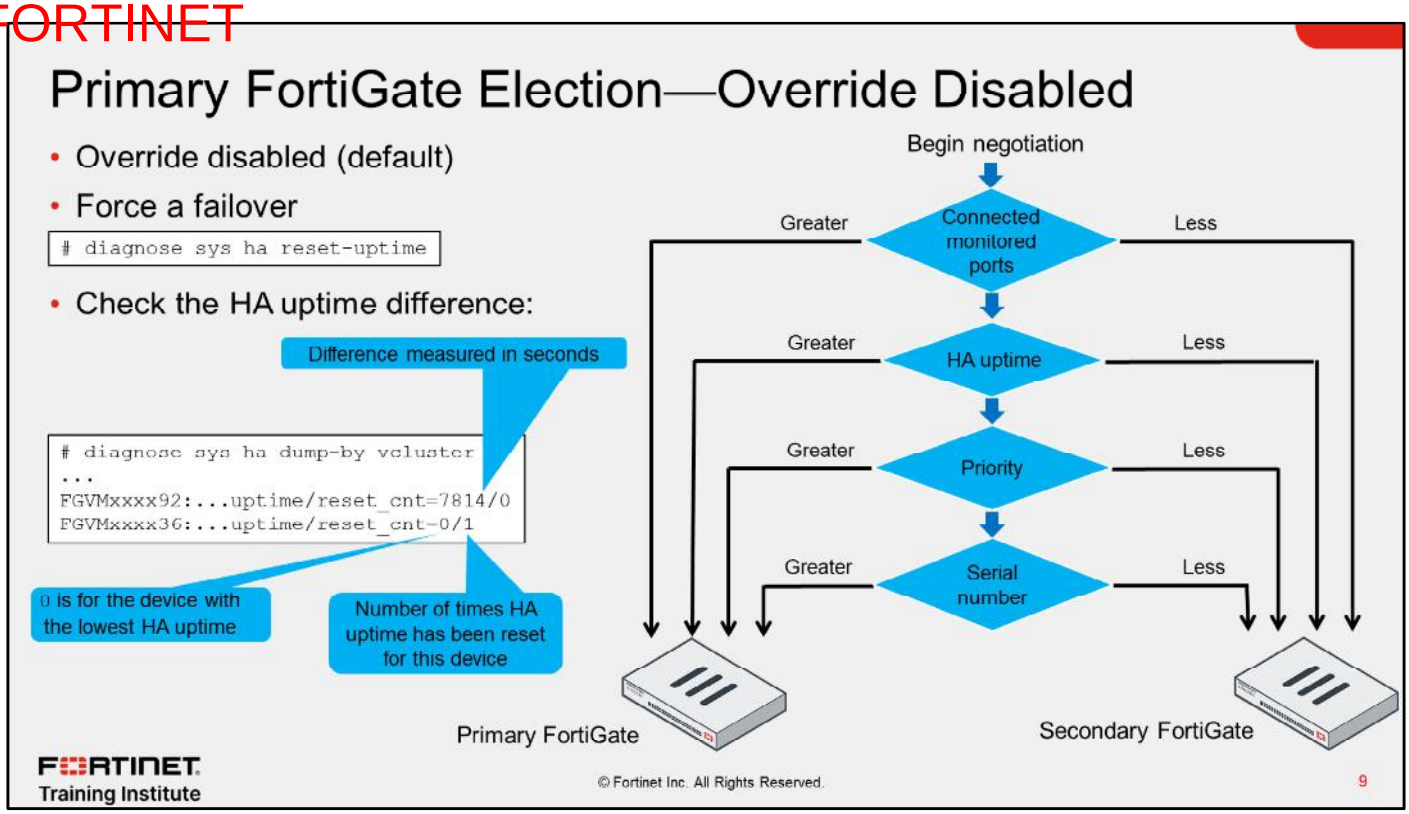
If the licensing level among members isn't the same, the cluster resolves to use the lowest licensing level among all members. For example, if you purchase FortiGuard Web Filtering for only one of the members in a cluster, none of the members will support FortiGuard Web Filtering when they form the cluster.

From a configuration and setup point of view, you must also make sure that:

- The HA settings on each member have the same group ID, group name, password, and heartbeat interface settings.
- The heartbeat interfaces on each member can see each other. This usually means placing all heartbeat interfaces in the same broadcast domain, or for two-member clusters, connecting them directly.

It's also a best practice to:

- Configure at least two heartbeat interfaces for redundancy purposes. This way, if one heartbeat link fails, the cluster uses the next one, as indicated by the priority and position in the heartbeat interface list.
- If using DHCP or PPPoE interfaces, use static configuration during the cluster initial setup to prevent incorrect address assignment. After the cluster is formed, you can put back the original interface settings.



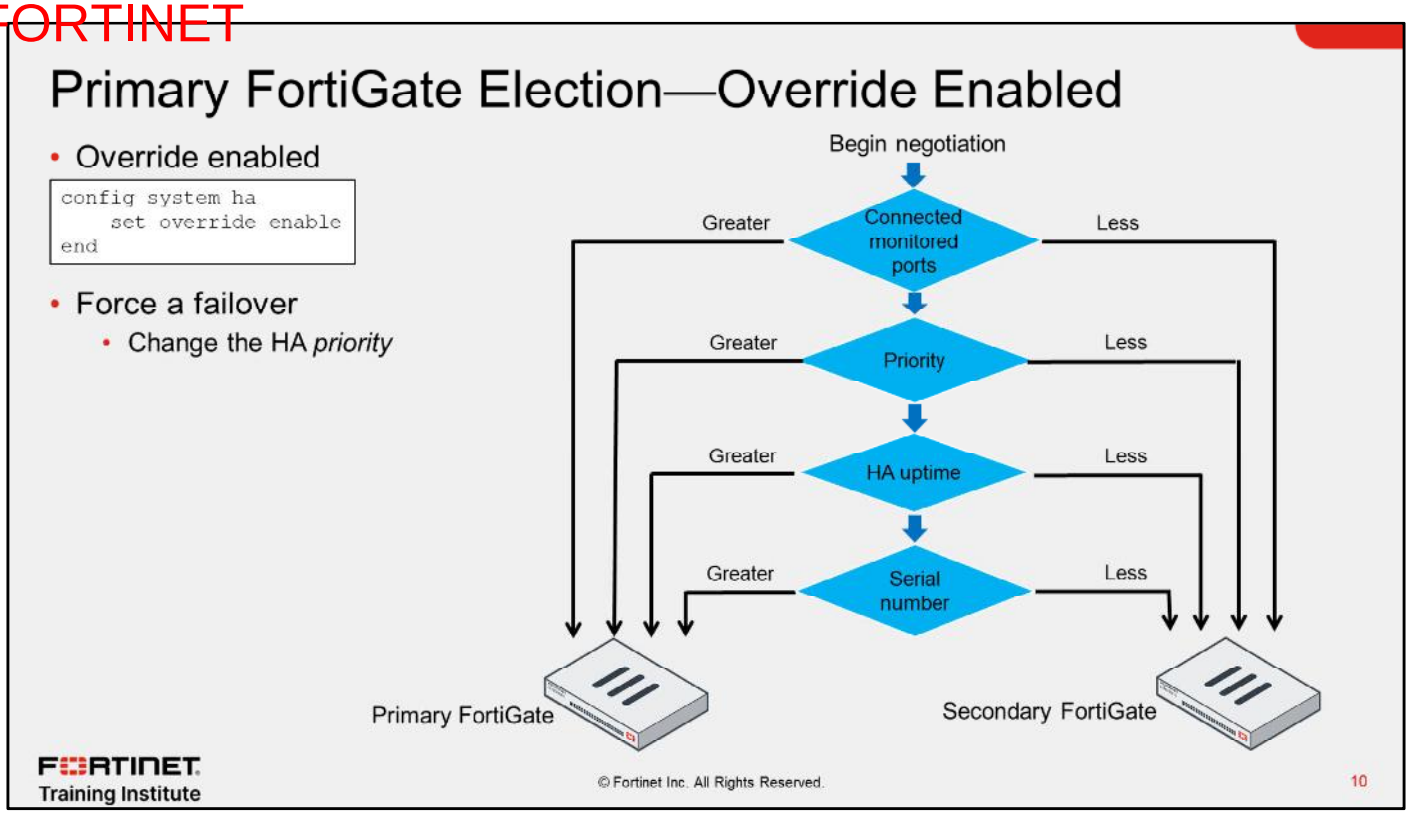
This slide shows the different criteria that a cluster considers during the primary FortiGate election process. The criteria order evaluation depends on the HA override setting. This slide shows the order when the HA override setting is disabled, which is the default behavior. Note that the election process stops at the first matching criteria that successfully selects a primary FortiGate in a cluster.

1. The cluster compares the number of monitored interfaces that have a status of up. The member with the most available monitored interfaces becomes the primary.
2. The cluster compares the HA uptime of each member. The member with the highest HA uptime, by at least five minutes, becomes the primary.
3. The member with the highest priority becomes the primary.
4. The member with the lowest serial number becomes the primary.

When HA override is disabled, the HA uptime has precedence over the priority setting. This means that if you must manually fail over to a secondary device, you can do so by reducing HA uptime of the primary FortiGate. You can do this by running the `diagnose sys ha reset-uptime` command on the primary FortiGate, which resets its HA uptime to 0.

Note that the `diagnose sys ha reset-uptime` command resets the HA uptime and not the system uptime. Also, note that if a monitoring interface fails, or a member reboots, the HA uptime for that member is reset to 0.

This slide also shows how to identify the HA uptime difference between members. The member with `0` in the `uptime` column indicates the device with the lowest uptime. The example shows that the device with the serial number ending in `92` has an HA uptime that is `7814` seconds higher than the other device in the HA cluster. The `reset_cnt` column indicates the number of times the HA uptime has been reset for that device.



If the HA override setting is enabled, the priority is considered before the HA uptime.

The advantage of this method is that you can specify which device is the preferred primary every time (as long as it is up and running) by configuring it with the highest HA priority value. The disadvantage is that a failover event is triggered not only when the primary fails, but also when the primary is available again. That is, when the primary becomes available again, it takes its primary role back from the secondary FortiGate that temporarily replaced it.

When override is enabled, the easiest way of triggering a failover is to change the HA priorities. For example, you can either increase the priority of one of the secondary devices, or decrease the priority of the primary.

The override setting and device priority values are not synchronized to cluster members. That is, on each member, you must manually enable override and adjust the priority.

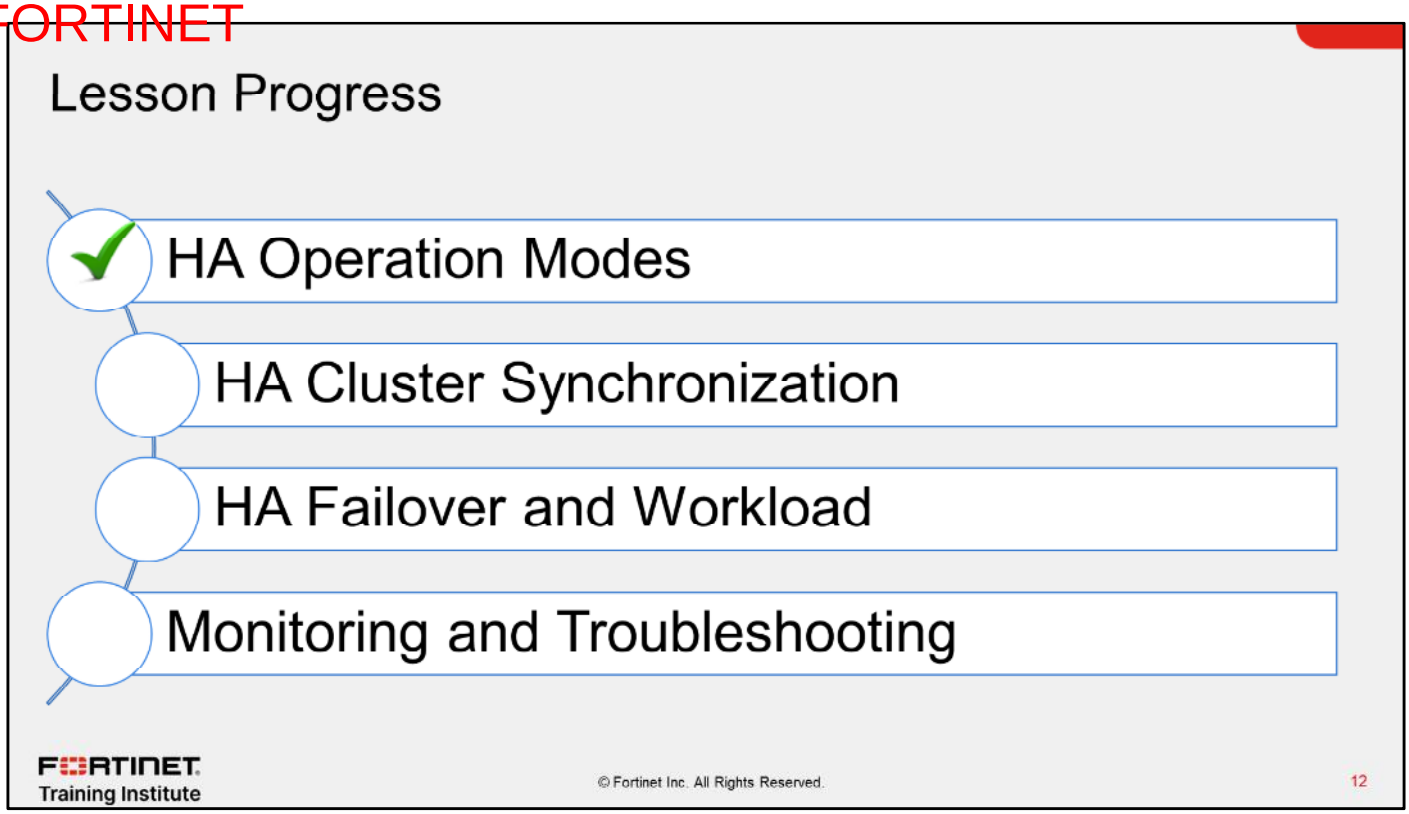
Knowledge Check

1. What is a requirement for members to form an HA cluster?
   - A. They must have same host name
   - ✓ B. They must run the same firmware version

2. What is the default order criteria (override disabled) for selecting the primary in an HA cluster?
   - ✓ A. Connected monitored ports > HA uptime > priority > serial number
   - B. Priority > HA uptime > connected monitored ports > serial number

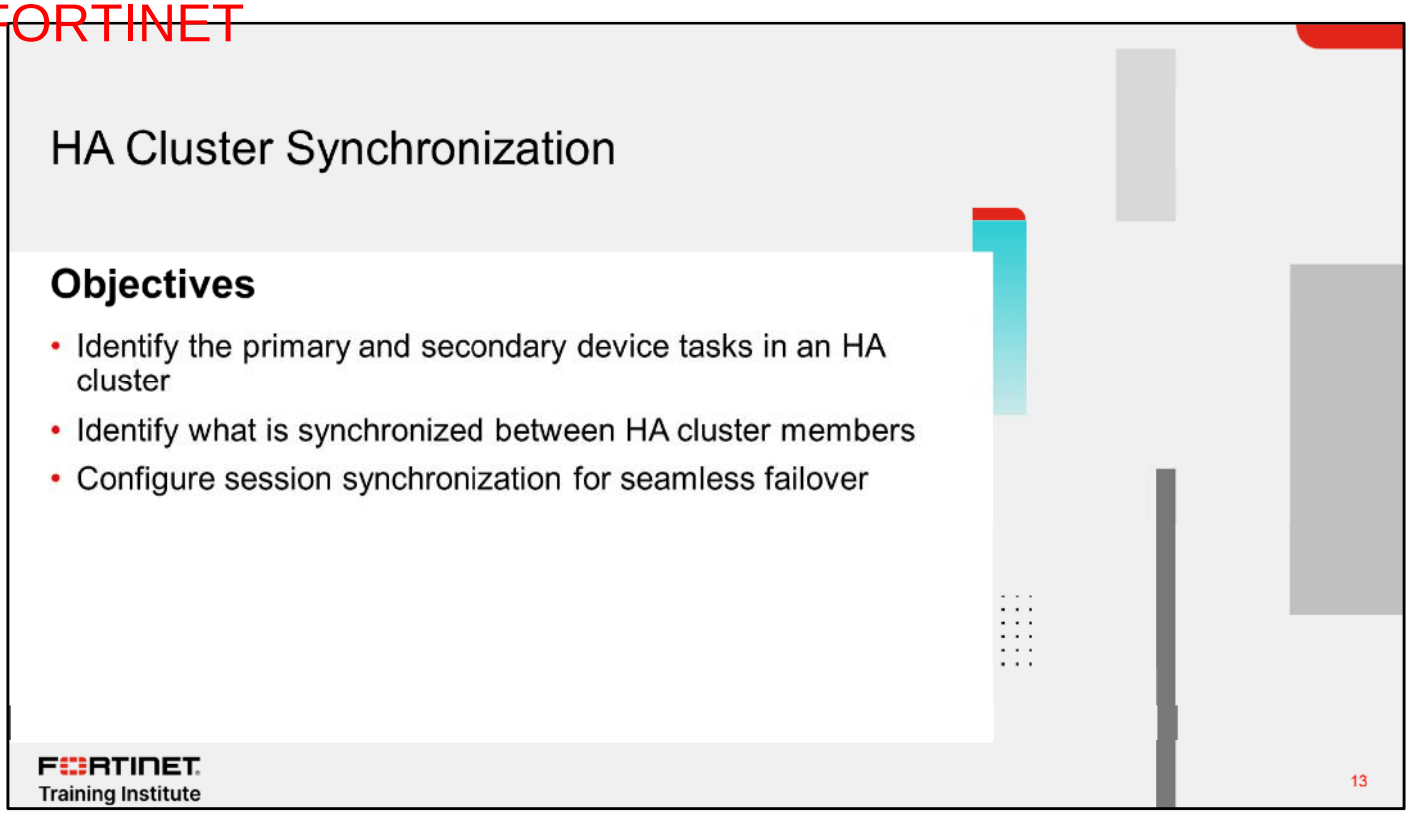
FORTINET Training Institute

11



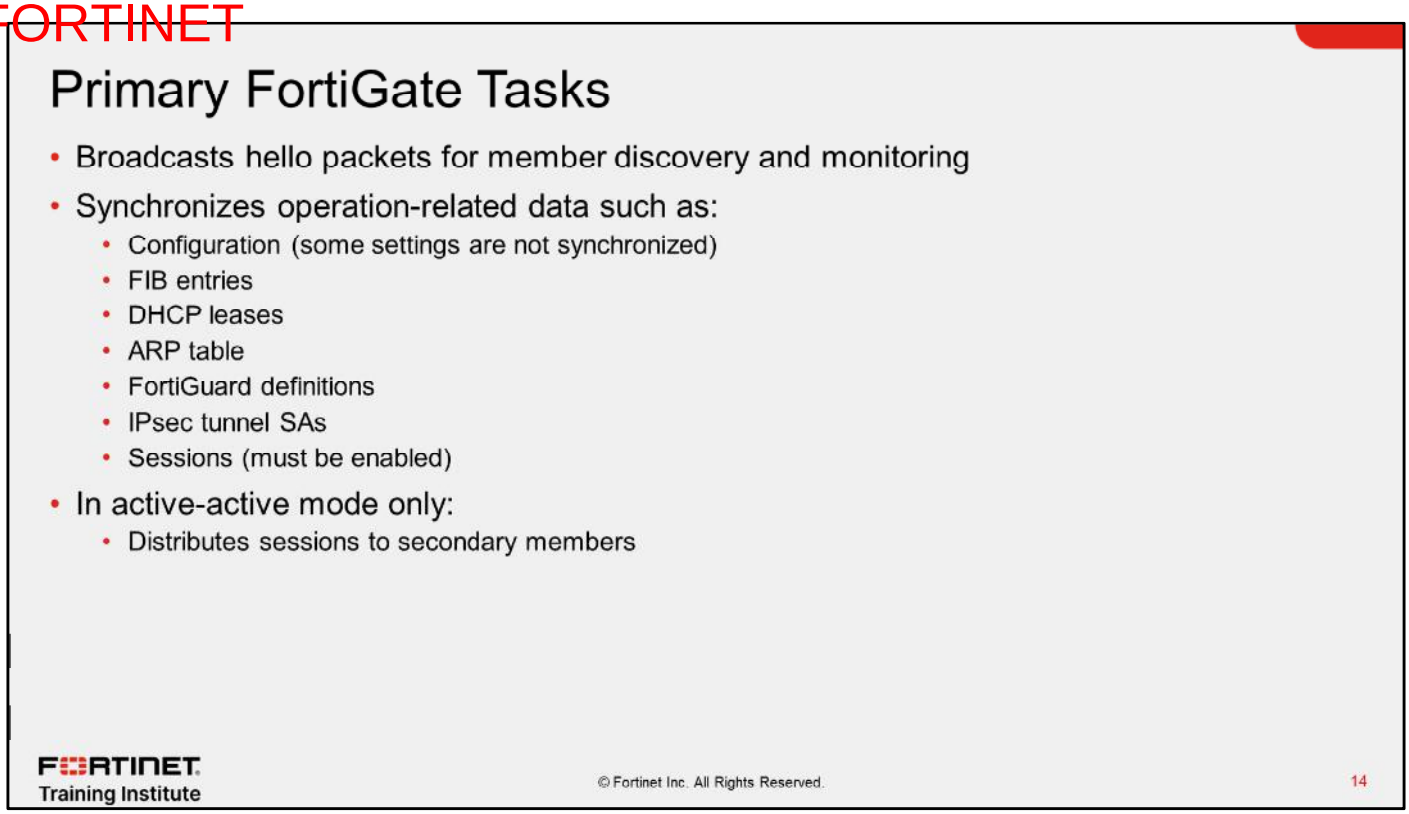
Good job! You now understand HA operation modes and the election of the primary FortiGate in an HA cluster.

Now, you will learn about HA cluster synchronization.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in cluster synchronization, you will be able to identify the tasks assigned to members based on their roles, as well as what information is synchronized between members. You will also learn how to configure session synchronization to perform session failover to specific types of traffic.



