


In this lesson, we are going to talk about how to route traffic with FortiGate devices.

Objectives

- Interpret the routing table
- Implement routing failover by using link health monitors
- Balance the traffic among multiple links with WAN link load balancing and equal cost multipath
- Increase the bandwidth and reliability between two devices by using link aggregation
- Block traffic spoofing your IP addresses with reverse path forwarding check
- Override static routes with policy routes
- Fix routing loop issues with black hole routes
- Diagnose and correct routing issues

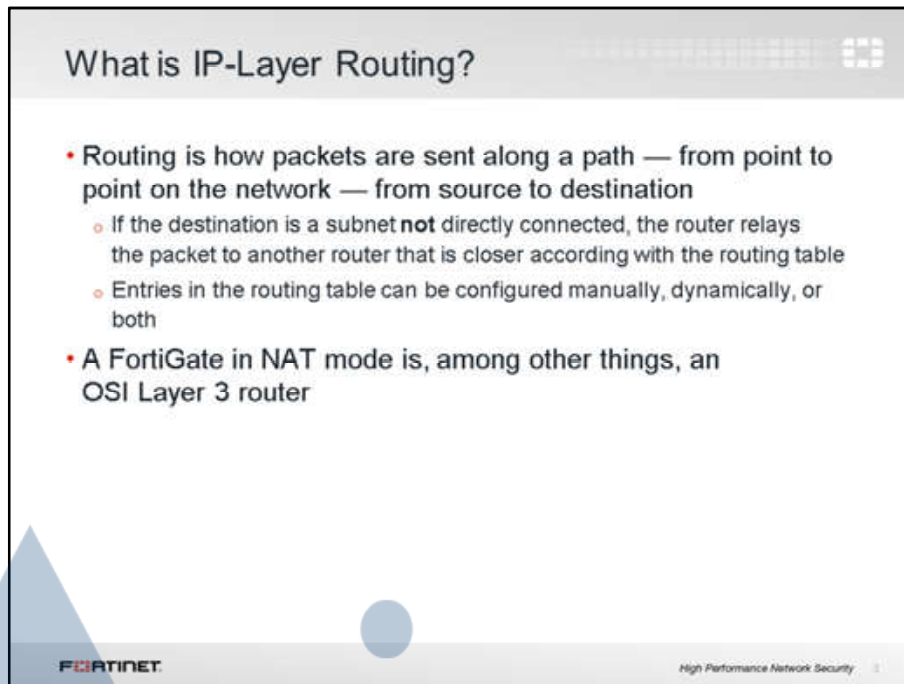


FORTINET

High Performance Network Security

After completing this lesson, you should have these practical skills that you can use to implement routing failover and load balancing using static routes. You will also learn how to configure link aggregation, policy routes, and black hole routes. Finally, you will learn some debug commands for troubleshooting routing problems. Although this lesson briefly introduces the concept of dynamic routing, it is mostly about implementing routing with static and policy-based routes.

Lab exercises can help you to test and reinforce your skills.



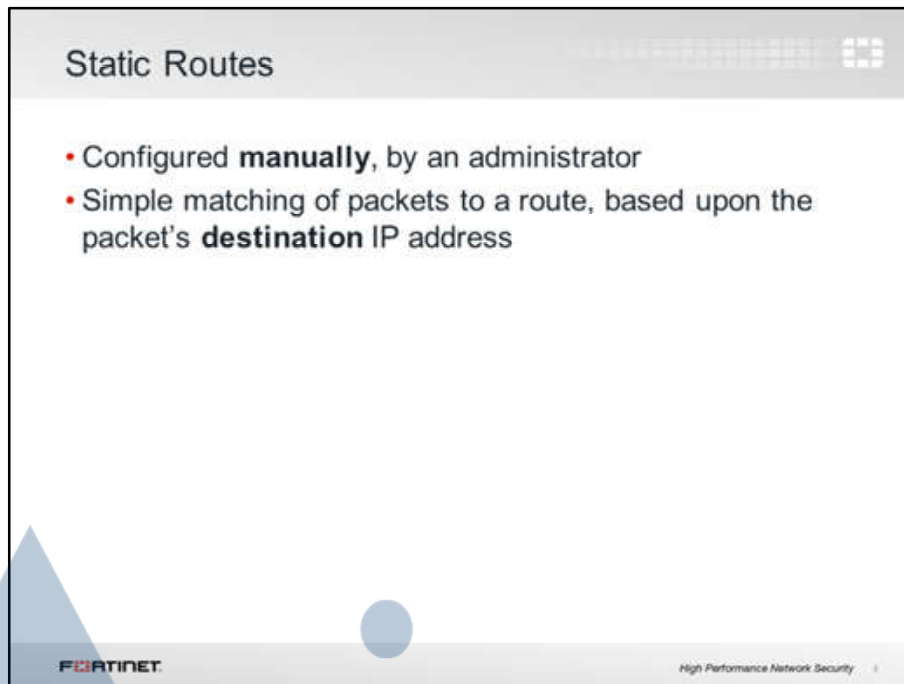
What is routing?

Routing decides where FortiGate in NAT mode will send the packets that it receives, and that it generates. A routing table contains routing rules. For example, FortiGate can check the destination field of the packet's IP header. If routing rules match that destination, FortiGate can transmit the packet from port1 to port2, towards Router 1.

If an allowed packet is not destined for the FortiGate itself — not administrative access, for example — FortiGate must relay the packet. FortiGate searches for matching routes in the routing table that it can use to deliver the packet. FortiGate either delivers the packet directly to its final destination, or relays it to the next router along the path towards the destination.

Usually, IP routing is done by taking into account only the destination IP address. However, as we'll see later, you can decide to route packets using more than just that.

Proper routing configuration is important. If the routing directions are misconfigured, packets will not reach their destination and will be lost.



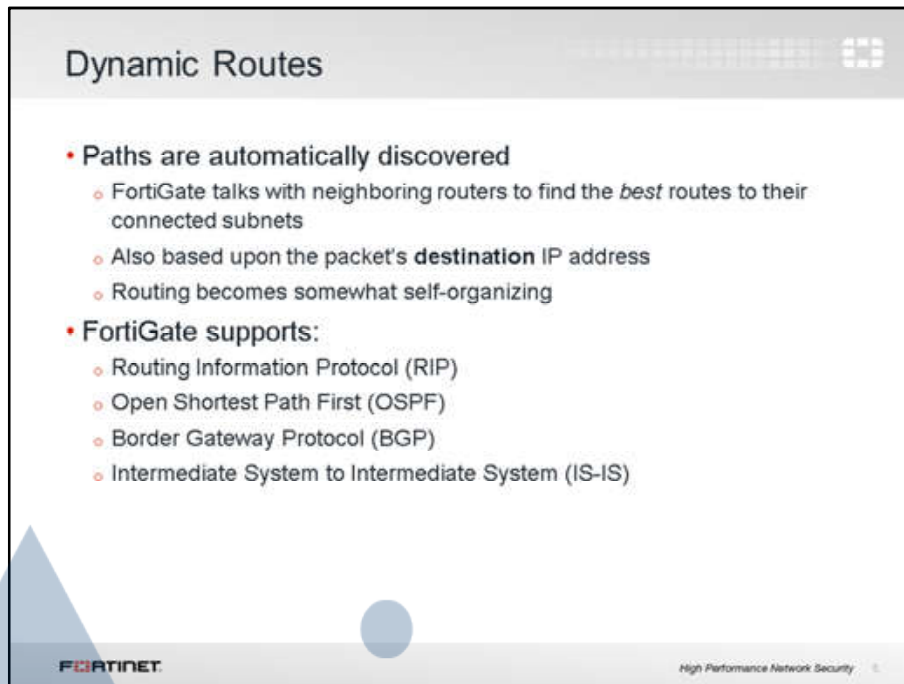
One type of manually configured route is called a static route. In the route table, its “Type” column is “*Static*”.