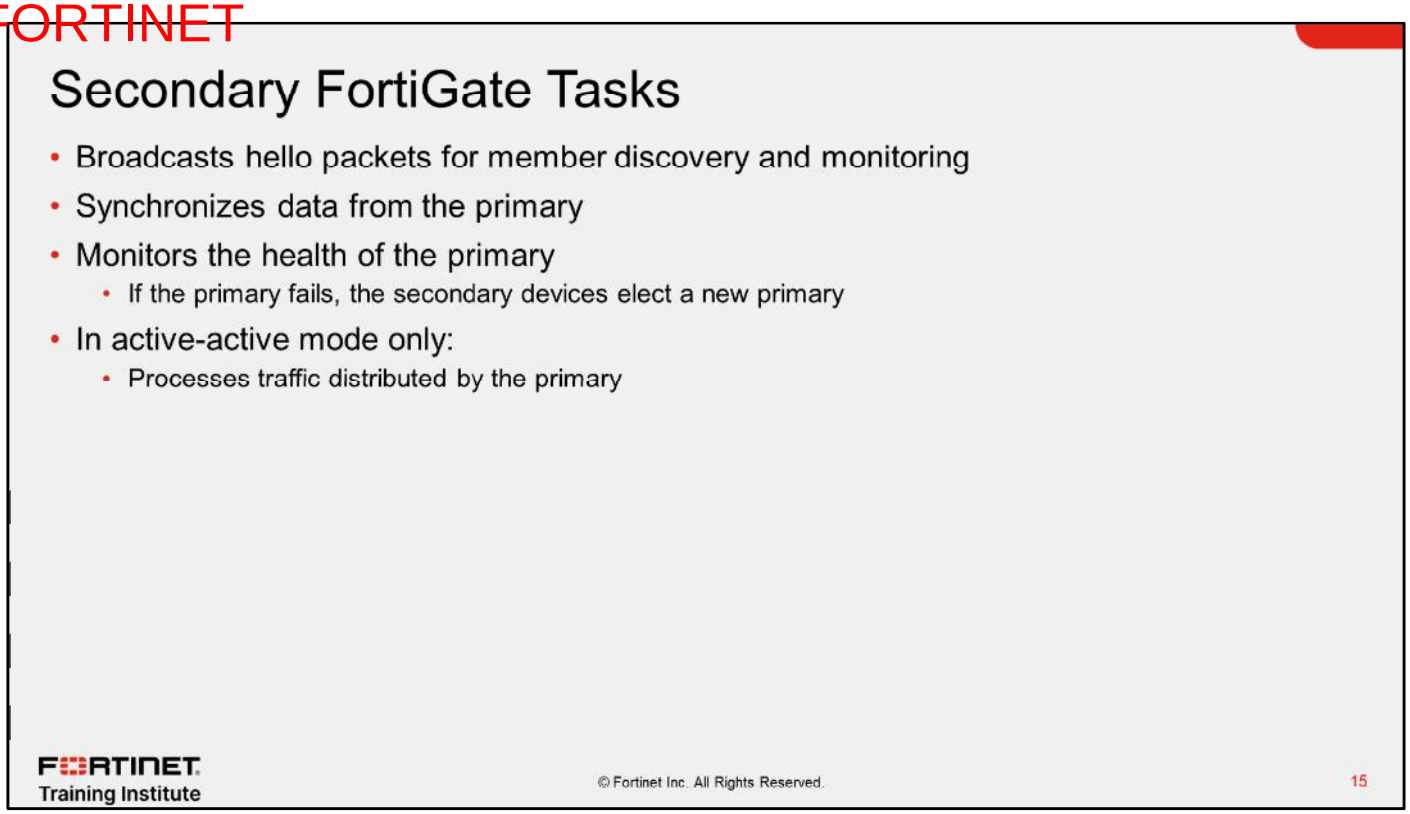
So, what are the tasks of a primary FortiGate?

It monitors the cluster by broadcasting hello packets and listening for hello packets from other members in the cluster. The members use the hello packets to identify if other FortiGate devices are alive and available.

The primary FortiGate also synchronizes its operation-related data to the secondary members. Some of the data synchronized includes its configuration, FIB entries, DHCP leases, ARP table, FortiGuard definitions, and IPsec tunnel security associations (SAs). Note that some parts of the configuration are not synchronized because they are device-specific. For example, the host name, HA priority, and HA override settings are not synchronized.

Optionally, you can configure the primary FortiGate to synchronize qualifying sessions to all the secondary devices. When you enable session synchronization, the new primary can resume communication for sessions after a failover event. The goal is for existing sessions to continue flowing through the new primary FortiGate with minimal or no interruption. You will learn which types of sessions you can enable synchronization for later in the lesson.

In active-active mode only, a primary FortiGate is also responsible for distributing sessions to secondary members.

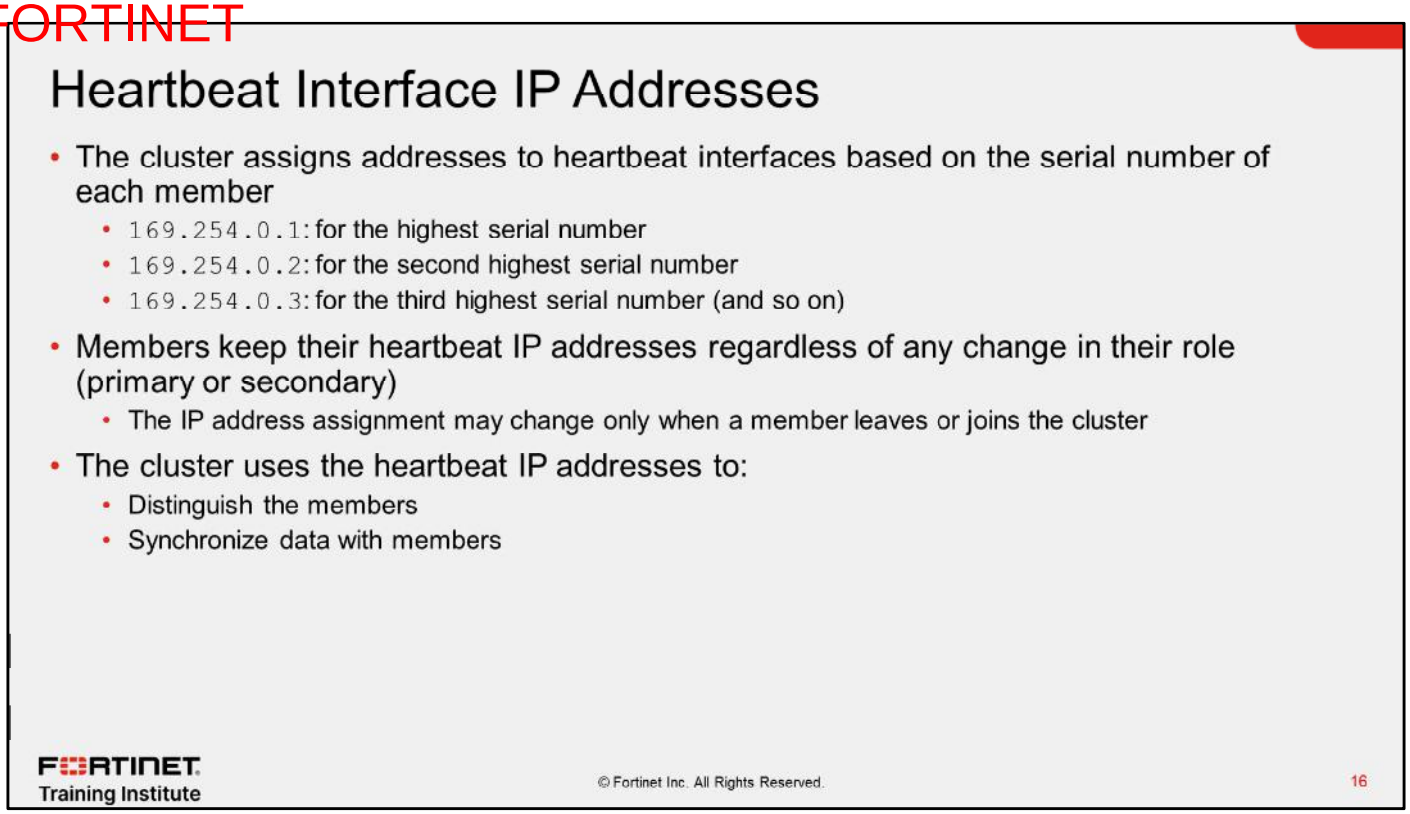


Now, take a look at the tasks of secondary FortiGate devices.

Like the primary, secondary members also broadcast hello packets for discovery and monitoring purposes.

In addition, in active-passive mode, the secondary devices act as a standby device, receiving synchronization data but not actually processing any traffic. If the primary FortiGate fails, the secondary devices elect a new primary.

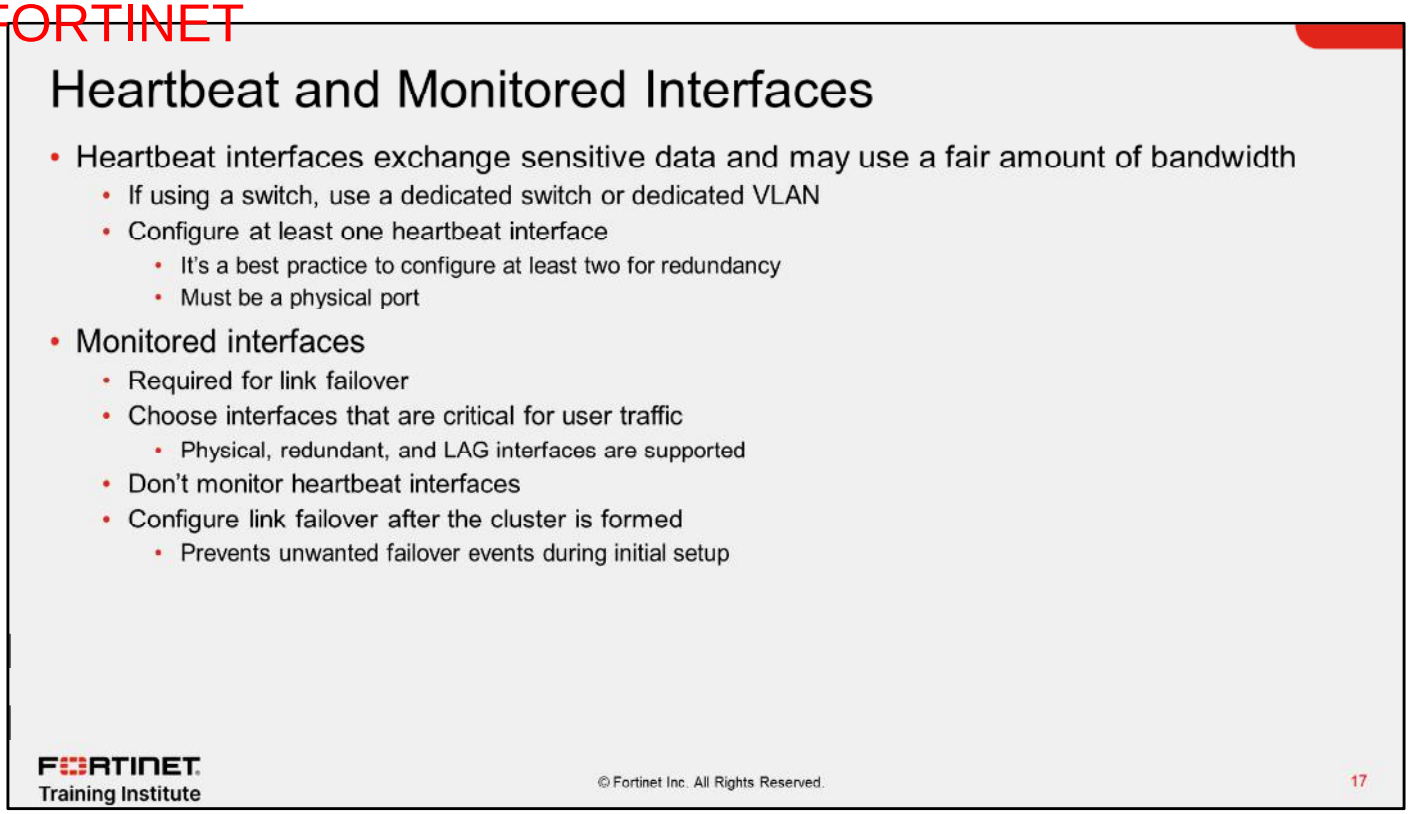
In active-active mode, the secondary devices don’t wait passively. They process all traffic assigned to them by the primary device.



FGCP automatically assigns the heartbeat IP addresses based on the serial number of each device. The IP address 169.254.0.1 is assigned to the device with the highest serial number. The IP address 169.254.0.2 is assigned to the device with the second highest serial number, and so on. The IP address assignment does not change when a failover happens. Regardless of the device role at any time (primary or secondary), its heartbeat IP address remains the same.

A change in the heartbeat IP addresses may happen when a FortiGate device joins or leaves the cluster. In those cases, the cluster renegotiates the heartbeat IP address assignment, this time taking into account the serial number of any new device, or removing the serial number of any device that left the cluster.

The HA cluster uses the heartbeat IP addresses to distinguish the cluster members and synchronize data.

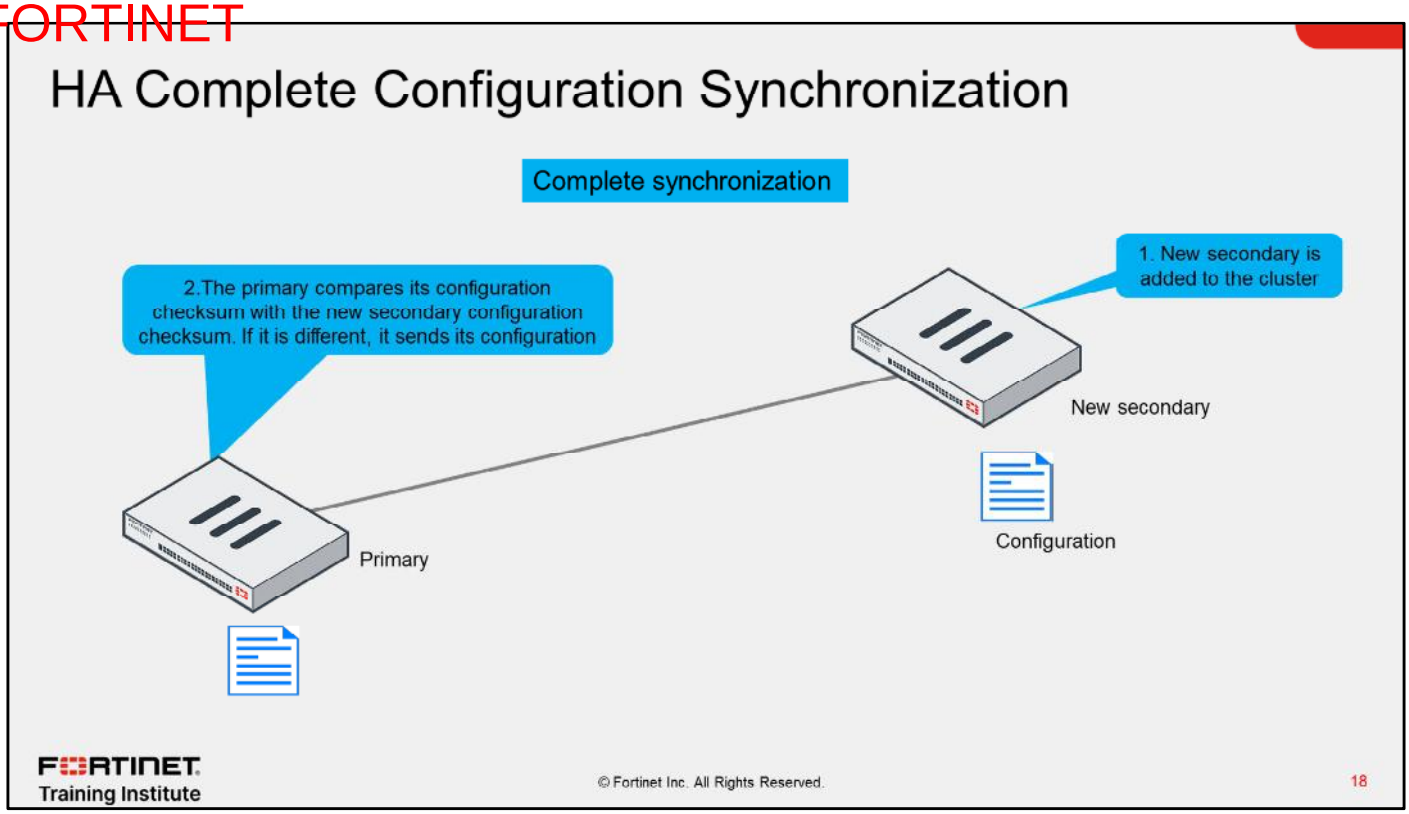


Heartbeat interfaces exchange sensitive information about the cluster operation and may require a fair amount of bandwidth for data synchronization. For this reason, if you use a switch to connect the heartbeat interfaces, it's recommended that you use a dedicated switch or, at least, that you place the heartbeat traffic on a dedicated VLAN.

In addition, you must configure at least one port as a heartbeat interface, but preferably two for redundancy. For heartbeat interfaces, you can use physical interfaces only. That is, you can't use VLAN, IPsec VPN, redundant, or 802.3ad aggregate interfaces. You cannot use FortiGate switch ports either.

For link failover to work, you must configure one or more monitored interfaces. A monitored interface should be an interface whose failure has a critical impact in the network, and therefore, should trigger a device failover. For example, your LAN or WAN interfaces are usually good choices for monitored interfaces. Heartbeat interfaces, however, should not be configured as monitored interfaces because they are not meant to handle user traffic. Note that you can monitor physical ports, redundant interfaces, and link aggregation group (LAG) interfaces.

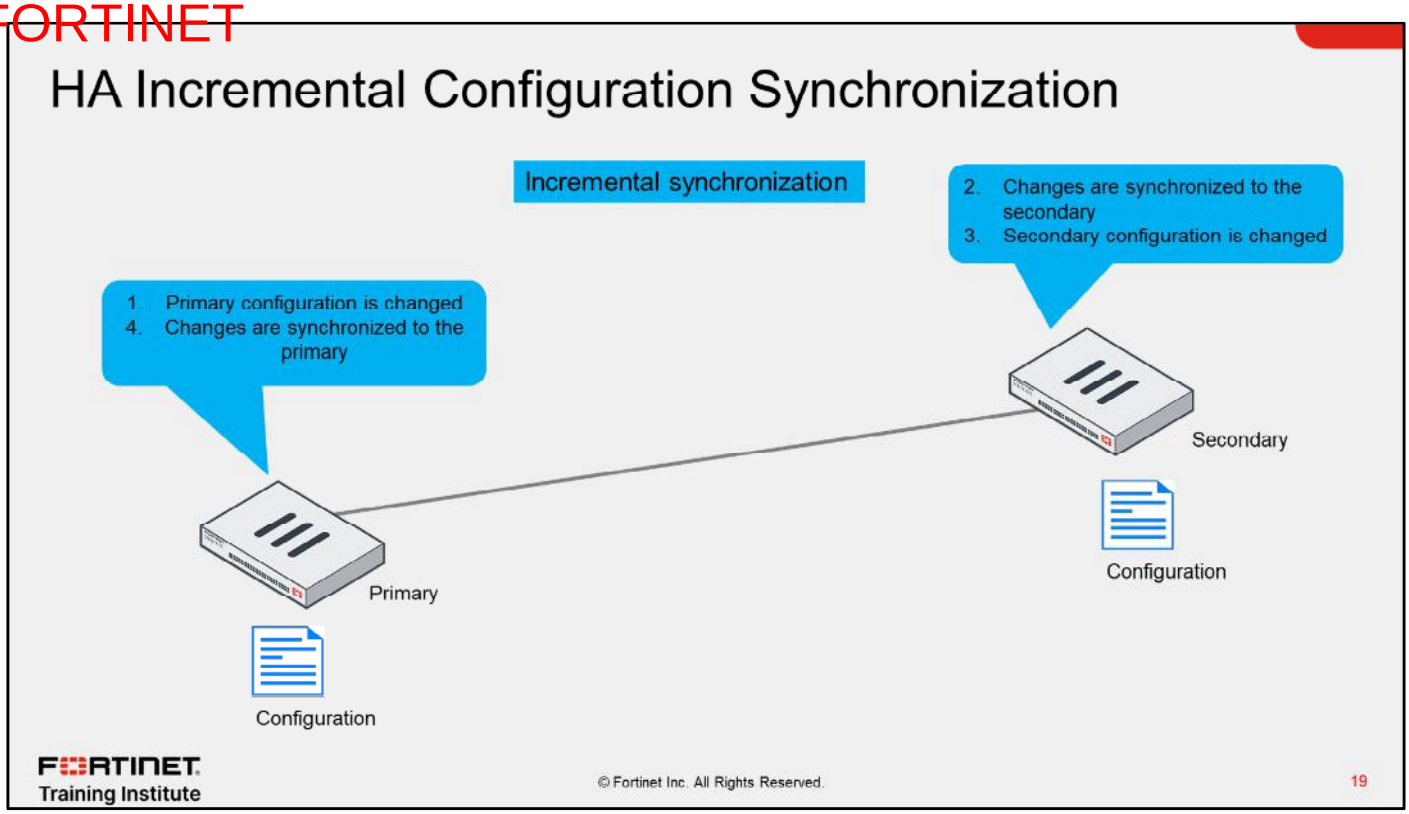
As a best practice, wait until a cluster is up and running and all interfaces are connected before configuring link failover. This is because a monitored interface can be disconnected during the initial setup and, as a result, trigger a failover before the cluster is fully configured and tested.



To prepare for a failover, an HA cluster keeps its configurations in sync. You will explore that now.

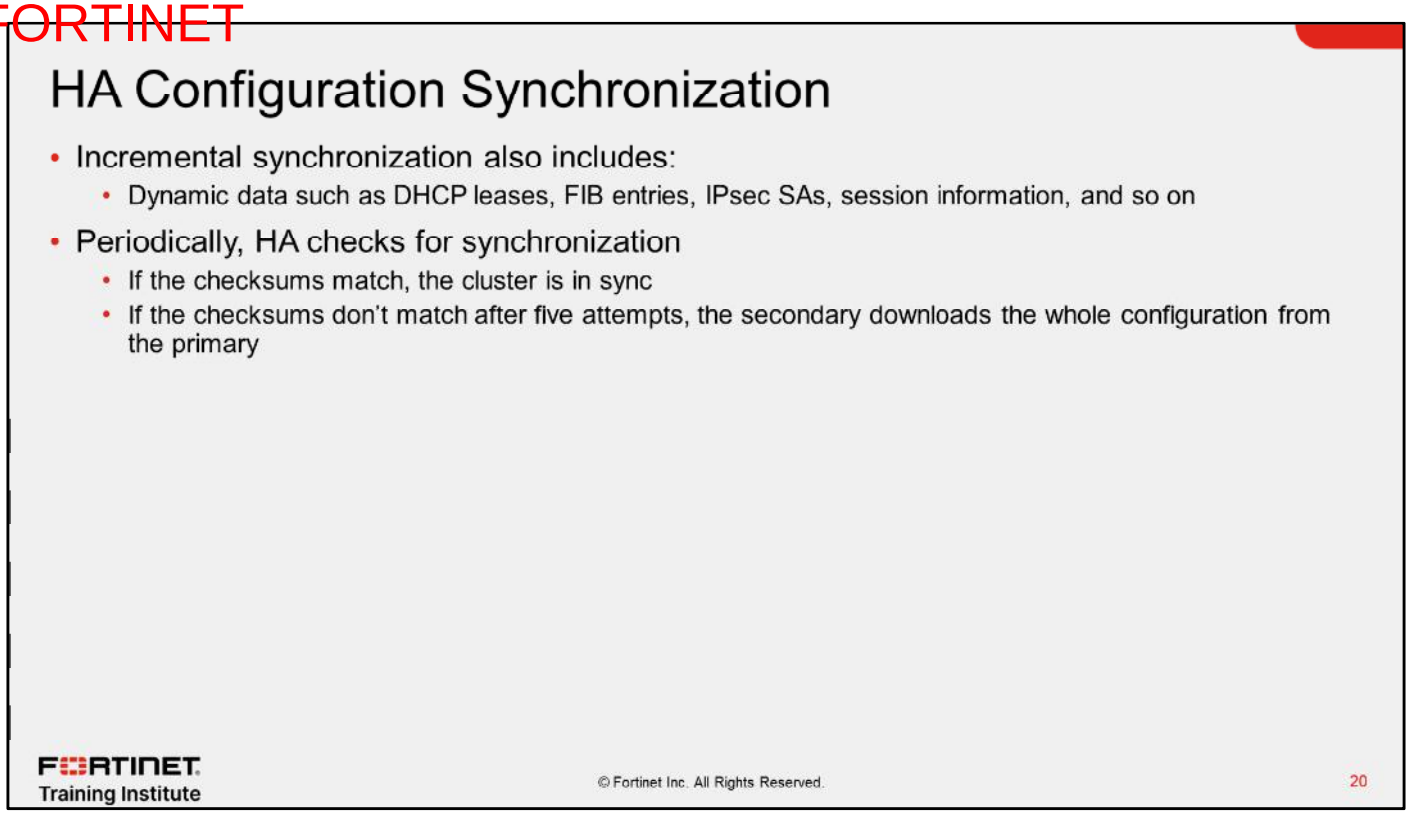
FortiGate HA uses a combination of both incremental and complete synchronizations.