We are manually telling the FortiGate device, “When you see a packet whose destination is within this specific range of destination addresses, send it through this network interface, towards this router.” We also configure the distance and priority so that FortiGate knows which routes to load into memory, and in what order. We will talk about distance and priority in later slides.

For example, in simple home networks, DHCP automatically retrieves and configures one static route. Your modem then sends all outgoing traffic through your ISP’s Internet router, which can relay packets to their destination.

When do you not require a static route?

When a destination is cabled directly to one of FortiGate’s network interfaces, with no router in between, FortiGate will be aware of the destination. In the route table, its “Type” is “*Connected*”.



For large networks, manually configuring hundreds static routes may not be practical.

Your FortiGate can help, by configuring routes automatically. FortiGate supports several dynamic routing protocols: RIP, OSPF, BGP, and IS-IS.

In dynamic routing, FortiGate communicates with nearby routers to discover their paths, and to advertise its own directly connected subnets. Discovered paths are automatically added to FortiGate's routing table. (So verify that your neighbor routers are trusted and secured!)

Larger networks also may need to balance routing load among multiple valid paths, and detect and avoid routers that are down. We'll discuss that soon also.

FortiGate Routing Table

• Only currently active, best static & dynamic routes

Type	Network	Gateway	Interface	Up Time	Distance	Metric
Static	0.0.0.0/0	172.20.181.177	port1		10	0
Connected	10.0.0.0/24	0.0.0.0	port2		0	0
Static	10.0.1.0/24	10.0.0.2	port2		10	0
Static	10.0.10.0/24	10.0.0.2	port2		10	0
BGP	10.13.0.0/24	10.0.0.2	port2	0 00:02:13	20	0
OSPF	10.13.1.0/24	172.20.181.153	port1	0 16:29:50	110	2
OSPF	10.14.1.0/24	172.20.181.153	port1	0 16:29:50	110	2
OSPF	10.20.1.0/24	172.20.181.153	port1	0 16:29:50	110	2
Static	172.18.0.0/16	10.0.0.2	port2		10	0
Connected	172.20.181.0/24	0.0.0.0	port1		0	0

• No policy routes

FORTINET High Performance Network Security

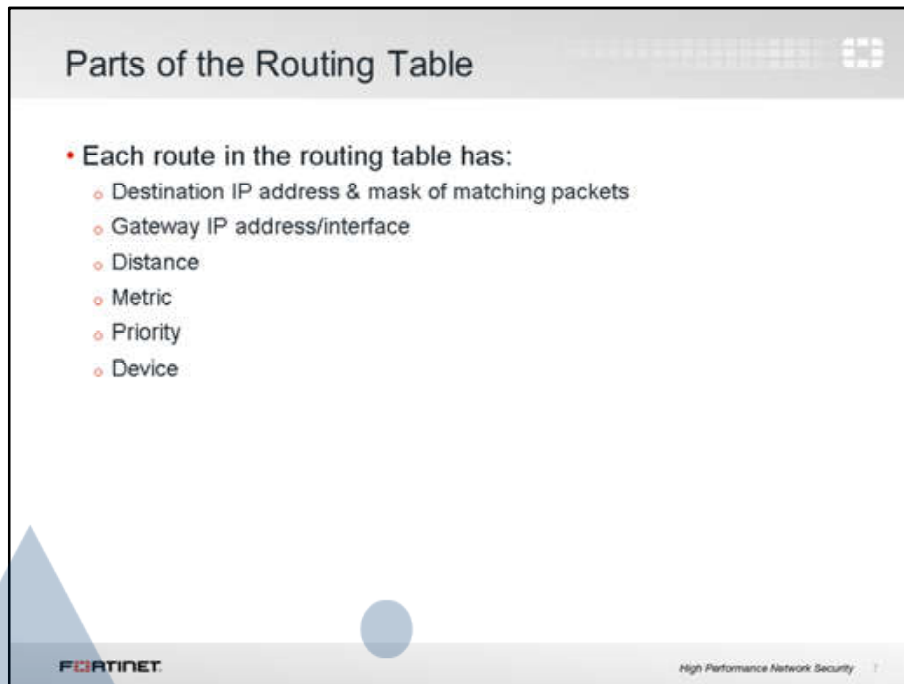
Which rows are “extra” – automatic entries that aren’t from your static routes configuration?

- **Directly connected subnets** – When a subnet is assigned to a FortiGate’s interface, a route to the subnet is automatically added to the routing table. The FortiGate knows how to route those packets.
- **Dynamic routes** – On larger networks, your FortiGate may receive routes from other routers, via protocols such as BGP. This is faster and more scalable than manually configuring many routers.

Which configured routes *aren’t* loaded into this table?

- **Worse routes to the same IP** – Only the best paths should be used. We will see in a later slide how the best path is elected when there are multiple routes to the same destination.
- **Policy routes** – These are omitted, too. Why? By design, policy routes override the routing table – we don’t want them to be ignored, losing precedence to OSPF or static routes. So they have to be in a separate table, which is searched before this one. We’ll discuss policy routes later.

So remember, expect differences from your configured list of static routes. And when troubleshooting, don’t only check this table. Also check the table for policy-based routes, and (if you’re using dynamic routing) your other routers.



In the routing table, each of the entries has a few pieces of data, such as distance and gateway IP. They are used to relay or deliver each matching packet.

Destination IP addresses and gateway routers are self-explanatory. The device is the name of the outgoing interface where the packet will be routed to. But what about the distance, metric, and priority? How do they effect which routing path packets will use?

Let's explain each briefly.

Distance

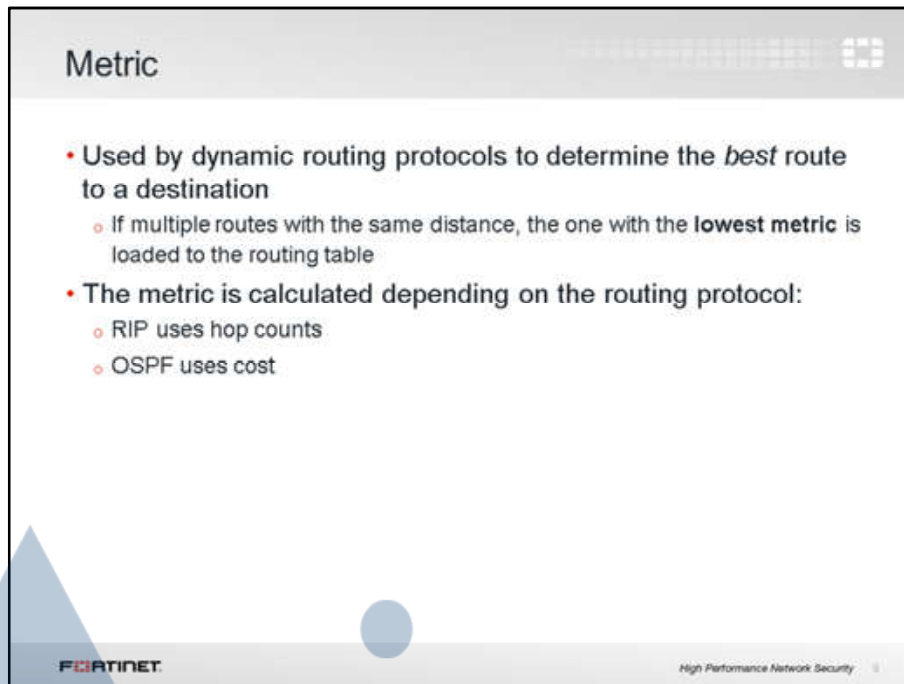
- Estimates the *quality* of each routing protocol and static route
 - A **lower distance** is considered **more reliable**
 - If multiple routes, the one with the **lowest distance** is loaded to the routing table
- Default distance values:

◦ Directly connected	0
◦ DHCP gateway	5
◦ Static routes	10
◦ EBGp routes	20
◦ OSPF routes	110
◦ RIP routes	120
◦ IBGP routes	200