When you add a new FortiGate to the cluster, the primary FortiGate compares its configuration checksum with the new secondary FortiGate configuration checksum. If the checksums don't match, the primary FortiGate uploads its complete configuration to the secondary FortiGate.



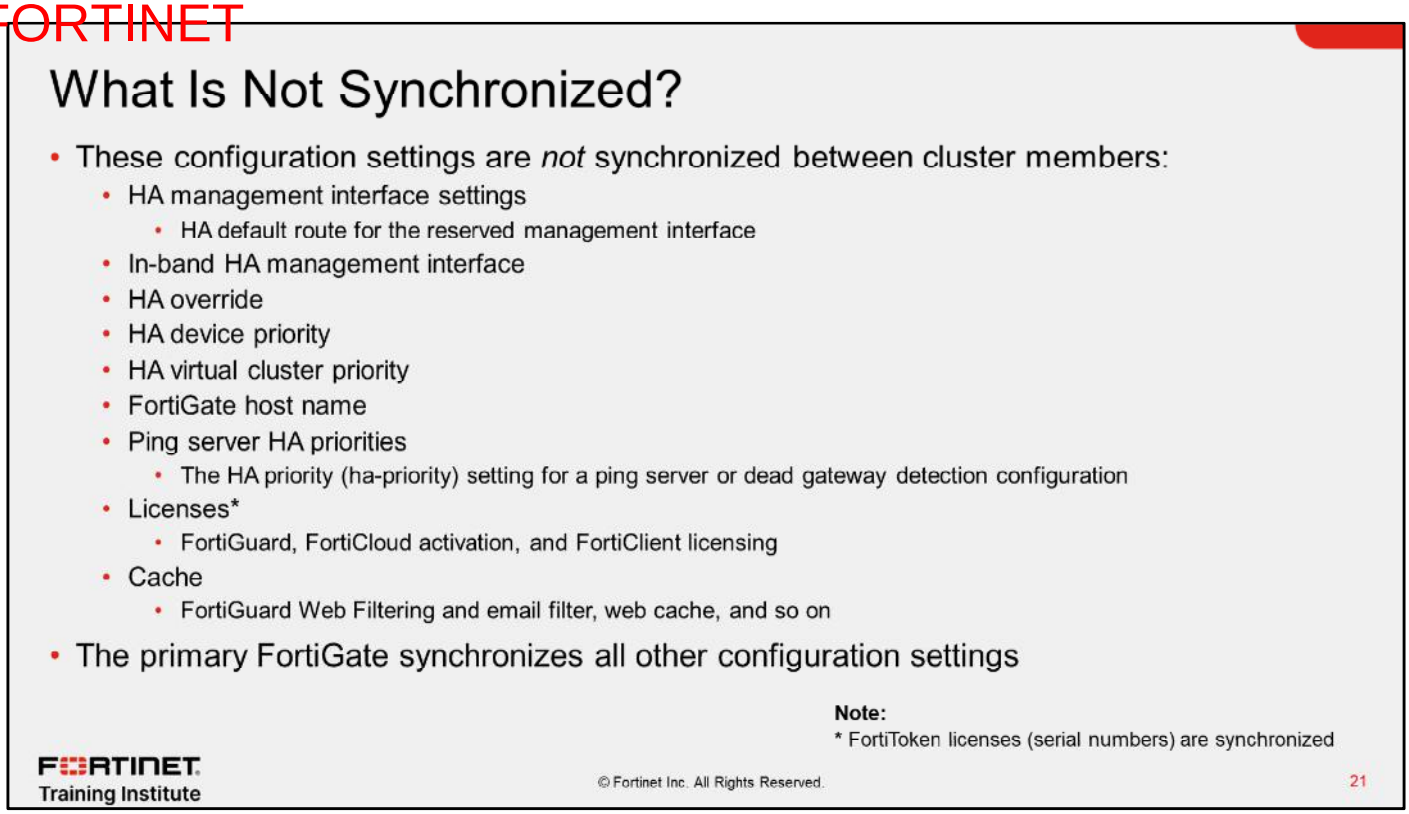
After the initial synchronization is complete, whenever a change is made to an HA cluster device's (primary or secondary) configuration, incremental synchronization sends the same configuration change to all other cluster devices over the HA heartbeat link. An HA synchronization process running on each cluster device receives the configuration change and applies it to the cluster device. For example, if you create a firewall address object, the primary doesn't resend its complete configuration—it sends only the new object.

Another example is in an HA setup with multiple VDOMs and virtual clustering, where the secondary device is acting as the primary FortiGate for VDOM2. Any changes made on VDOM2 is synchronized with the primary FortiGate.



HA propagates more than just configuration details. Some runtime data, such as DHCP leases and FIB entries, are also synchronized.

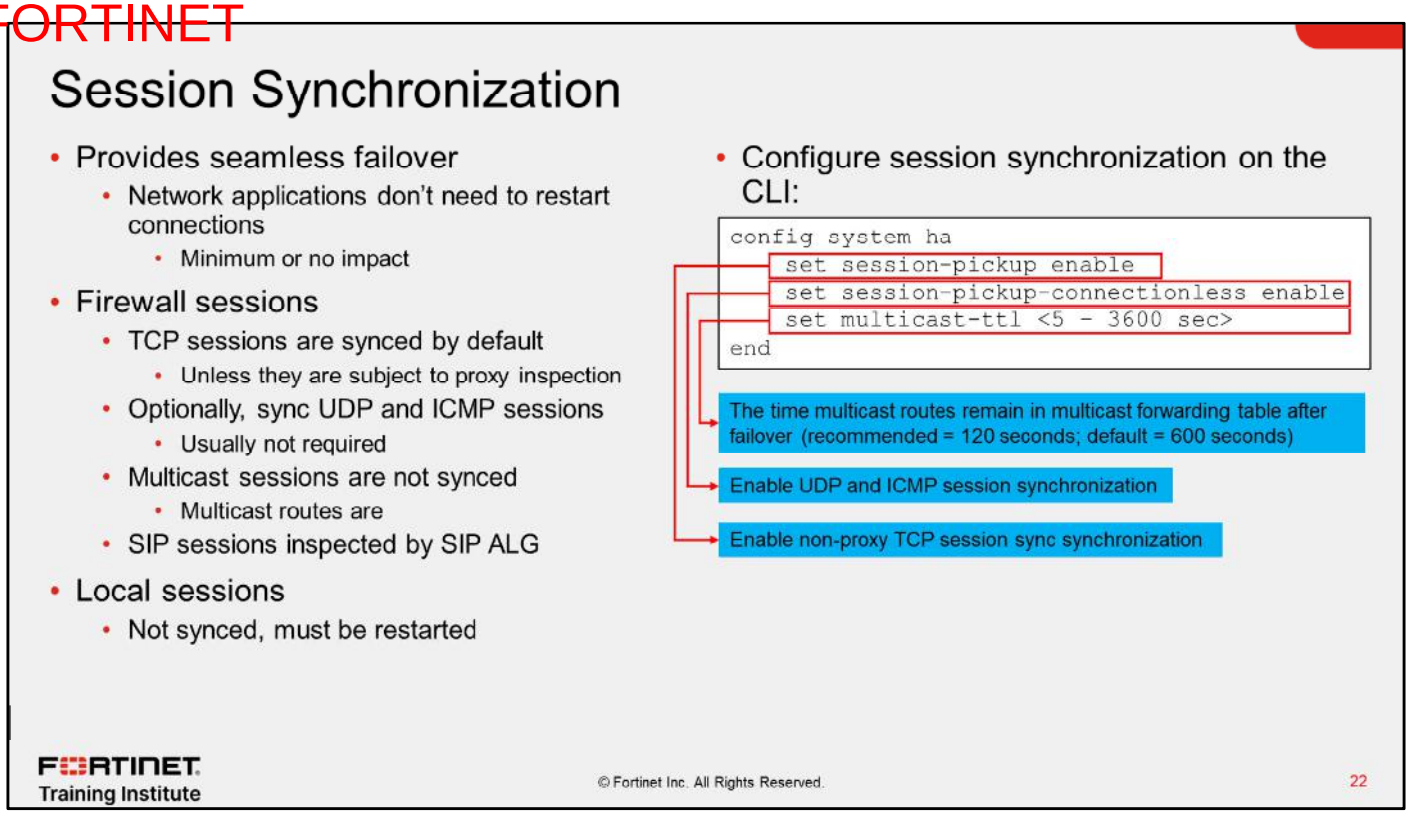
By default, the cluster checks every 60 seconds to ensure that all devices are synchronized. If a secondary device is out of sync, its checksum is checked every 15 seconds. If the checksum of the out-of-sync secondary device doesn't match for five consecutive checks, a complete resynchronization to that secondary device is done.



Not all the configuration settings are synchronized. There are a few that are not, such as:

- System interface settings of the HA reserved management interface and the HA default route for the reserved management interface
- In-band HA management interface
- HA override
- HA device priority
- Virtual cluster priority
- FortiGate host name
- HA priority setting for a ping server (or dead gateway detection) configuration
- All licenses except FortiToken licenses (serial numbers)
- Cache

The primary FortiGate synchronizes all other configuration settings, including all other HA settings.



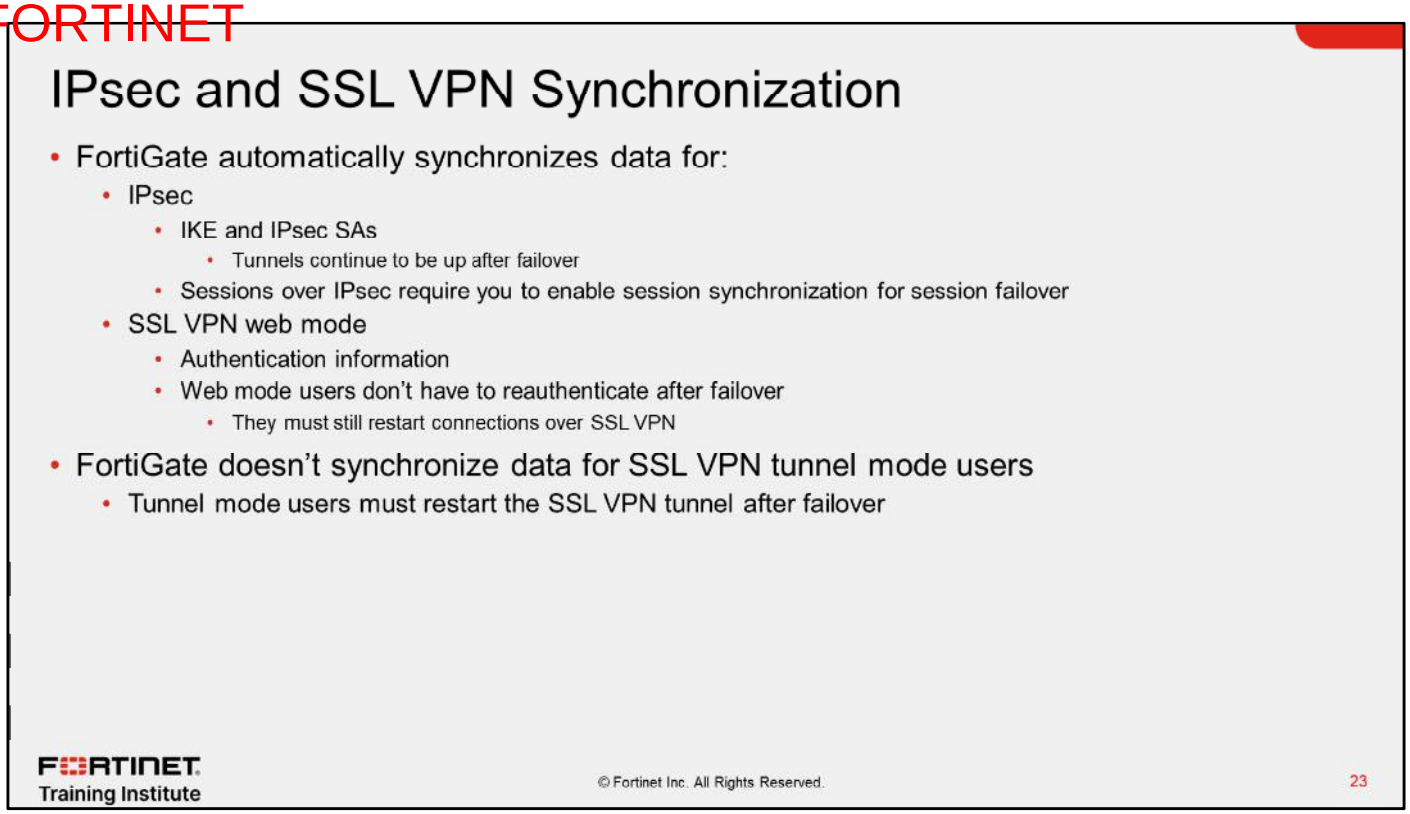
Session synchronization provides seamless session failover. When the primary fails, the new primary can resume traffic for synchronized sessions without network applications having to restart the connections.

By default, the feature synchronizes TCP firewall sessions that are not subject to proxy-based inspection. An exception to this rule is TCP SIP sessions inspected by SIP ALG. Even though SIP ALG performs proxy-based inspection on SIP sessions, FortiGate can still synchronize such SIP sessions. Firewall sessions, also known as pass-through sessions, are user traffic sessions that travel across FortiGate. TCP firewall sessions that are subject to flow-based inspection or no inspection at all, are synchronized to secondary members.

You can also enable the synchronization of UDP and ICMP sessions. Although both protocols are connectionless protocols, FortiGate still allocates sessions for UDP and ICMP connections in its session table. Usually, the synchronization of UDP and ICMP sessions is not required because most UDP and ICMP connections can resume communication if their session information is lost.

For multicast traffic, FortiGate synchronizes multicast routes only. That is, FortiGate doesn't synchronize multicast sessions, which should be fine because multicast sessions are mostly UDP-based and, as mentioned before, UDP sessions can usually resume communication if their session information is lost. To ensure the multicast routing information across members is accurate, you can adjust the multicast time to live (TTL) timer. The timer controls how long the new primary keeps the synced multicast routes in the multicast forwarding table. The smaller the timer value, the more often the routes are refreshed, and so the more accurate the multicast forwarding table is. The recommended timer value is 120 seconds.

Local-in and local-out sessions, which are sessions that are terminated at or initiated by FortiGate, respectively, are not synchronized either. For example, BGP peerings, OSPF adjacencies, as well as SSH and HTTPS management connections must be restarted after a failover.



The primary FortiGate automatically synchronizes all IKE and IPsec security associations (SAs) to secondary members. This enables the new primary to resume existing IPsec tunnels after a failover. Note that you must also enable session synchronization if you want the new primary to also resume existing IPsec sessions. Otherwise, after a failover, you must still restart existing TCP connections made over IPsec tunnels, even though the IPsec tunnels continue to be up on the new primary.

For SSL VPN, the primary FortiGate synchronizes the authentication information for SSL VPN web mode users only. That is, they are using SSL VPN web mode, the SSL VPN users don't have to authenticate again after a failover. However, the users must still restart the connections made using SSL VPN web mode to regain access to protected resources. Note that FortiGate doesn't synchronize any information for SSL VPN tunnel mode. That is, after a failover, SSL VPN tunnel mode users must restart their SSL VPN tunnel connection, as well as any connection made through the tunnel.

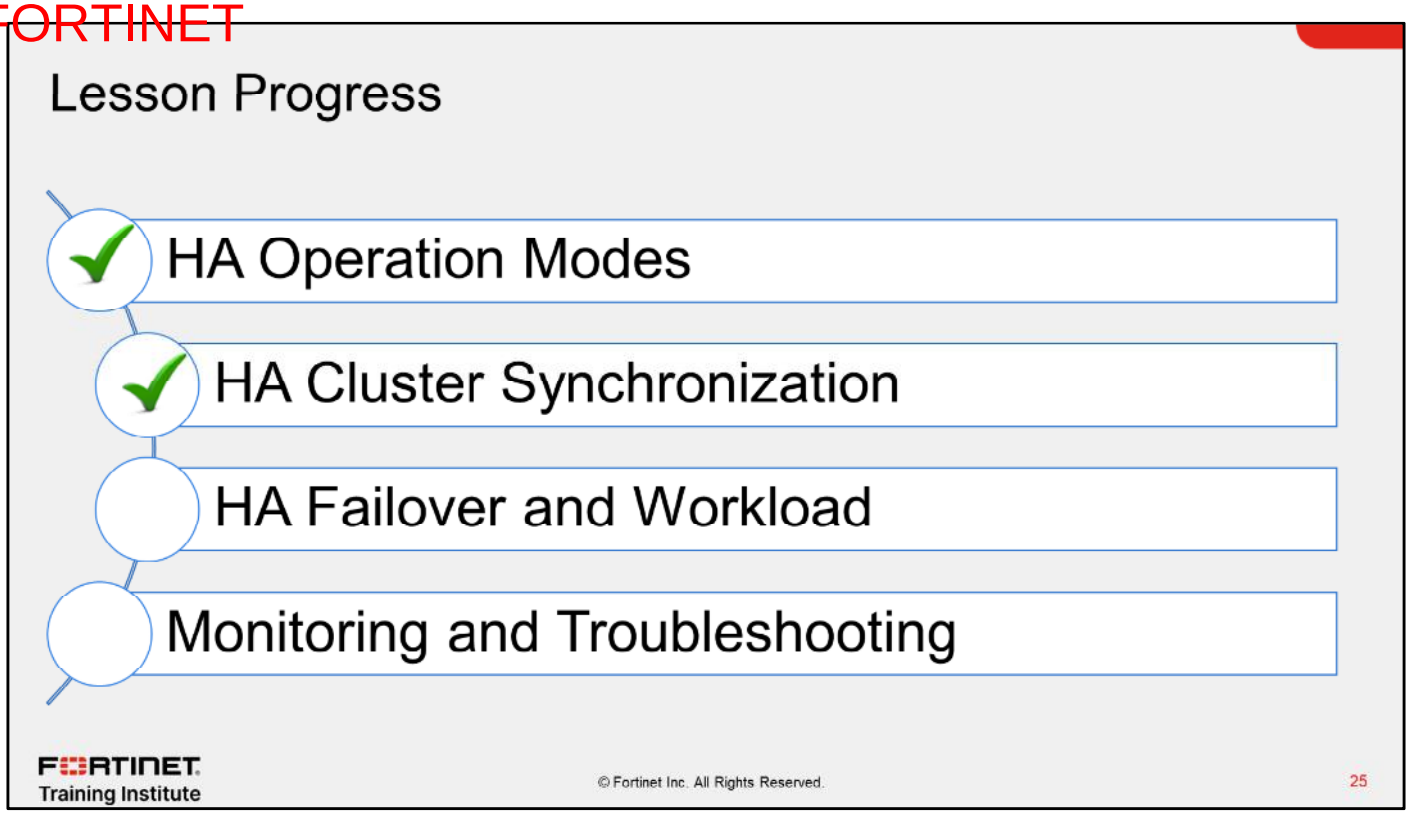
Knowledge Check

1. Which information is synchronized in an HA cluster?
   - ✓ A. Firewall policies and objects
   - B. FortiGate host name

2. Which one of the following session types can be synchronized in an HA cluster?
   - A. BGP peerings
   - ✓ B. Non-proxy TCP sessions