FORTINET High Performance Network Security

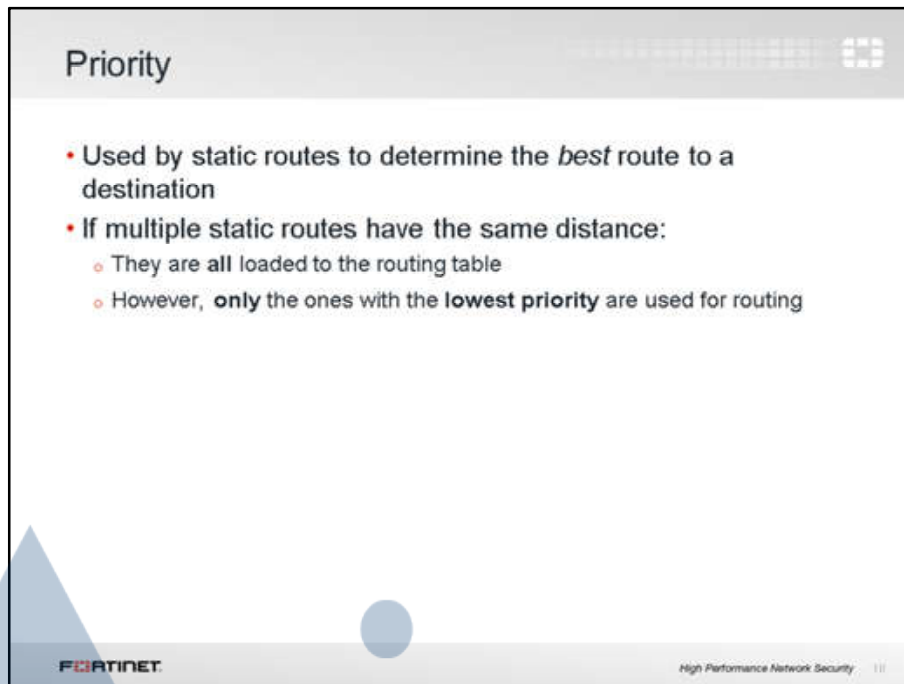
Distance, or administrative distance, is a number that estimates the reliability or quality of each routing protocol and static route. If there are two routes to the same destination, the one with the lower distance is added or loaded to the routing table, as it is considered to be more reliable.

By default, for example, routes learned via the RIP protocol have a higher distance than routes learned via the OSPF protocol, as OSPF is considered to be more accurate than RIP.



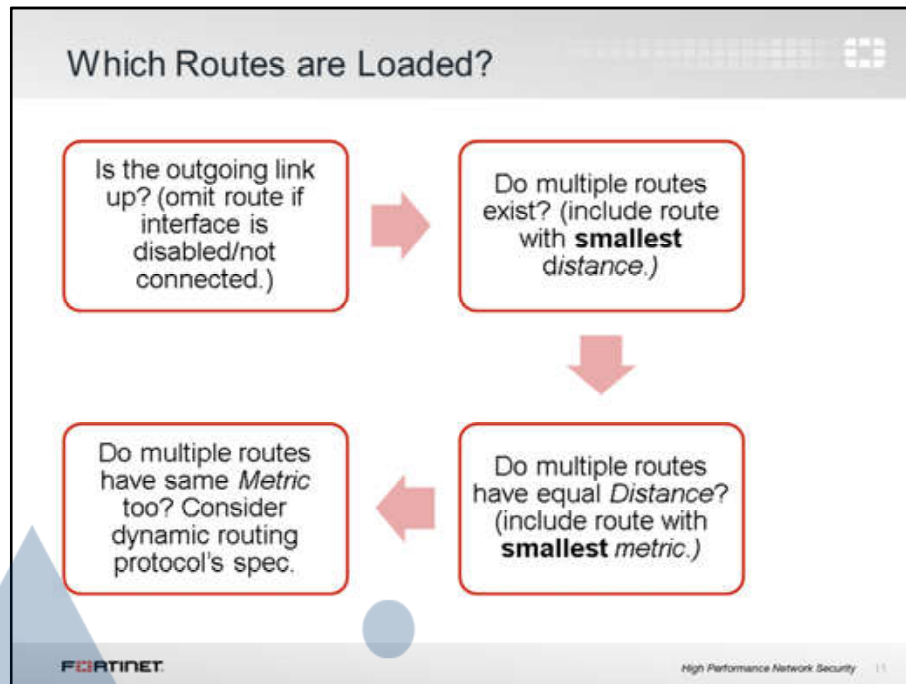
In the case of routes learned via a dynamic routing protocols, metric is another element that is used to determine the best route to a destination. If two routes have the same distance, the metric is then used for tie breaking. The route with the lowest metric is loaded to the routing table.