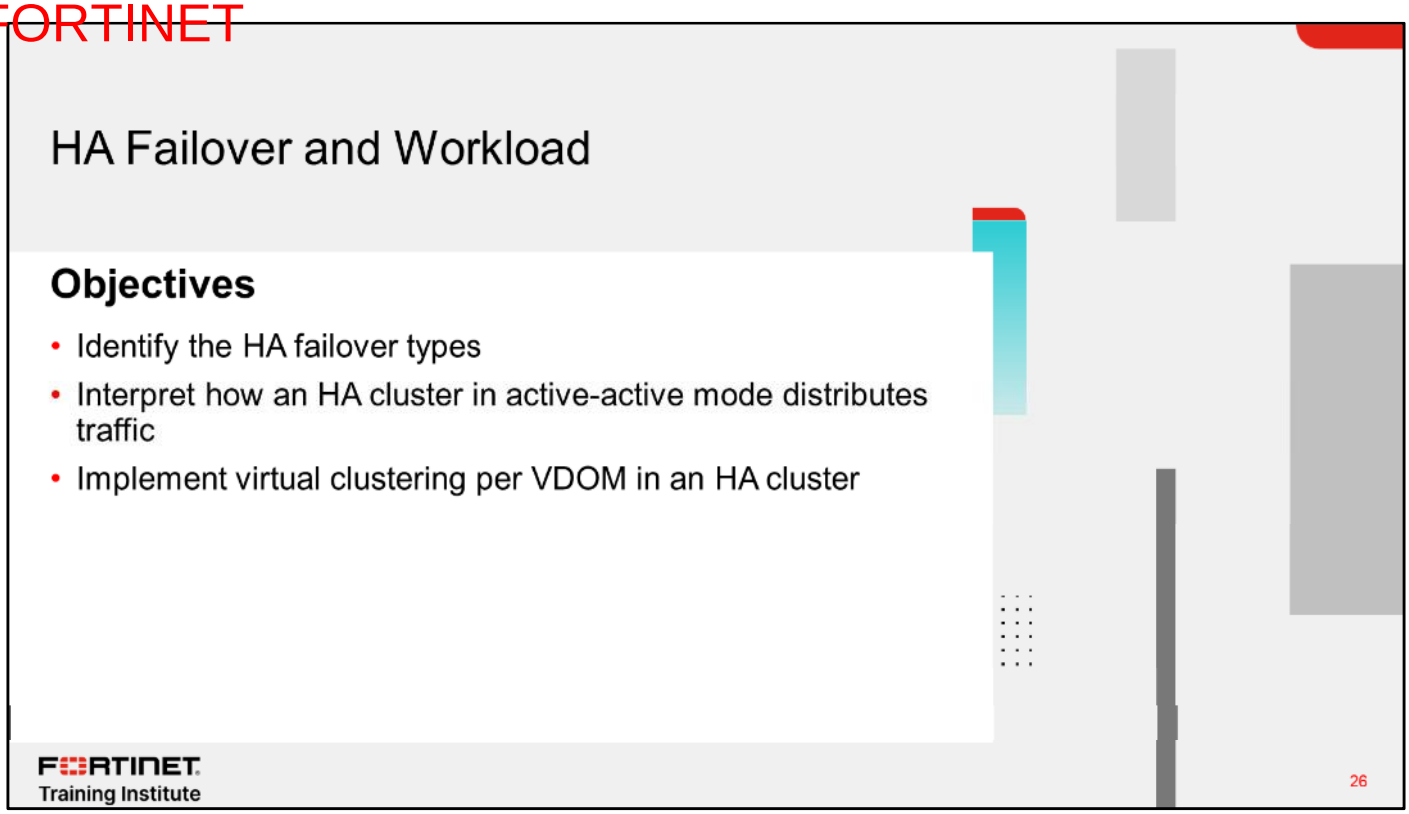
FORTINET Training Institute

24



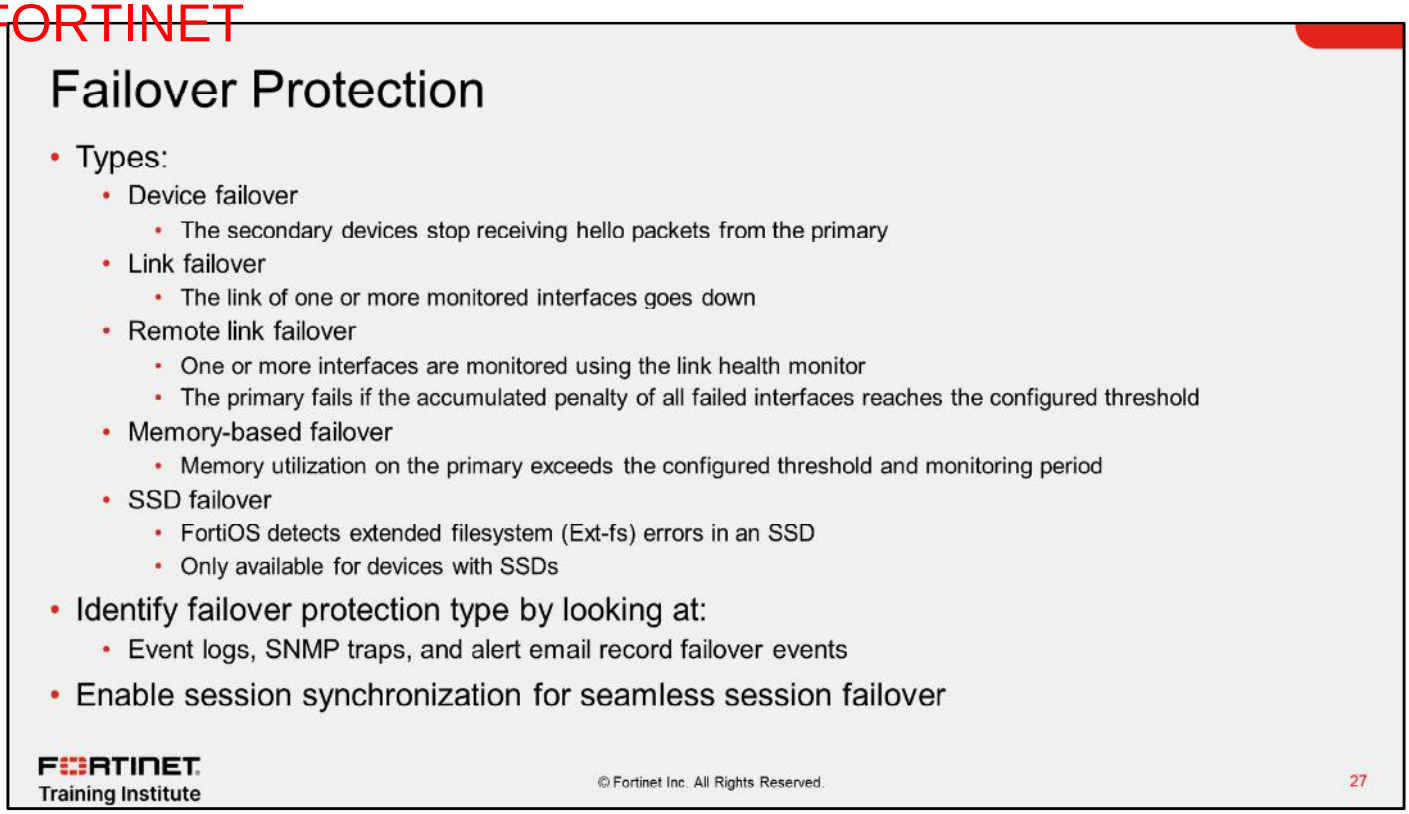
Good job! You now understand HA cluster synchronization.

Now, you will learn about HA cluster failover protection types and workload for primary and secondary FortiGate devices in an HA cluster.



After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in failover types and workload, you will be able to identify how enhanced reliability is achieved through HA failover protection. You will also learn about the distribution of traffic in an active-active cluster and distributing traffic using virtual clustering.



The most common types of failovers are device failovers and link failovers. However, you can also configure remote link failover and memory-based failover. When a failover event is triggered, the secondary devices elect a new primary.

A device failover is triggered when the secondary devices stop receiving the heartbeat hello packets from the primary.

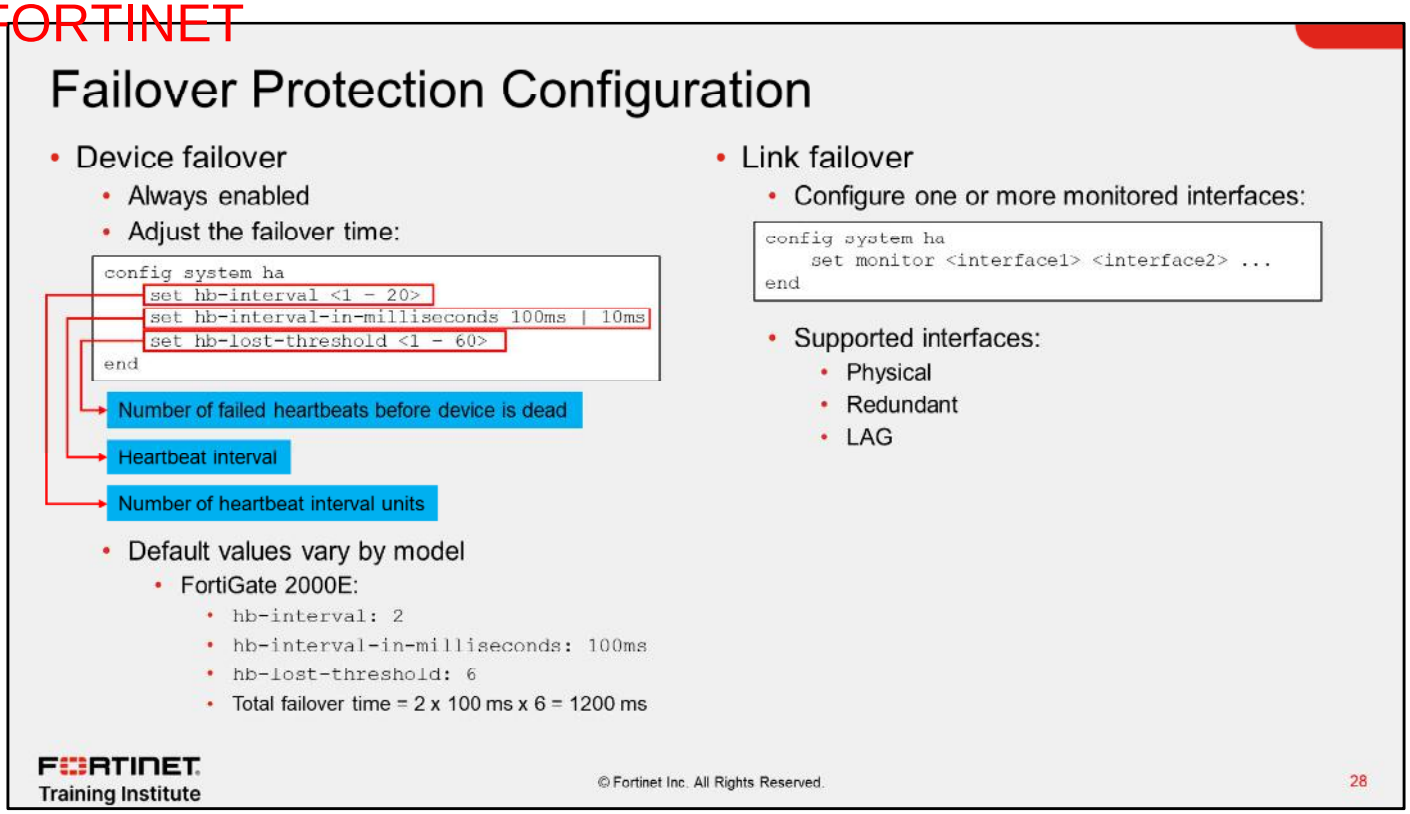
A link failover occurs when the link status of a monitored interface on the primary FortiGate goes down. You can configure an HA cluster to monitor one or more interfaces. If a monitored interface on the primary FortiGate is unplugged, or its link status goes down, a new primary FortiGate is elected.

When you configure remote link failover, FortiGate uses the link health monitor feature to monitor the health of one or more interfaces against one or more servers that act as beacons. The primary FortiGate fails if the accumulated penalty of all failed interfaces reaches the configured threshold.

If you enable memory-based failover, an HA failover is triggered when the memory utilization on the primary FortiGate reaches the configured threshold for the configured monitoring period. You can also enable SSD failover, which triggers a failover if FortiOS detects Ext-fs errors on an SSD on the primary FortiGate.

There are multiple events that might trigger an HA failover, such as a hardware or software failure on the primary FortiGate or an issue on one of the interfaces on the primary. When a failover occurs, an event log is generated. Optionally, you can configure the device to also generate SNMP traps and alert emails.

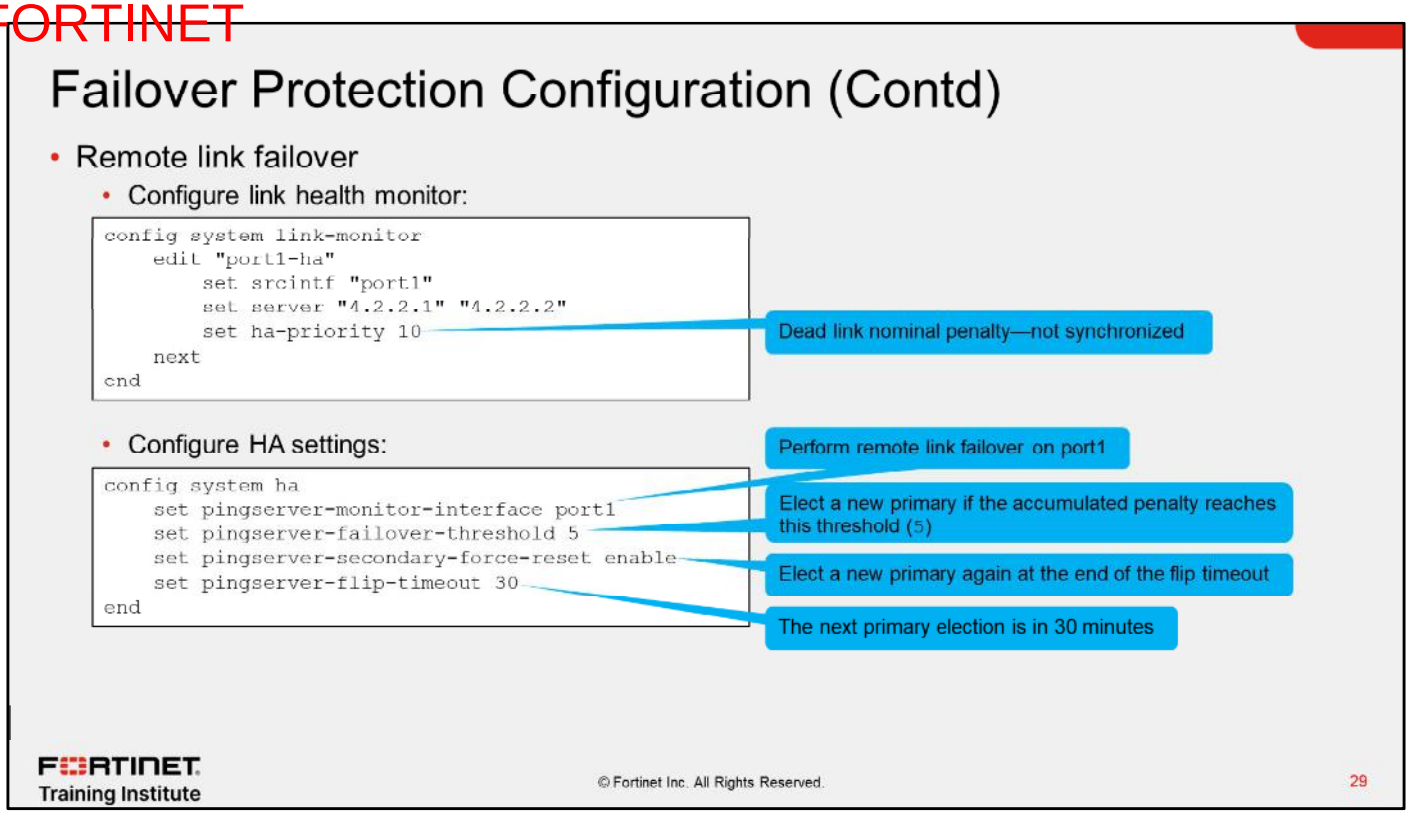
Make sure that you enable session pickup for sessions you want to protect from a failover event. This way, the new primary can resume traffic for these sessions.



When you configure HA, device failover is always enabled. However, you can adjust the settings that dictate the failover time. To speed up failover, you can reduce the values for all three settings shown on this slide. To reduce false positives, increase their values.

The default values for the three settings vary by model. For example, using the default values on a FortiGate 2000E model results in a device failover time of 1200 milliseconds (1.2 seconds). Note that the 10-millisecond heartbeat interval is supported on NP6 platforms only.

To configure link failover, you must configure one or more monitored interfaces, as shown on this slide. Note that you can configure only physical, redundant, and LAG interfaces as monitored interfaces.



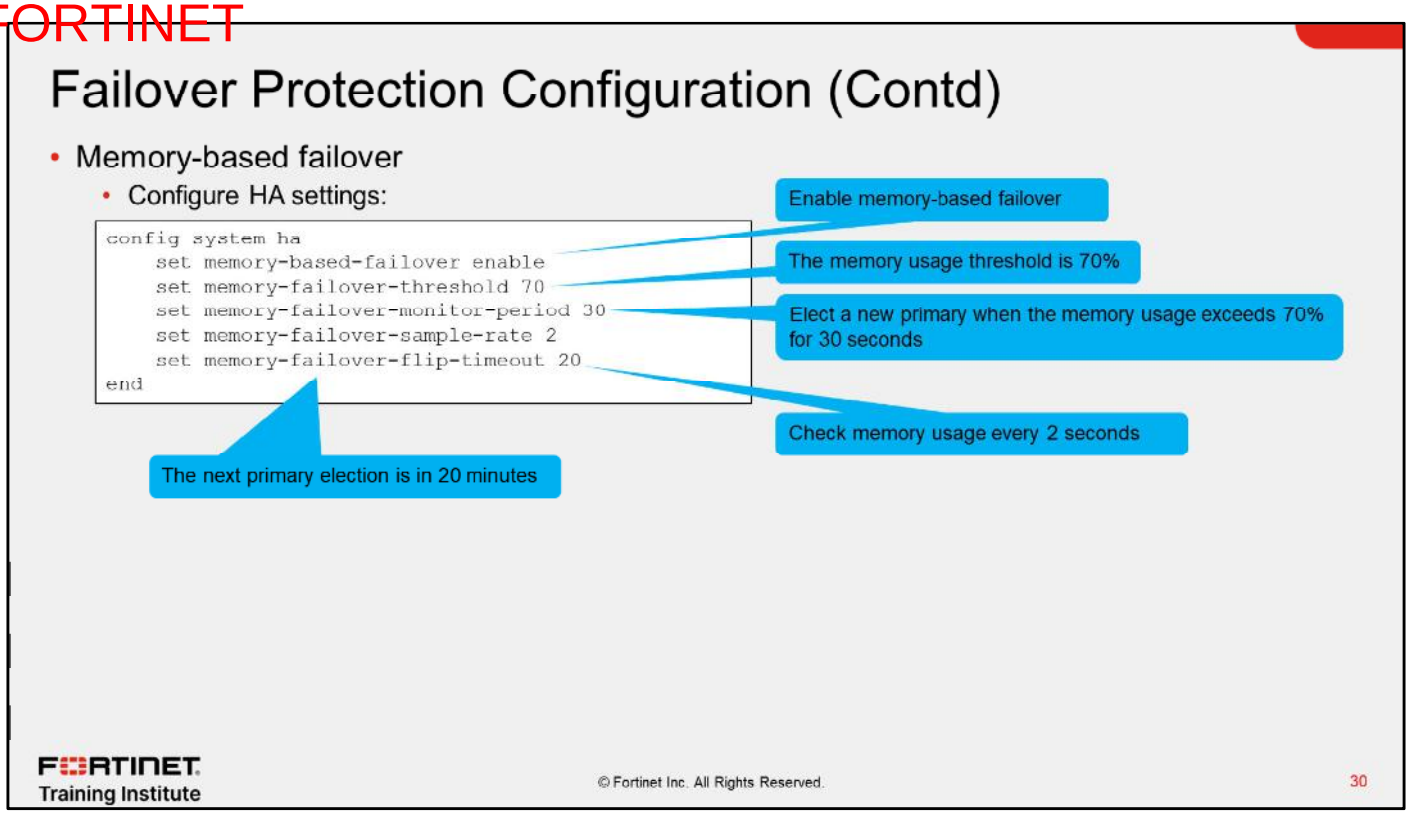
This slide shows a configuration example for remote link failover.

First, you configure link health monitor, as shown in the *Routing* lesson. The `ha-priority` setting in the link health monitor configuration defines the penalty applied to the member after the link is detected as dead. Note that the `ha-priority` setting has local significance only, and therefore, is not synchronized to other members.

The next step is to configure the HA settings related to remote link failover. The configuration on this slide instructs FortiGate to perform remote link failover on port1 as follows:

- When port1 is detected as dead, the nominal penalty (10) is added to the global penalty, which is initially set to 0.
- If the accumulated penalty reaches the penalty threshold (5), then the cluster elects a new primary. A failover occurs when a secondary member has a lower accumulated penalty than the primary. If so, the secondary member with the lowest accumulated penalty becomes the new primary.
- The cluster doesn't elect a new primary again until the flip timeout (30 minutes) has passed.

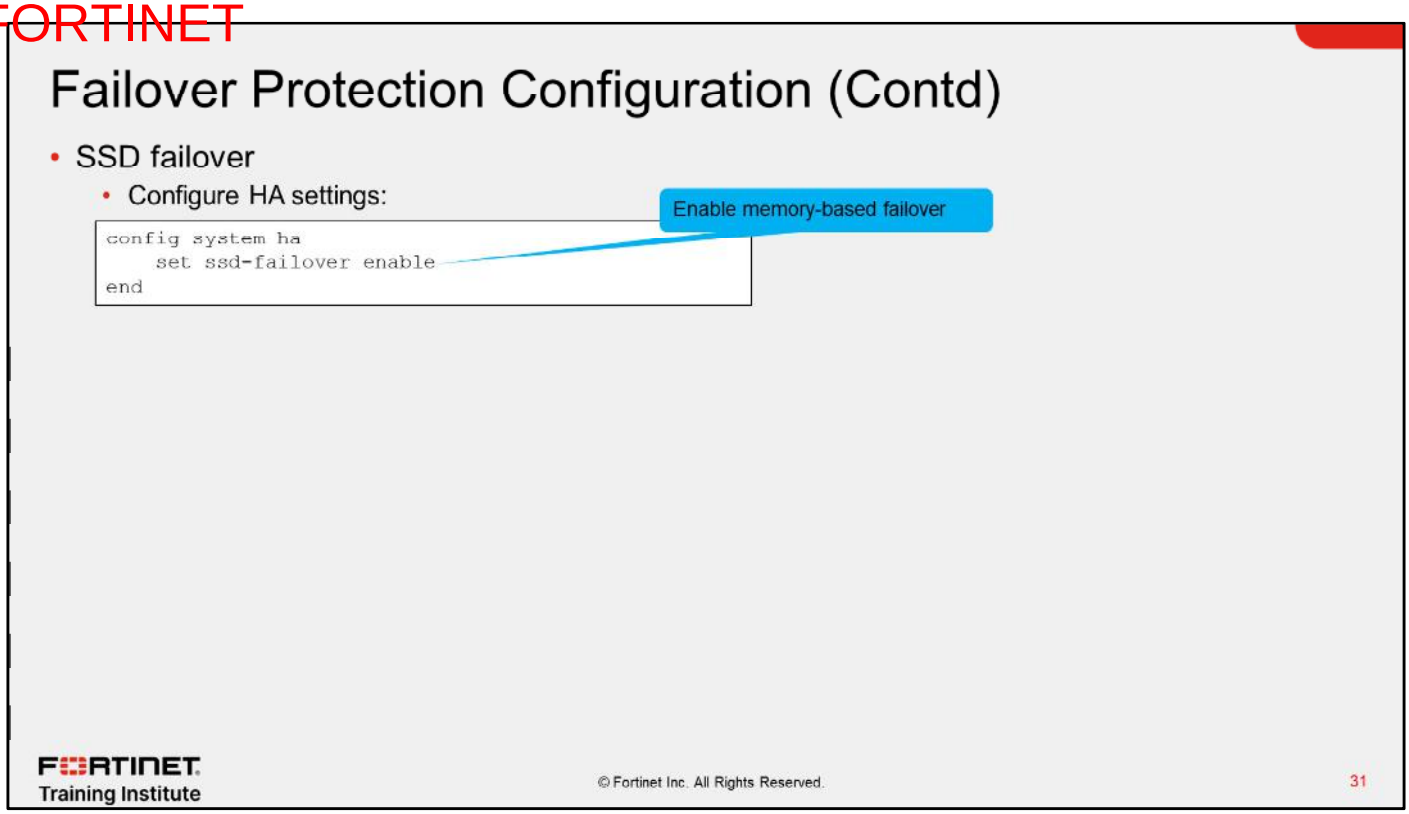
If during the primary election, the accumulated penalty of all members is the same, then other criteria, such as monitored interfaces, priority, uptime, and so on, are used as tiebreakers to elect the new primary.



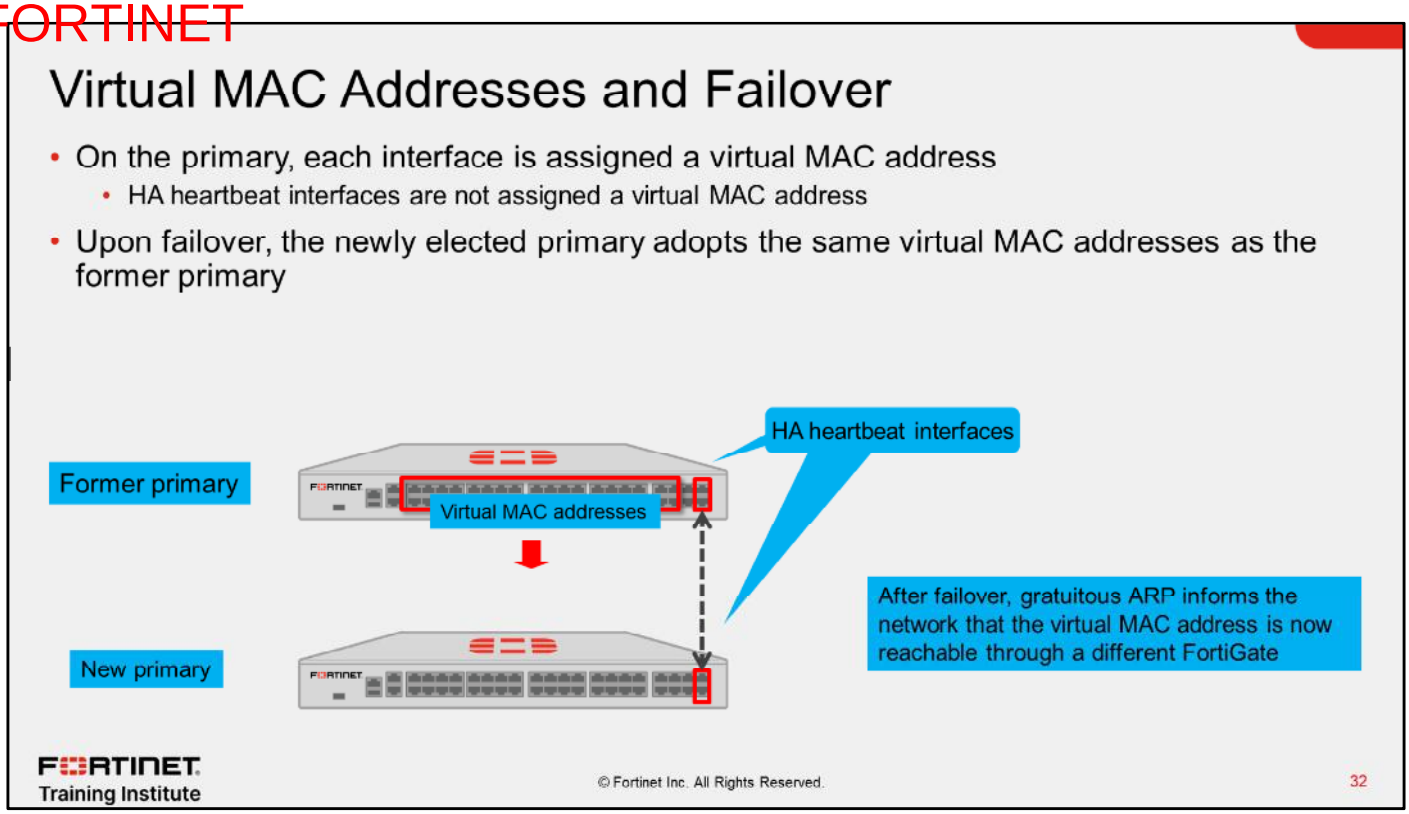
The HA configuration shown on this slide instructs FortiGate to perform memory-based failover as follows:

- When the memory on the primary reaches the threshold (70%) and stays like that for 30 seconds, then the cluster elects a new primary.
- During primary election, a failover occurs when the memory usage on a secondary member is lower than the configured memory threshold (70%). If so, the secondary member becomes the new primary.
- The cluster doesn't elect a new primary again until the flip timeout (20 minutes) has passed.
- Each member in the cluster checks its memory usage every 2 seconds.

If during the primary election, the memory usage of all members are below or above the threshold, then other criteria, such as monitored interfaces, priority, uptime, and so on, are used as tiebreakers to elect the new primary.



The HA configuration shown on this slide instructs FortiGate to perform a failover when any of the SSD disks on the primary FortiGate report Ext-fs errors. Note that this feature is supported only on FortiGate models with SSD disks.

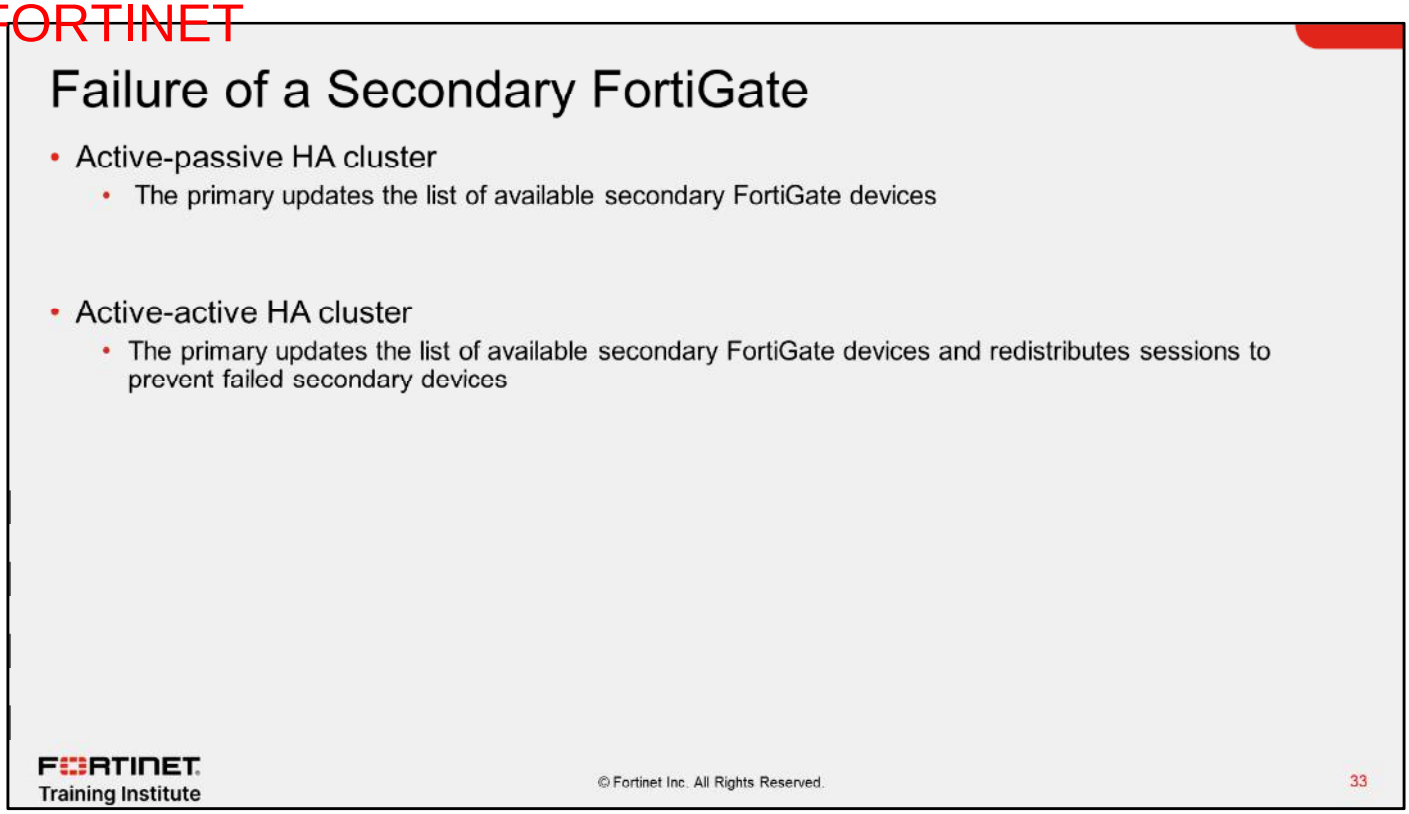


To forward traffic correctly, a FortiGate HA solution uses virtual MAC addresses. When a primary joins an HA cluster, each interface is assigned a virtual MAC address. The HA group ID is used in the creation of virtual MAC addresses assigned to each interface. So, if you have two or more HA clusters in the same broadcast domain, and using the same HA group ID, you might get MAC address conflicts. For those cases, it is strongly recommended that you assign different HA group IDs to each cluster.

Through the heartbeats, the primary informs all secondary devices about the assigned virtual MAC address. Upon failover, a secondary adopts the same virtual MAC addresses for the equivalent interfaces.

The new primary broadcasts gratuitous ARP packets, notifying the network that each virtual MAC address is now reachable through a different switch port.

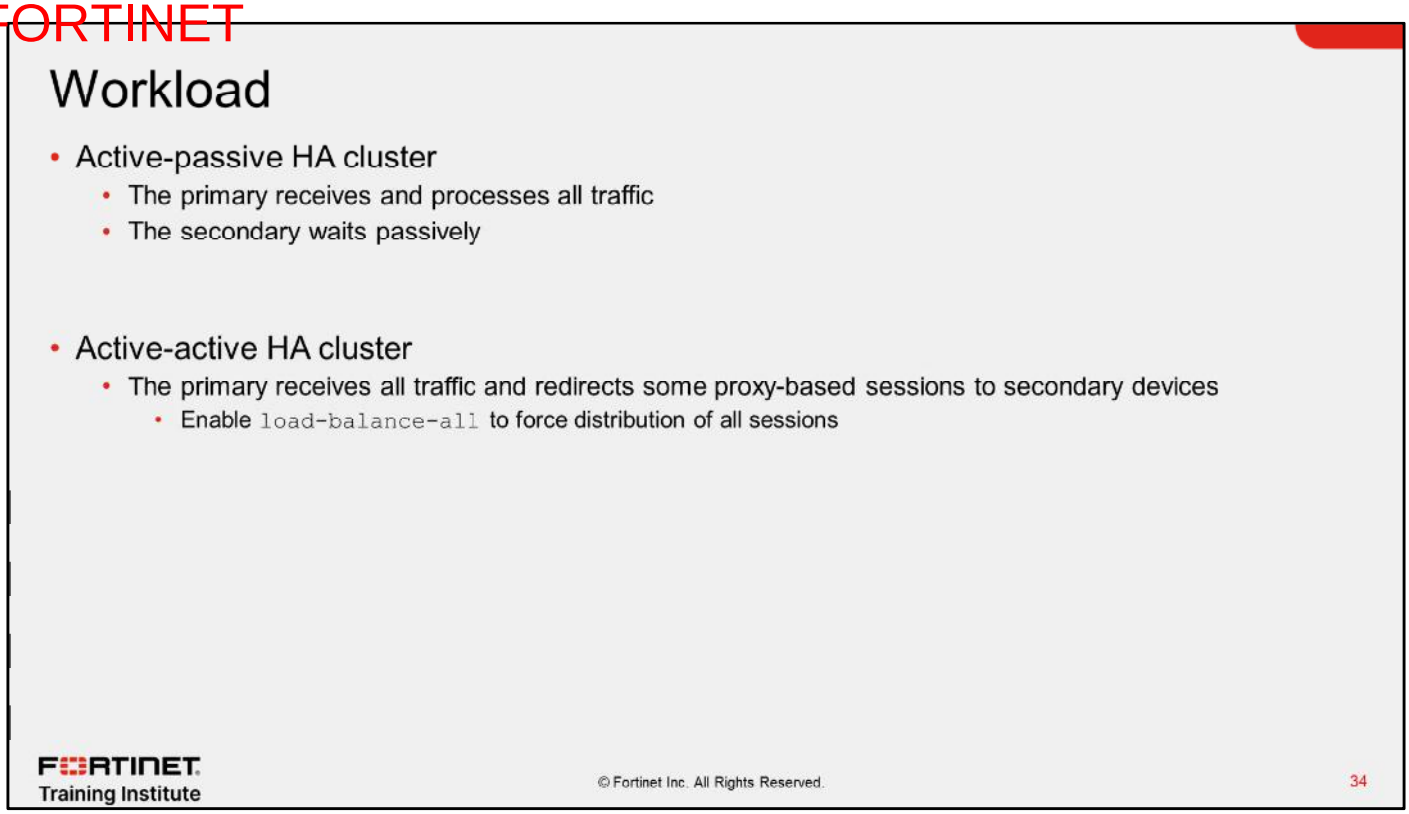
Note that the MAC address of a reserved HA management interface is not changed to a virtual MAC address. Instead, the reserved management interface keeps its original MAC address.



As you learned earlier in this lesson, if a primary fails, a new primary is elected. But what happens if a secondary FortiGate device fails? It depends on the HA mode.

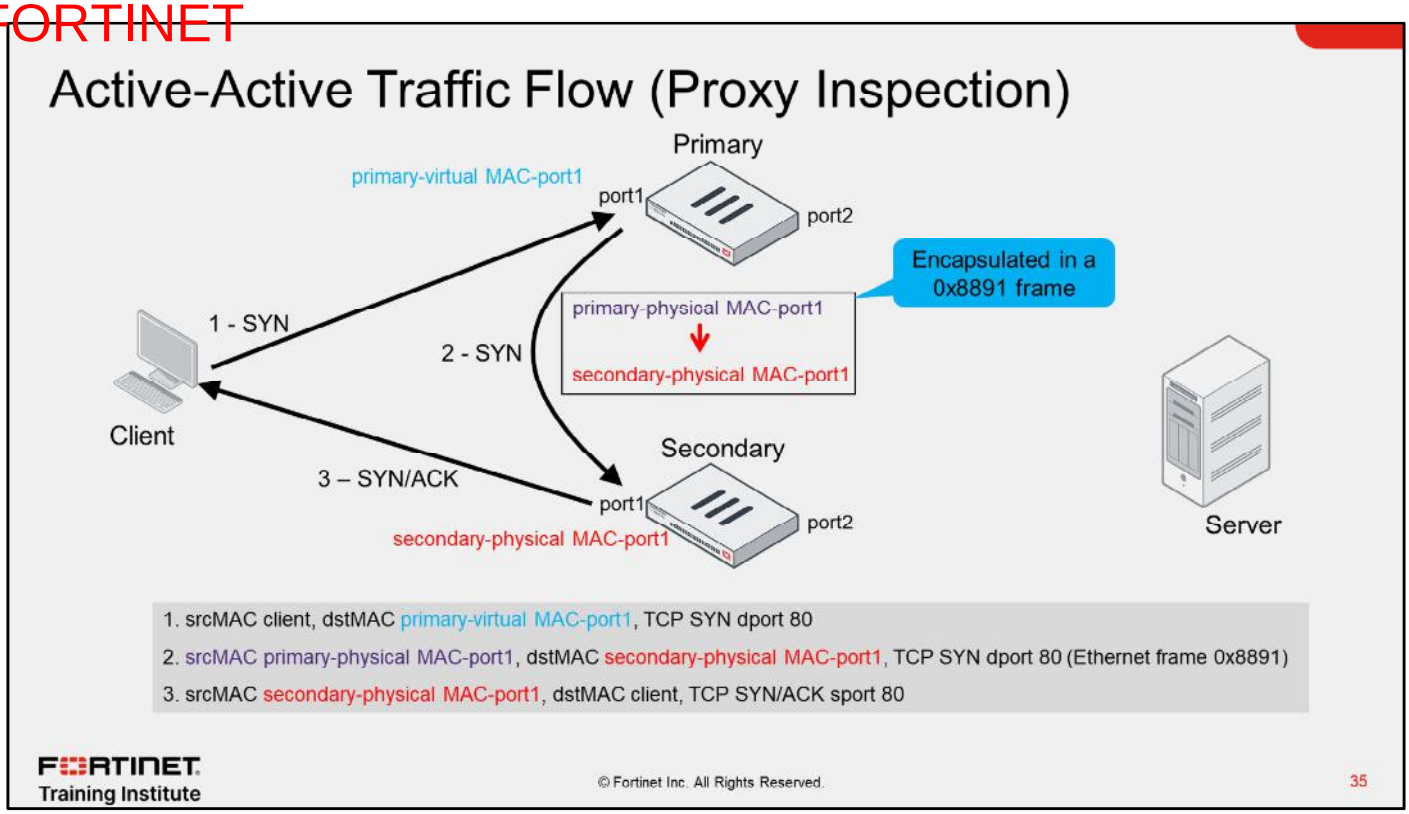
In an active-passive cluster, the primary only updates its list of available secondary FortiGate devices. It also starts monitoring for the failed secondary, waiting for it to come online again.

However, in an active-active cluster, the secondary devices can handle traffic. So, the primary (which tracks and assigns sessions to each secondary) must not only update its list of available secondary FortiGate devices, but also reassign sessions from the failed FortiGate to a different secondary FortiGate.



This is how the workload is distributed between roles, depending on the HA mode.

Notice that traffic workload is distributed in active-active mode only. However, keep in mind that by default, only sessions that are subject to proxy inspection are distributed to secondary devices. If you want to force the distribution of sessions that are subject to flow inspection or no inspection at all, then you must enable the `load-balance-all` setting under HA configuration—this setting is disabled by default.

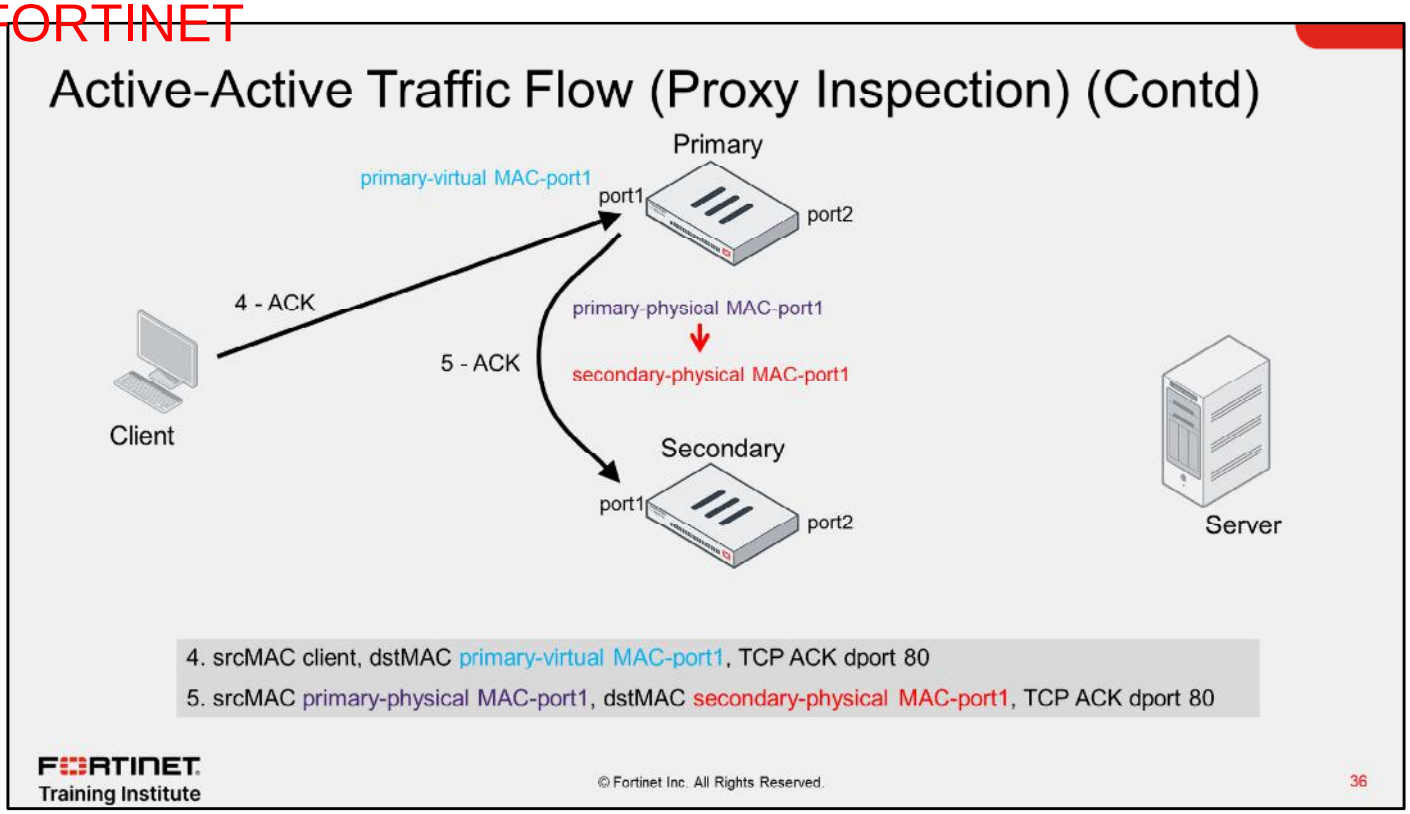


In active-active mode, the following occurs:

- The traffic destined to the cluster is sent to the primary. Because all network ports on the primary—except the heartbeat ports—are assigned a virtual MAC address, the traffic is destined to the virtual MAC address of the receiving port on the primary FortiGate.
- For traffic that is distributed to the secondary, the traffic destined to the endpoints is sent by the secondary. The traffic is sourced from the physical MAC address of the egressing port on the secondary.

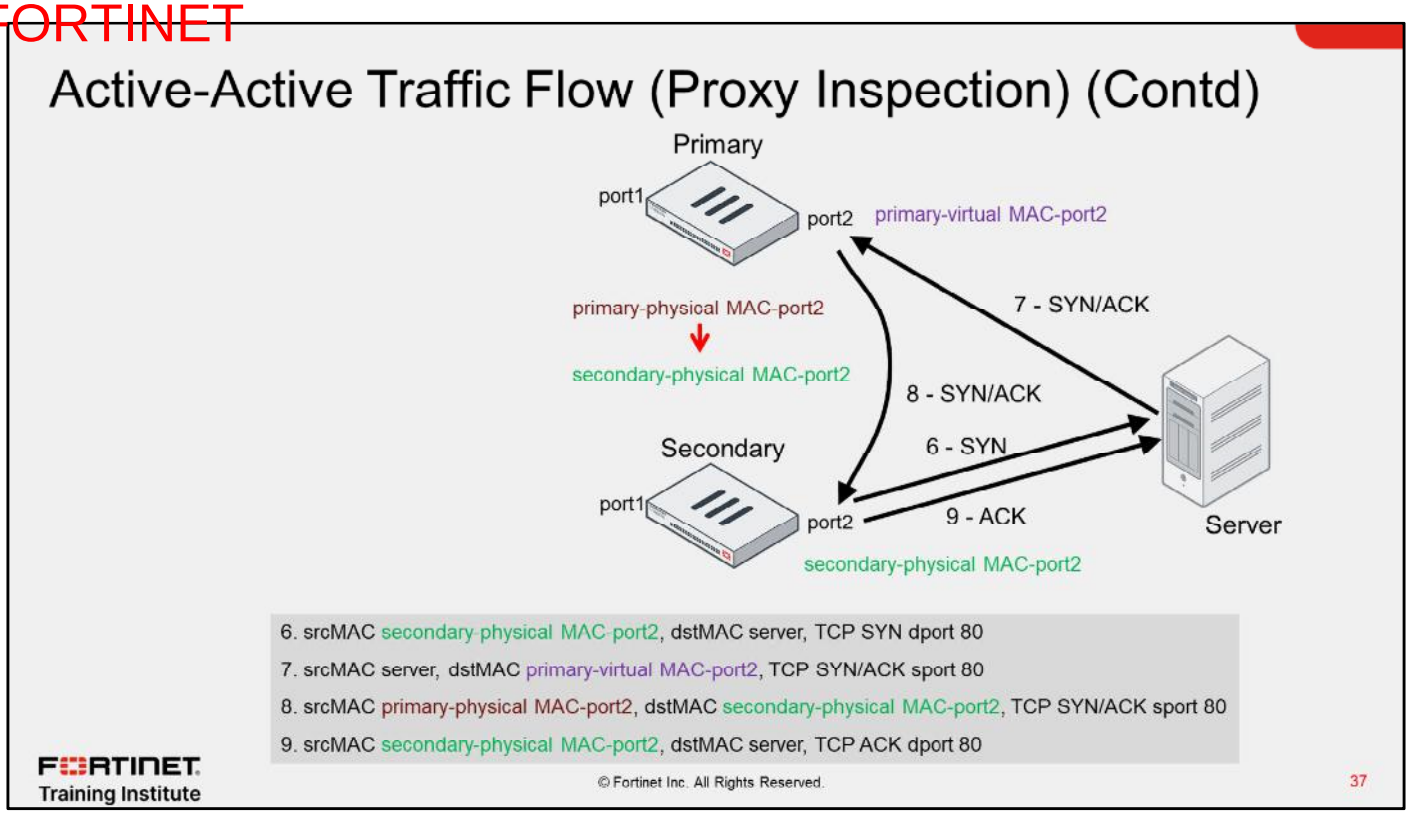
This slide shows the flow for distributed traffic that is subject to proxy inspection:

1. The client sends a SYN packet, which is forwarded to port1 on the primary. The packet destination MAC address is the virtual MAC address on port1.
2. The primary forwards the SYN packet to the selected secondary. In this example, the source MAC address of the packet is changed to the physical MAC address of port1 on the primary and the destination MAC address to the physical MAC address of port1 on the secondary. This is also known as MAC address rewrite. In addition, the primary encapsulates the packet in an Ethernet frame type 0x8891. The encapsulation is done only for the first packet of a load balanced session. The encapsulated packet includes the original packet plus session information that the secondary requires to process the traffic.
3. The secondary responds to the client with a SYN/ACK packet that contains the physical MAC address of port1 on the secondary as the source and the MAC address of the client as the destination.



4. The client acknowledges the SYN/ACK by sending an ACK to the cluster. The ACK packet is destined to port1 on the primary.

5. The primary receives the packet and knows that it matches a session that was previously distributed to the secondary. As a result, the primary forwards the ACK packet to the corresponding secondary FortiGate. The packet is sourced from the physical MAC address of port1 on the primary and destined to the physical MAC address of port1 on the secondary. The three-way handshake on the client side is complete.



6. The secondary starts the connection with the server by sending a SYN packet using the physical MAC address of port2 as the source. Note that FortiGate contacts the server after it finishes the three-way handshake to the client, not before. The same behavior is seen when FortiGate operates in standalone mode and performs proxy-based inspection.

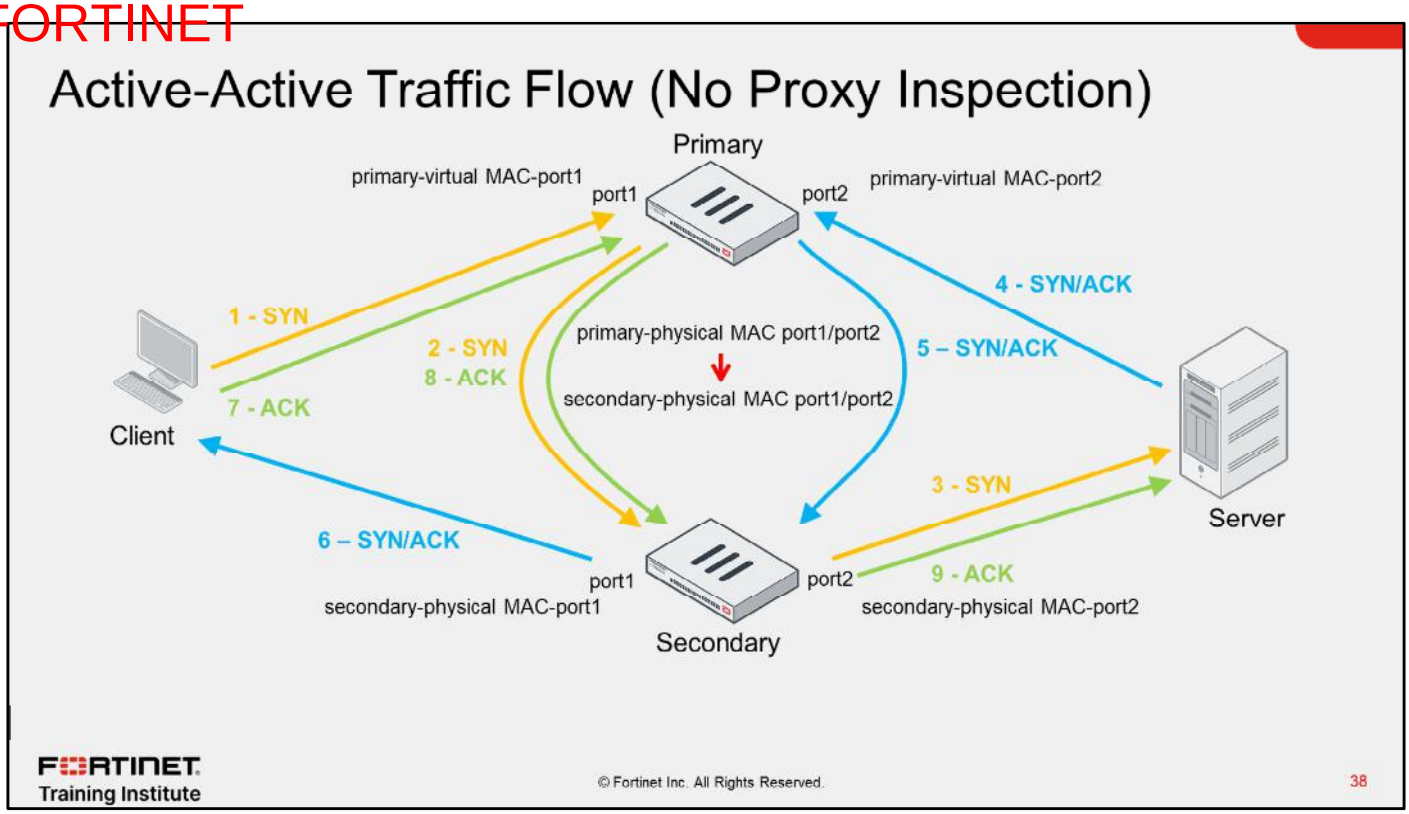
7. The SYN/ACK packet from the server is sent to port2 on the primary. The destination MAC address is the virtual MAC address of port2.

8. The primary receives the packet and knows that it matches a session that was previously distributed to the secondary. The primary forwards the SYN/ACK packet to the corresponding secondary FortiGate. The packet is sourced from the physical MAC address of port2 on the primary and destined to the physical MAC address of port2 on the secondary.

9. The secondary responds to the server with an ACK packet that contains the physical MAC address of port2 on the secondary as the source and the MAC address of the server as the destination.

The three-way handshake on the server side is also complete. From now on, packets that the client sends follow the same flow. For example, an HTTP GET request packet from the client is first received by the primary, which then forwards it to the secondary for proxy-based inspection. If the packet is allowed, the secondary forwards the packet to the server. Any server response packets to the client HTTP GET request are sent to the primary, which then forwards the packets to the secondary for inspection, and so on.

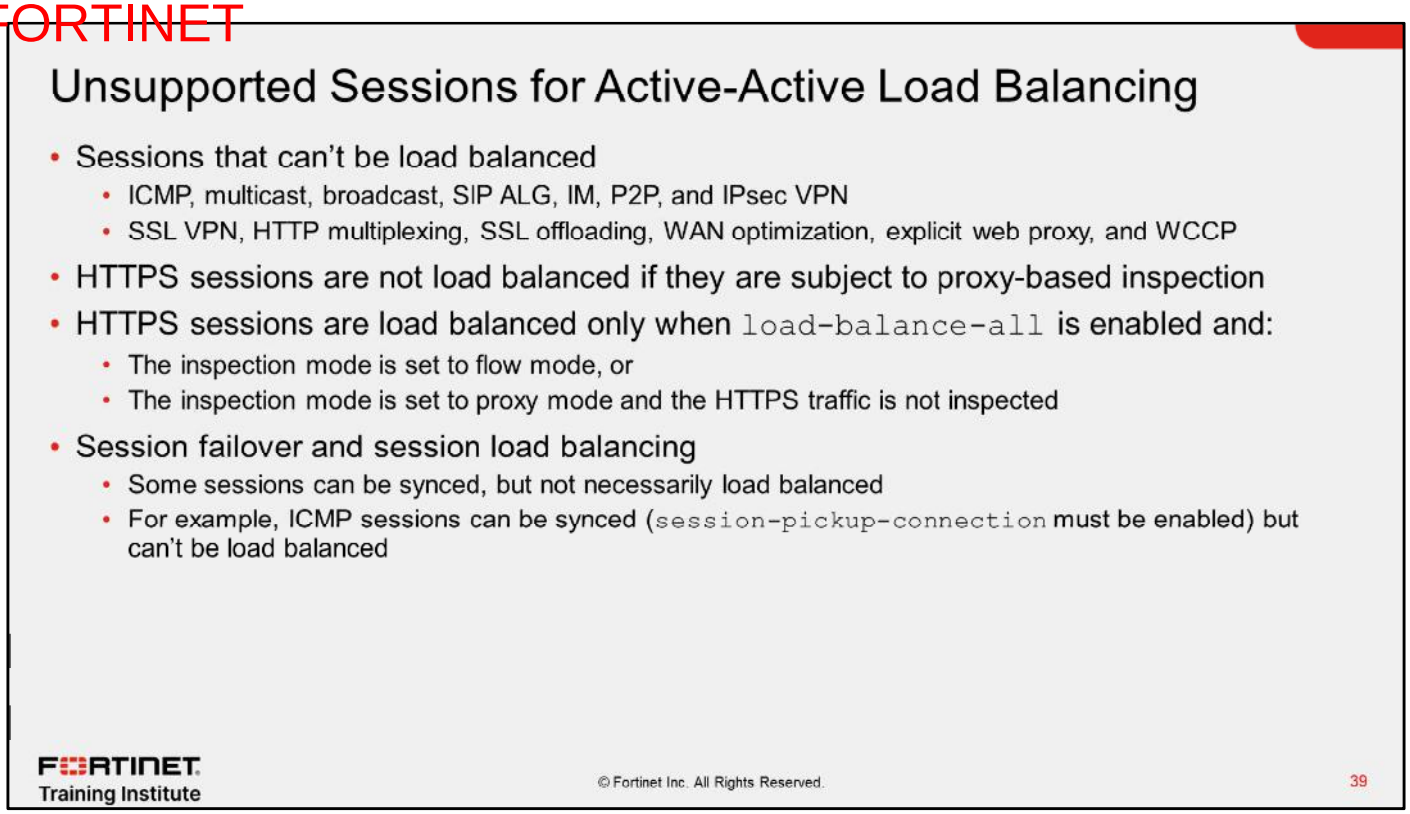
Note that the goal of active-active mode is to leverage unused CPU and memory resources on secondary devices. The intention is not really to load balance traffic. In fact, because the traffic from endpoints is always sent to the primary, you usually see more traffic on the primary than any secondary devices.



When there is no proxy inspection, that is, when traffic is either subject to flow inspection or no inspection at all, sessions are distributed to the secondary FortiGate only if you enable the `load-balance-all` setting (which is disabled by default) under HA configuration. In addition, as in proxy inspection, you will also see the following behavior:

- Traffic sourced from the client or server and destined to the FortiGate cluster is sent to the primary FortiGate. The source and destination MAC addresses are the endpoint (client or server) and the primary FortiGate virtual MAC address, respectively.
- The primary FortiGate may, in turn, forward the traffic to the secondary if the session is to be load balanced.
- When distributing the traffic to the secondary, FortiGate uses the physical MAC addresses of the primary and secondary devices interfaces as the source and destination MAC addresses, respectively.
- If traffic is load balanced to the secondary FortiGate, any traffic sourced from the cluster and destined to the endpoint is sourced from the secondary FortiGate. This means that the source MAC address is the physical address of the secondary egress interface.

When compared to proxy inspection, the difference is that FortiGate does not reply to packets on behalf of the client or server. For example, instead of replying to the SYN packet that the client sends, FortiGate forwards the packet to the server through the secondary. Similarly, FortiGate forwards packets that the server sends to the client through the secondary.



In active-active mode, not all sessions qualify for active-active load balancing. This slide shows a list of sessions that can't be load balanced.

Most of the internet traffic nowadays is HTTPS. For this reason, it is important to understand the limitations for HTTPS traffic load balancing. You must know that HTTPS sessions are not load balanced if they are subject to proxy-based inspection. In fact, the only two scenarios in which HTTPS sessions are load balanced is when the `load-balance-all` setting is enabled and:

- The inspection mode is set to flow mode, or
- The inspection mode is set to proxy mode and the HTTPS traffic is not inspected.

Do not confuse session failover with session load balancing. While some sessions can be synchronized to secondary members for session failover protection, those same sessions aren't necessarily supported for active-active load balancing. For example, ICMP sessions can be synchronized to secondary members if you enable the `session-pickup-connectionless` setting, but they cannot be load balanced.

Active-Active Load Balancing Methods

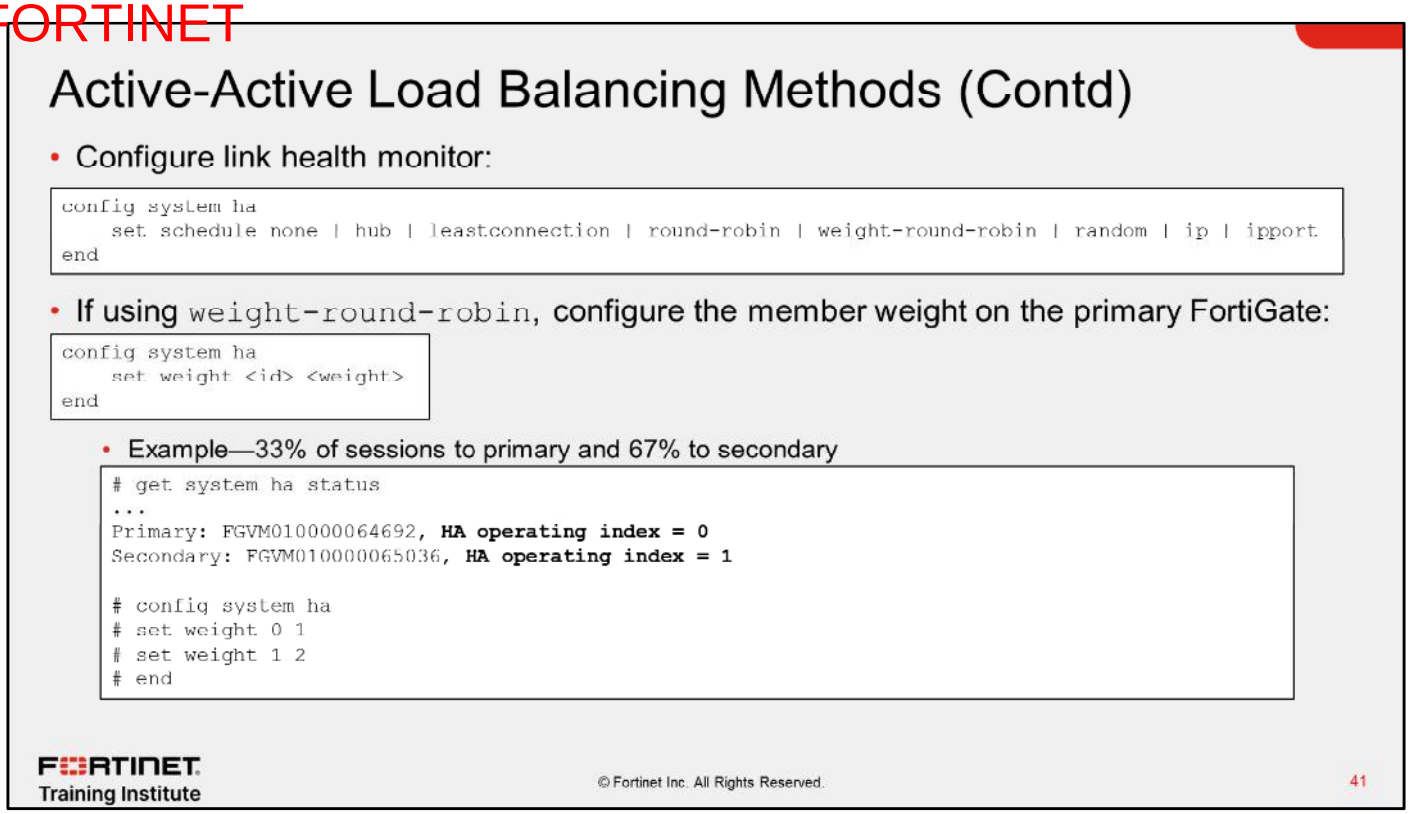
Method	Description
none	The primary handles all sessions
leastconnection	Sessions are sent to the member with the least number of sessions
round-robin	Default method. Sessions are distributed equally across members
weight-round-robin	The more weight a member is assigned, the more sessions it handles
random	Sessions are distributed randomly across members
ip hub	Sessions with the same source and destination IP pair are handled by the same member
ipport	Distribution based on source address, source port, destination address, and destination port information

FORTINET Training Institute

40

In active-active mode, when the primary device distributes sessions, it uses one of the following load balancing methods:

- `none`: Load balancing is turned off. The primary handles all sessions.
- `leastconnection`: The primary distributes sessions to the member with the least number of sessions.
- `round-robin`: This is the default method. The primary distributes sessions equally across members.
- `weight-round-robin`: The primary distributes sessions across members based on the member weight. The higher the member weight, the more sessions are distributed to that member.
- `random`: The primary distributes sessions randomly across members.
- `ip` and `hub`: The primary distributes sessions with the same source and destination IP pair to the same member. Both methods, `ip` and `hub`, work the same way. Both names in the configuration were kept for legacy compatibility purposes. The `hub` schedule will be removed in a future FortiOS version.
- `ipport`: The primary distributes sessions based on the source address, source port, destination address, and destination port information. The more diverse the traffic is, the more evenly the traffic is distributed across members.



You set the load balancing method by configuring the `schedule` setting, as shown on this slide.

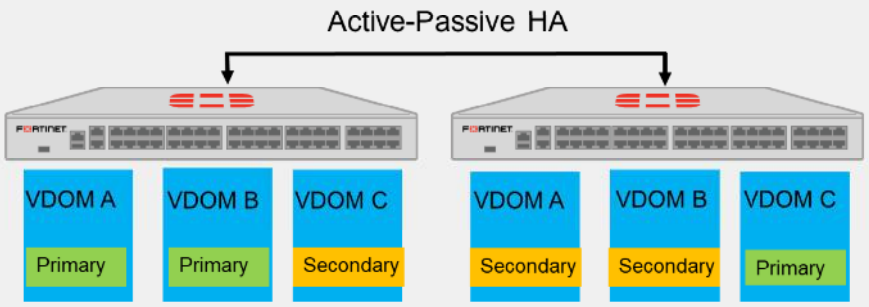
When you select the `weight-round-robin` method, you must also configure the weight for each member, as shown on this slide. You indicate the member ID followed by its weight. The higher the member weight, the more sessions are distributed to that member. You can obtain the member ID from the output of the `get system ha status` command.

This slide also shows a configuration example for a weight-based distribution of 67% of sessions to the secondary FortiGate and 33% of sessions to the primary device. That is, for every three connections that qualify for load balancing, two of them are distributed to the secondary, and one of them to the primary.

Note that you apply the member weight configuration for all members on the primary device. That is, you don't have to apply the weight on each member individually. The cluster will synchronize the configuration to each member for you.

Virtual Clustering

- Virtual clusters are an extension of FGCP for FortiGate with multiple VDOMs
  - The HA cluster *must* consist of *only two* FortiGate devices
- Allows FortiGate to be the primary for some VDOMs and the secondary for the other VDOMs



FORTINET Training Institute

42

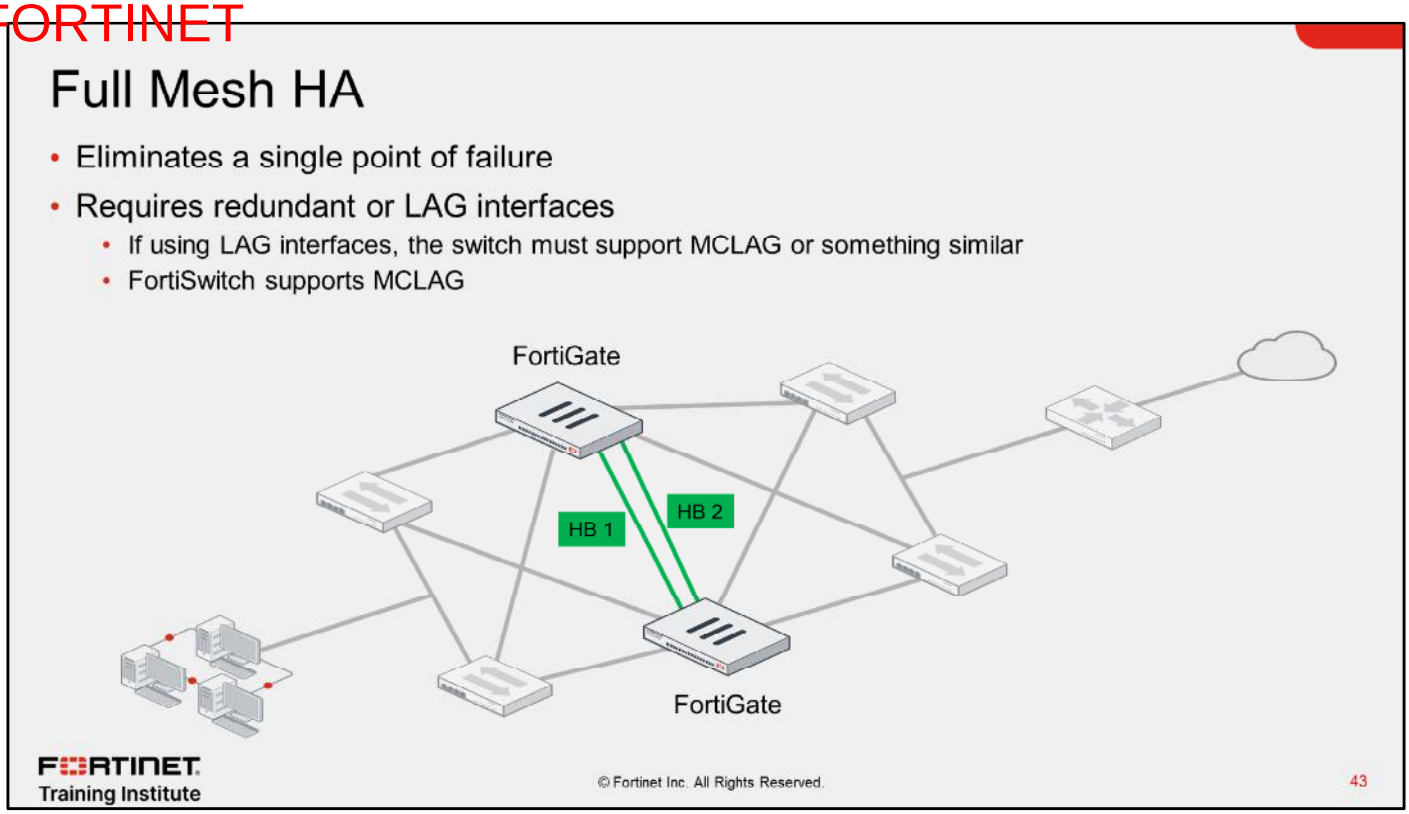
So far, you've learned about HA clustering where each FortiGate device acts as a whole security domain.

But, if you have an HA cluster with multiple VDOMs, you can configure *virtual clusters*.

Virtual clusters allow you to have one device acting as the primary for one VDOM, and as the secondary for a different VDOM. Each VDOM has a primary and a secondary FortiGate. Any device can act as the primary for some VDOMs, and the secondary for the other VDOMs, at the same time. Because traffic from different VDOMs can go to different primary FortiGate devices, you can use virtual clustering to manually distribute your traffic between the two cluster devices and allow the failover mechanism for each VDOM between two FortiGate devices.

Note that if you deploy virtual clustering with more than two FortiGate devices, only two FortiGate devices will process the traffic.
When you add additional (third or fourth) FortiGate devices to a virtual cluster, the primary FortiGate and first secondary FortiGate handle all traffic, and the remaining FortiGate(s) will be operating in standby mode. In the event of a failure of the primary or first secondary FortiGate, one of the remaining FortiGate devices takes over as the new primary or secondary FortiGate and starts handling the traffic.

FGCP in Active-Active mode cannot load balance any sessions that traverse NPU VDOM links or regular VDOM links. If Active-Active session load balancing between VDOMs is required, use an external router to handle the inter-VDOM routing.



At the beginning of this lesson, you reviewed a simple HA topology. Now, take a look at a more robust topology. It is called *full mesh HA*.

The goal of a full mesh HA topology is to eliminate a single point of failure, not only by having multiple FortiGate devices forming a cluster, but also by having redundant links to the adjacent switches. The goal is to have two switches for both upstream and downstream links, and then connect the redundant links to different switches. For example, the topology on this slide shows two FortiGate devices forming a cluster, and each FortiGate is connected to two redundant switches, using two different interfaces.

To achieve redundancy with adjacent switches, you must deploy redundant or LAG interfaces. If you use redundant interfaces, only one interface remains active. This avoids a Layer 2 loop and a standard switch should suffice. However, if you want to use LAG interfaces, then you must ensure that the switch supports multichassis link aggregation group (MCLAG) or a similar virtual LAG technology that enables you to form a LAG whose interface members connect to different switches. FortiSwitch, which is a Fortinet Ethernet switch, supports MCLAG. You can use FortiSwitch as the adjacent switch to deploy a full mesh HA topology with FortiGate.

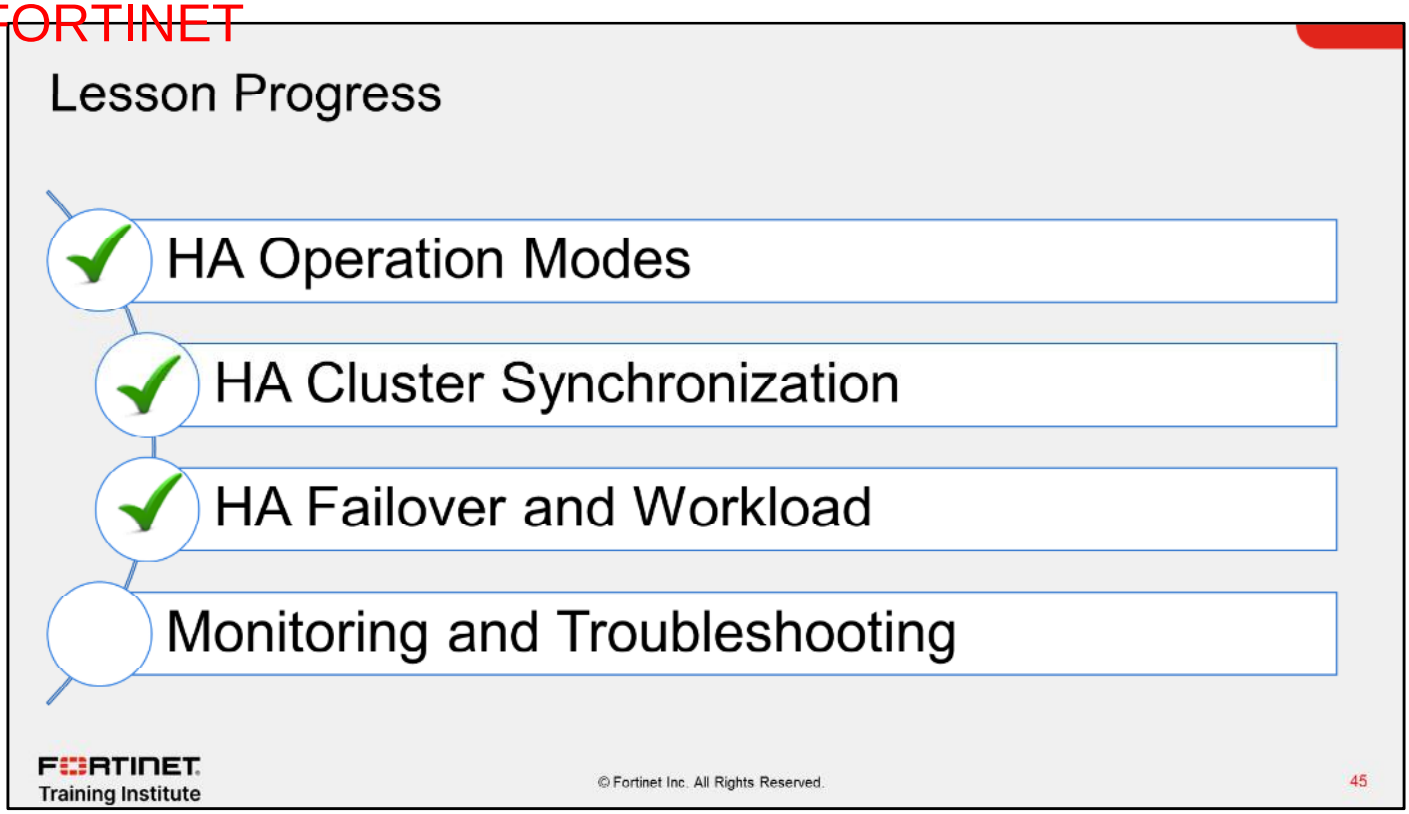
Knowledge Check

1. An HA failover occurs when the link status of a monitored interface on the _____ goes down.
   - ✓ A. Primary FortiGate
   - B. Secondary FortiGate

2. In an active-passive HA cluster, you can configure virtual clustering between only ___ FortiGate devices with multiple VDOMs.
   - ✓ A. Two
   - B. Four

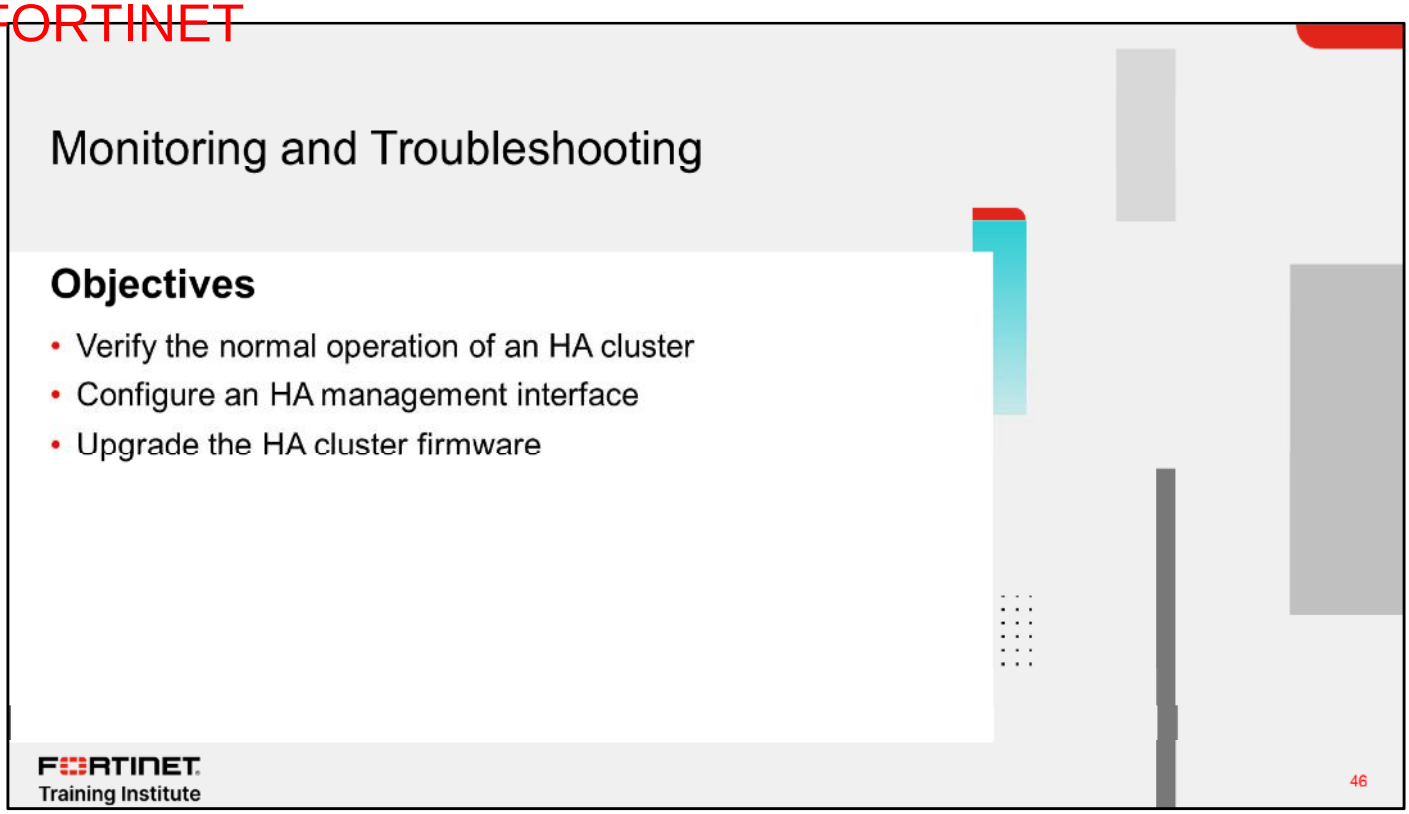
FORTINET Training Institute

44



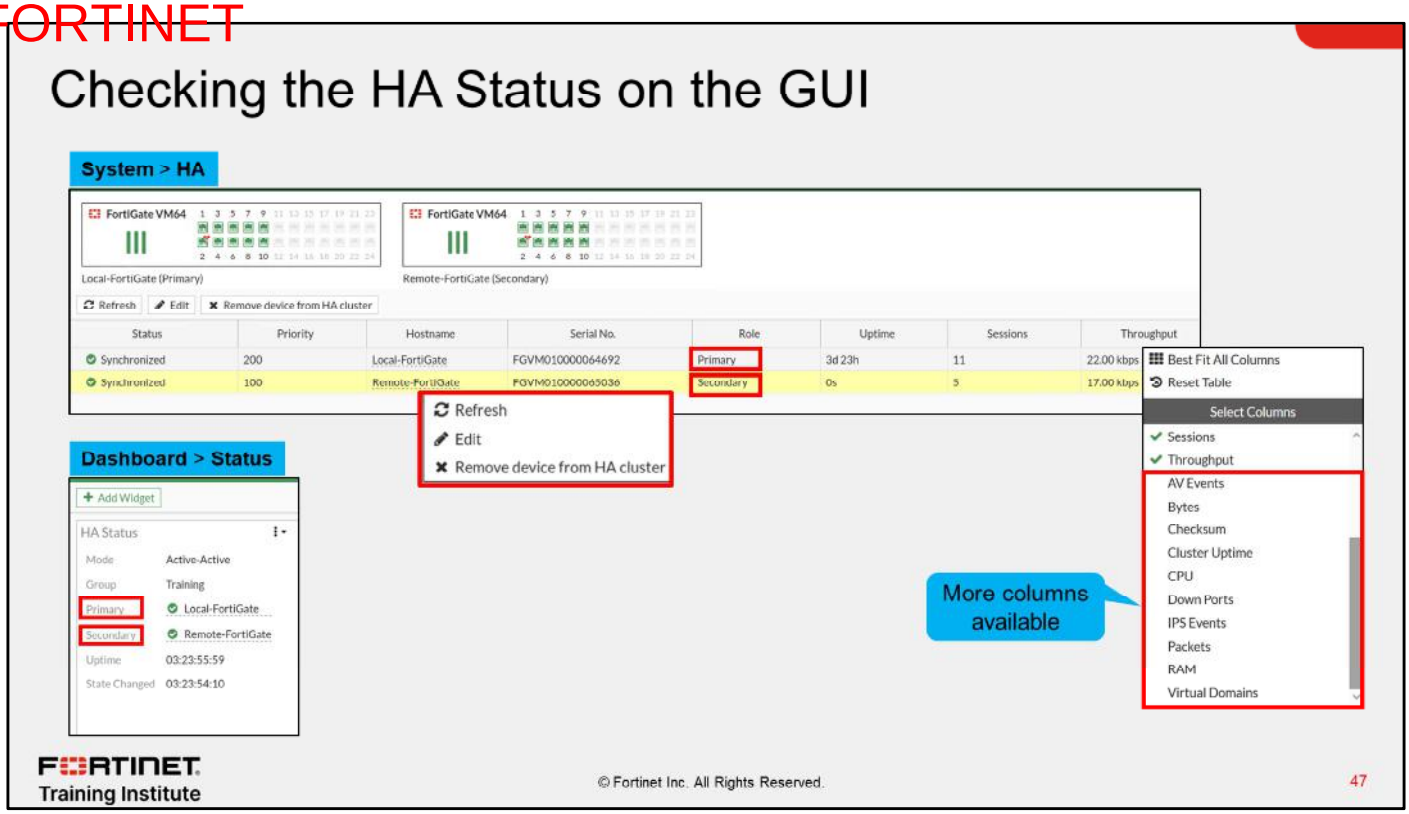
Good job! You now understand HA failover and workload.

Now, you will learn about monitoring and troubleshooting an HA cluster.



After completing this section, you should be able to achieve the objectives shown on this slide.

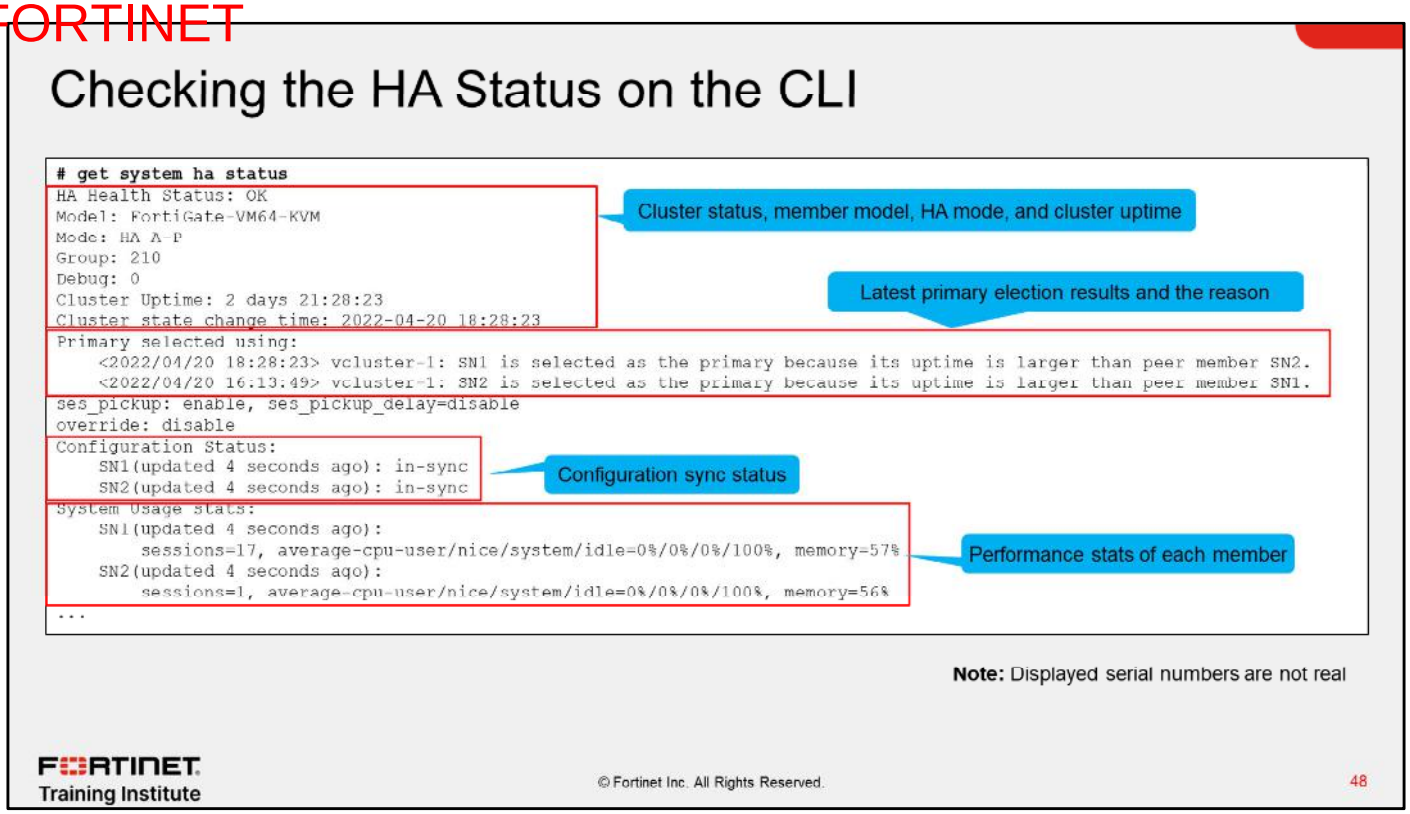
By demonstrating competence in monitoring and troubleshooting, you will be able to make sure the cluster is synchronized properly. You will also learn how to configure and access secondary devices in an HA cluster and how to upgrade the firmware on the HA cluster.



The **HA** page on the FortiGate GUI shows important information about the health of your HA cluster. For each cluster member, the page shows whether the member is synchronized or not, and its status, host name, serial number, role, priority, uptime, and active sessions.

On the **HA** page, you can remove a device from a cluster. When you remove a device from HA, the device operation mode is set to standalone. You can also enable more columns that display other important information about each member such as the checksum, CPU, and memory.

You can also add the **HA Status** widget on the **Dashboard** page. The widget provides a summary of the HA status on the device.



You can get more information about the HA status on the FortiGate CLI by using the `get system ha status` command.

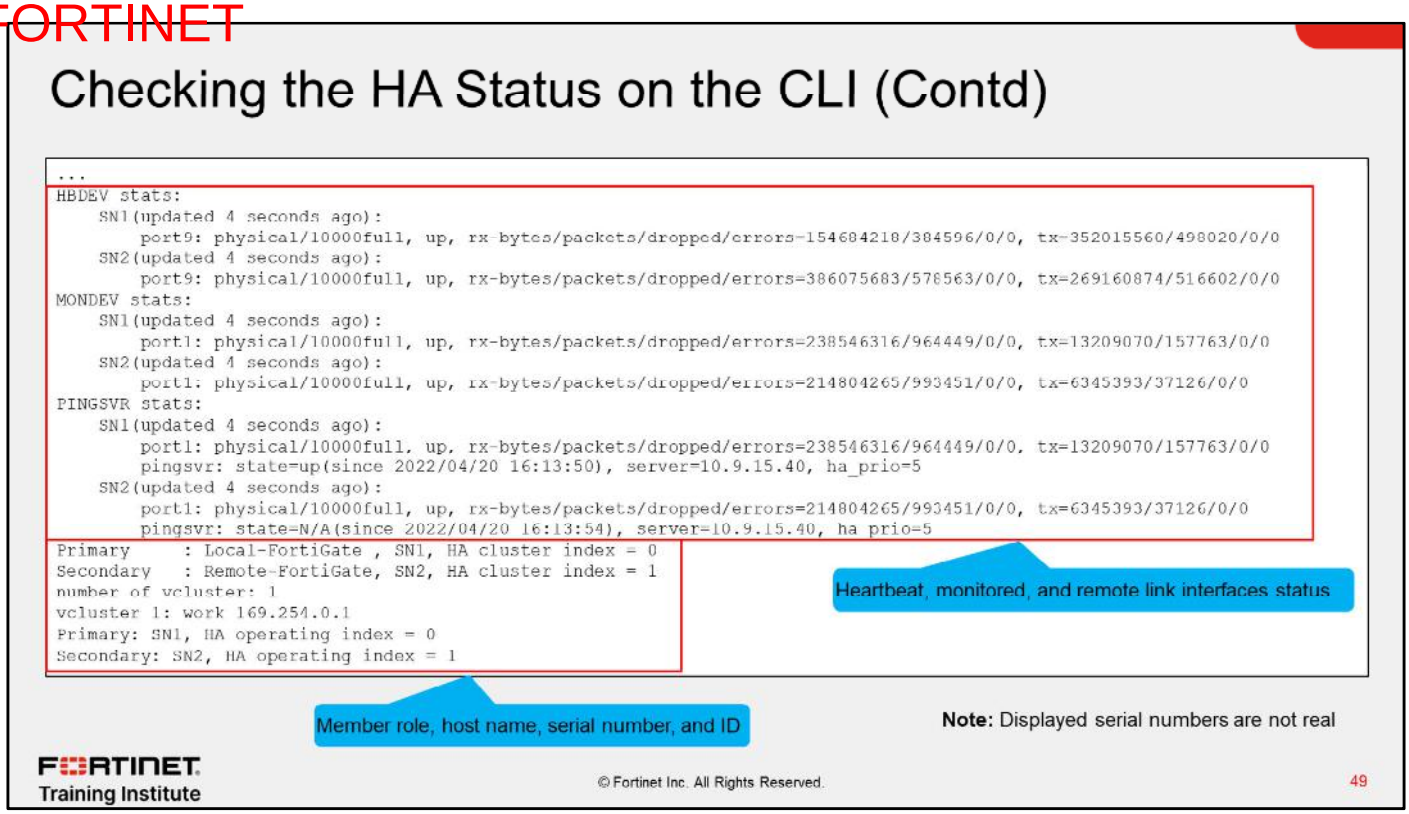
The command displays comprehensive HA status information in a user-friendly output and is usually executed as the first step when troubleshooting HA. This slide shows the first part of an example output that the command provides. Note that the serial numbers of members have been replaced by fake ones (SN1 and SN2), so the output fits on this slide.

At the beginning of the output, you can see the cluster status, the member model, the HA mode in use, and the cluster uptime. The example output shows that the cluster status is good, the member model is FortiGate-VM64-KVM, and the HA mode is active-passive. The cluster has also been up for almost three days.

Next, you can see the latest primary election events, the result, and the reason. The output indicates that a different member was elected as the primary during the last two election events. In both cases, the member was elected because it had a higher HA uptime.

The configuration status information is displayed next. It indicates the configuration sync status for each member. For both members, the configuration is in sync.

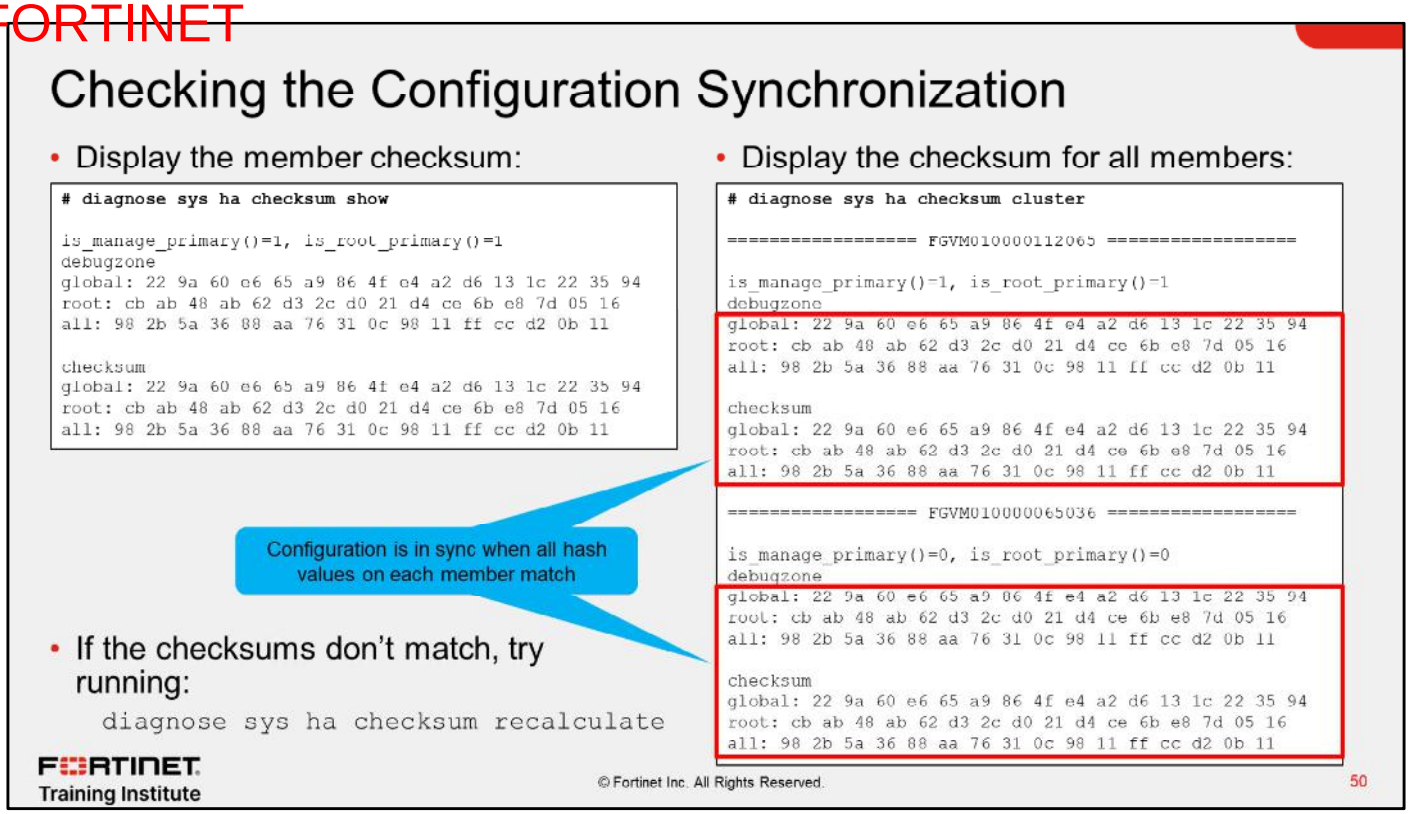
Following the configuration status information, you can see the system usage statistics, which report on performance statistics for each member. They indicate the number of sessions that each member handles, as well as the average CPU and memory usage. Note that the `sessions` field accounts for any sessions that the member handles, and not only the sessions that are distributed when the HA mode is active-active.



This slide shows the second part of the example output that the `diagnose system ha status` command provides.

The output begins with the status information for the configured heartbeat, monitored, and remote link interfaces. These interfaces enable the cluster to perform device failover, link failover, and remote link failover protection, respectively.

Next, the output shows the role, host name, serial number, and ID information for each member of the cluster. The output indicates that the `Local-FortiGate` and `Remote-FortiGate` devices are primary and secondary members, respectively.



The `diagnose sys ha checksum` command tree enables you to check the cluster configuration sync status. In most cases, you want to use the `diagnose sys ha checksum cluster` command to view the cluster checksum. The output includes the checksum of each member in the cluster.

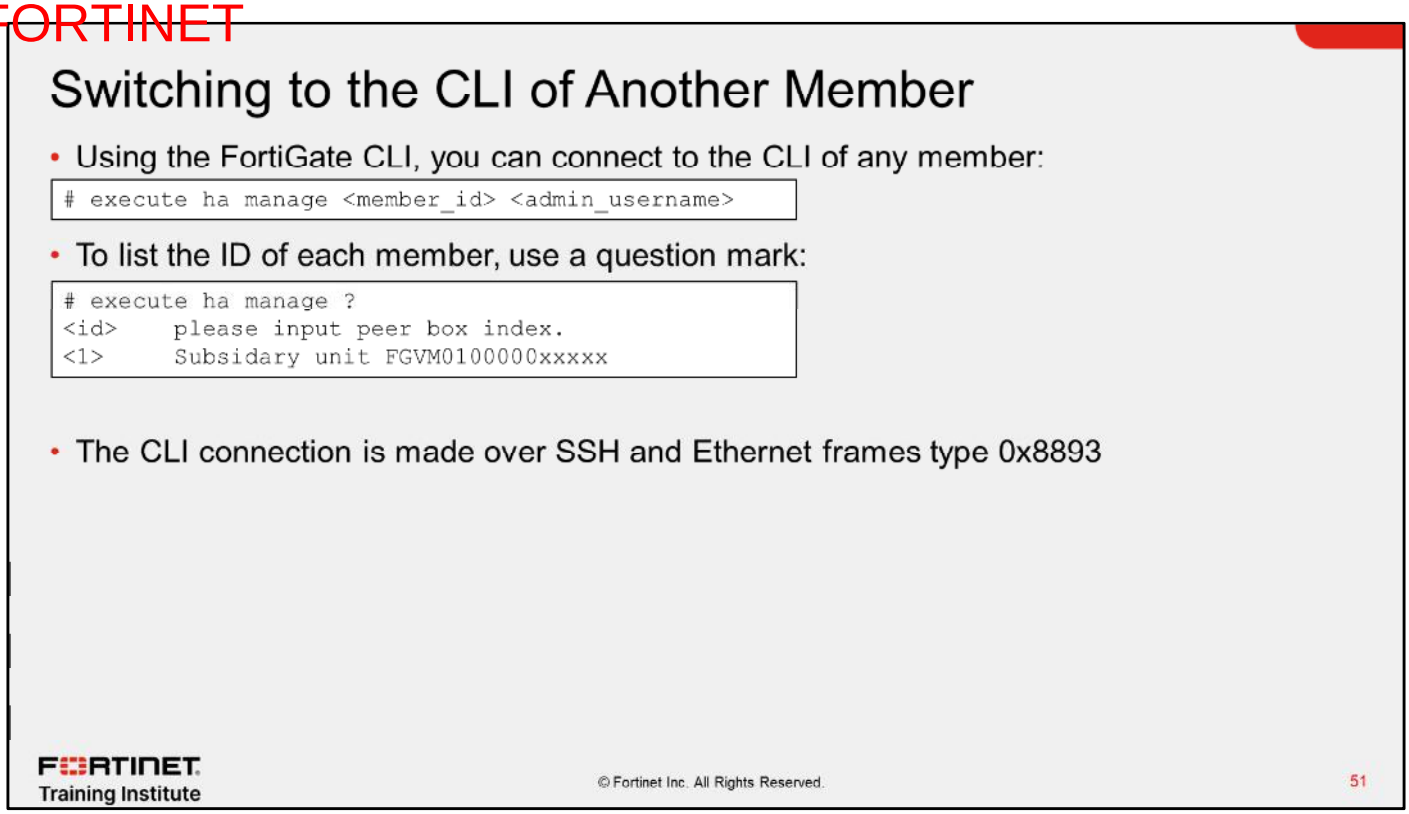
When you run the `diagnose sys ha checksum cluster` command, the checksum is polled from each member using the heartbeat interface. If HA is not working properly, of if there are heartbeat communication issues, then the command may not show the checksum for members other than the one you run the command on. An alternative is to connect to each member individually and run the `diagnose sys ha checksum show` command instead. This command displays only the checksum of the member you are connected to.

After you obtain the checksums of each member, you can identify the configuration sync status by comparing the checksums. If all members show the exact hash values for each configuration scope, then the configuration of all members is in sync.

To calculate checksums, FortiGate computes a hash value for each of the following configuration scopes:

- `global`: global configuration, such as global settings, FortiGuard settings, and so on
- `root`: settings and objects specific to the root VDOM—if you configure multiple VDOMs, FortiGate computes hash values for each VDOM
- `all`: global configuration plus the configuration of all VDOMs

In some cases, the configuration of members is in sync even though the checksums are different. For these cases, try running the `diagnose sys ha checksum recalculate` command to recalculate the HA checksums.

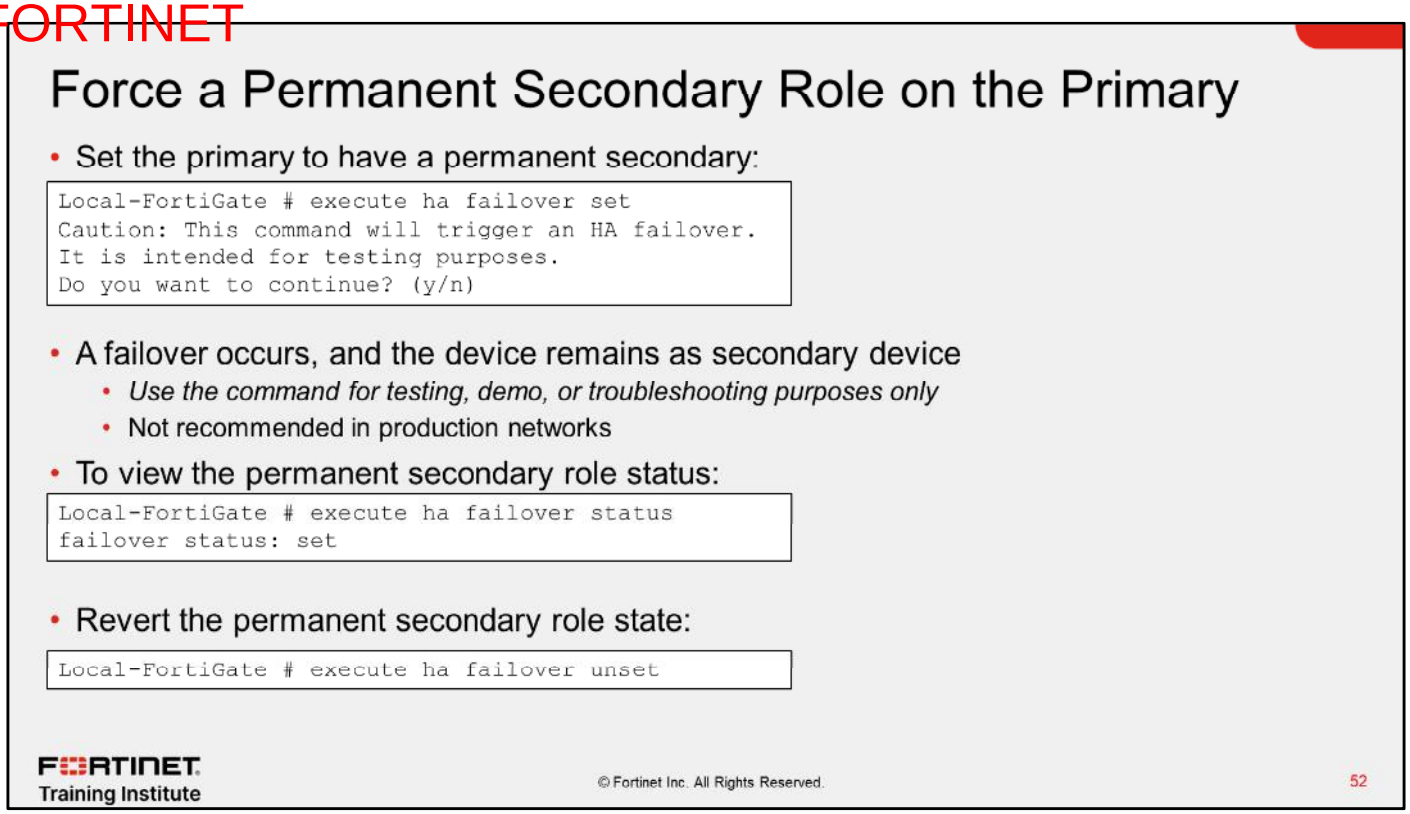


When troubleshooting HA, you may need to connect to the CLI of another member from the CLI of the member you are currently connected to. You do this by using the `execute ha manage` command to connect to the other member.

For example, when you connect to the cluster over SSH using any of the cluster virtual IP addresses, you connect to the primary member. If you then want to connect to another member, you can use the `execute ha manage` command to access its CLI.

This command requires you to indicate the ID of the member you want to connect to and the username you will use to log in. To get the list of member IDs, you can add a question mark to the end of the `execute ha manage` command, as shown on this slide.

Note that when you switch to the CLI of another member, FortiGate establishes an SSH session to that member over the heartbeat interface. The SSH session is then encapsulated in Ethernet frames type 0x8893.



You can set the primary FortiGate to have a permanent secondary role using the `execute ha failover set` command. When you do this, a failover occurs, and the former primary member remains as a secondary member permanently, regardless of the status of other members in the cluster. That is, the impacted member never takes over the cluster even if it's the best candidate for the primary role.

You can revert the permanent secondary role state by running the `execute ha failover unset` command. Note that you should set the primary member to a permanent secondary role for testing, troubleshooting, and demonstration purposes only. Do not use this feature in production networks.

Connect to Any Member Directly

- Reserved HA management interface
  - Out-of-band
  - Up to four dedicated interfaces
  - For local-in traffic and *some* local-out traffic
  - Separate routing table
  - Configuration example (not synchronized):

```
config system ha
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "port10"
            set gateway 192.168.100.254
        next
    end
end
config system interface
    edit "port10"
        set ip 192.168.100.1 255.255.255.0
        set allowaccess ping https ssh snmp
    next
end
```

- In-band HA management interface
  - In-band
  - Use any user-traffic interface
  - For local-in and local-out traffic
  - Shared routing table
  - Configuration example (not synchronized):

```
config system interface
    edit "port1"
        set management-ip 10.0.10.1 255.0.0.0
        set allowaccess ping https ssh snmp
    next
end
```

FORTINET Training Institute

53

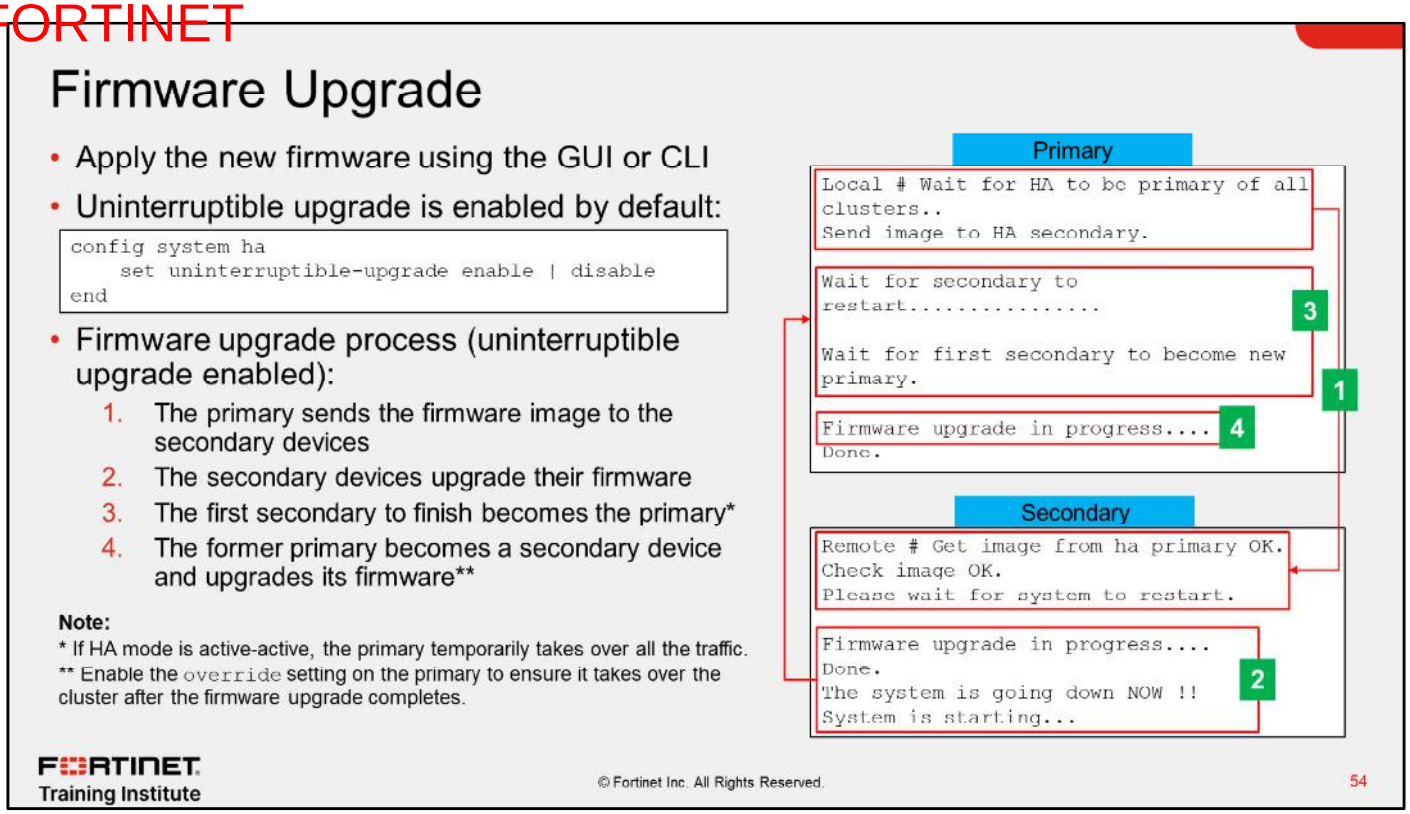
When you connect to a cluster using any of its virtual IP addresses, you always connect to the primary. You can then switch to the CLI of any member in the cluster by using the `execute ha manage` command. But what if you want to access the GUI of a secondary member or maybe poll data from it using SNMP? For this, you need a way to access each member directly regardless of its role in the cluster.

FortiGate provides two ways for the administrator to connect to a member directly no matter what the member role is. The reserved HA management interface is the out-of-band option. You configure up to four dedicated management interfaces, and you assign them a unique address on each member. You can then use the unique address assigned to each member to connect to them directly. You can also instruct FortiGate to use the dedicated management interface for some outbound management services such as SNMP traps, logs, and authentication requests.

Alternatively, you can configure in-band HA management, which enables you to assign a unique management address to a member without having to set aside an interface for that purpose. You assign the management address to any user-traffic that the member uses, and then connect to the member using that unique management address.

If you have unused interfaces, then it's generally more convenient to use a reserved HA management interface because the user and management traffic don't have to compete. Many FortiGate models come with a management interface that you can use for this purpose. Also, the routing information for a reserved HA management interface is placed in a separate routing table, which means that you don't see the interface routes in the FortiGate routing table. This allows for segmentation between data and management traffic.

This slide also shows configuration examples for both management options. For both options, the configuration you apply on a member is not synchronized to other members in the cluster.



You upgrade an HA cluster in the same way you do for standalone FortiGate devices. That is, you can apply the new firmware using the GUI firmware upgrade tool. In HA, this usually means connecting to the primary FortiGate GUI to apply the new firmware. You can also use the CLI if you prefer.

Also, like on standalone FortiGate devices, the device must reboot to apply the new firmware. However, by default, members in a cluster are upgraded one at a time to minimize service disruption. This feature is called uninterruptible upgrade and is enabled by default. After the administrator applies the new firmware on the primary, uninterruptible upgrade works as follows:

1. The primary sends the firmware to all secondary members using the heartbeat interface.
2. The secondary devices upgrade their firmware first. If the cluster is operating in active-active mode, the primary temporarily takes over all traffic.
3. The first secondary that finishes upgrading its firmware takes over the cluster.
4. The former primary becomes a secondary device and upgrades its firmware next.

Note that depending on the HA settings and uptime, the original primary may remain as a secondary after the upgrade. Later, if required, you can issue a manual failover. Alternatively, you can enable the `override` setting on the primary FortiGate to ensure it takes over the cluster again after it upgrades its firmware, as long as the device is assigned the higher priority.

If you want the cluster to upgrade all members at the same time to speed up the firmware upgrade process, you can disable uninterruptable upgrade, as shown on this slide. Just keep in mind this will result in a service impact during the firmware upgrade.

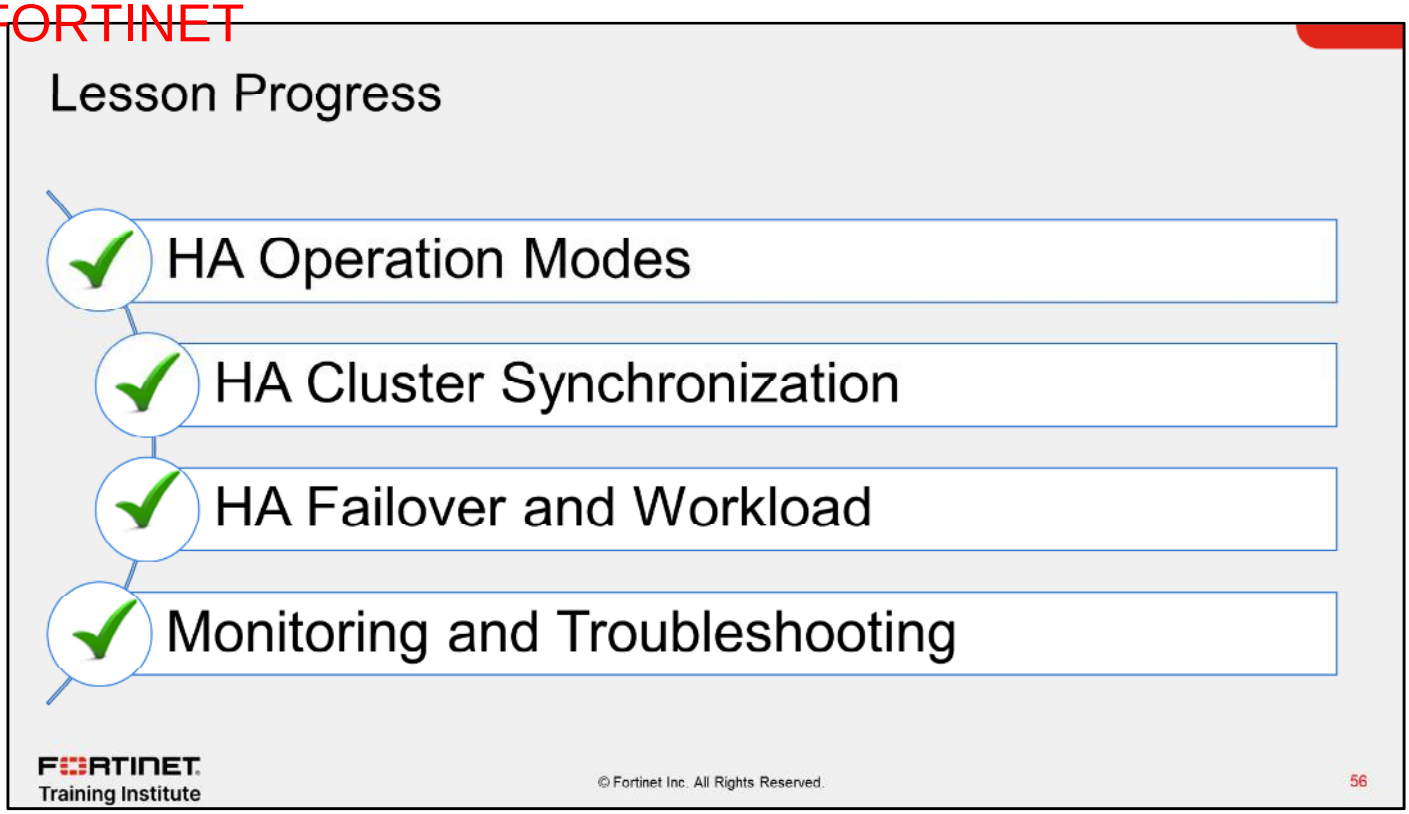
Knowledge Check

1. Which member is the heartbeat interface IP address `169.254.0.1` assigned to?
   - ✓ A. The member with the highest serial number
   - B. The member with the highest priority

2. Which statement about the firmware upgrade process on an HA cluster is true?
   - ✓ A. You upload the new firmware to the primary FortiGate only.
   - B. The members do not reboot.

FORTINET Training Institute

55



Congratulations! You have completed this lesson.

Now, you will review the objectives that you covered in this lesson.