How the metric is measured depends on the routing protocol. RIP uses hop counts: how many routers must be used to reach the destination. OSPF uses cost, which is determined by how much bandwidth a link has.



In the case of static routes, the priority is used for tie breaking when the distances are the same. FortiGate will use the route with the smallest number configured in the route's priority setting.

In other words, if we have two routes with the same distance to the same destination, only the one with the smallest priority will be used. Note that unlike with distances/metrics, both routes with the same distance are loaded into the routing table. However, only the route with the smallest priority will be routing traffic. This, as we will see later, is an important concept when dealing with reverse forwarding path check issues.



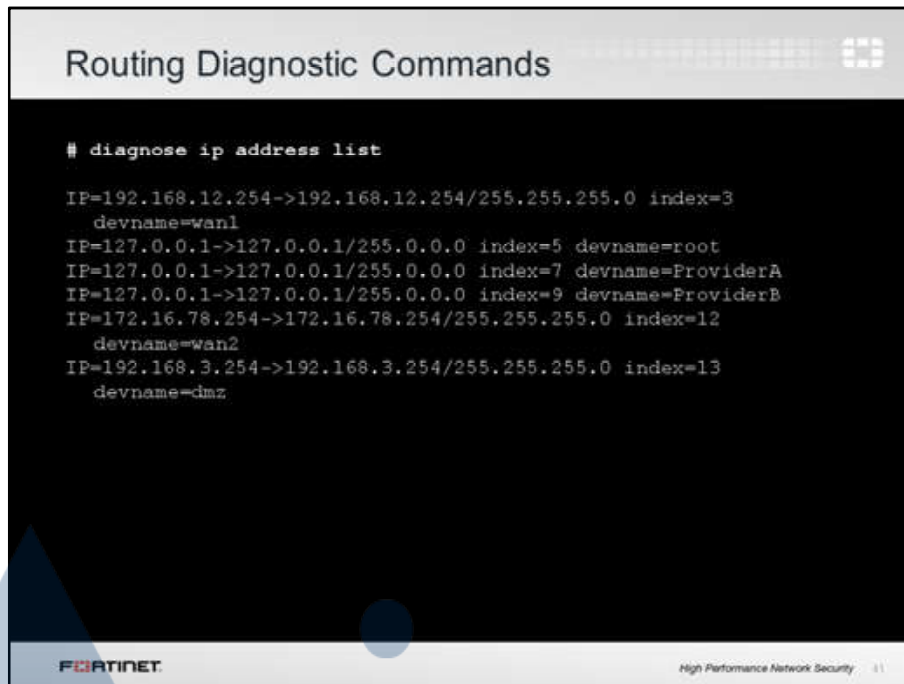
This is summary of the logic behind which routes are loaded into the routing table.

Routes are only active if the interface is currently both physically linked and administratively "up." If the cable isn't plugged in, or if a Wi-Fi network has no signal, for example, packets can't be transmitted along that path. All routes through that link will be temporarily unloaded from the table until the link is available again.

When 2 or more active routes have the same destination subnet, only the one with the smallest distance is loaded to the routing table.

If the distances are equal, only the routes with the smallest metric are included.

If the metric also is identical, then, depending on the dynamic routing protocol's rules, FortiGate will select which one to include in the routing table.



The screenshot shows a terminal window titled "Routing Diagnostic Commands". The command entered is "# diagnose ip address list". The output lists several IP addresses and their associated interfaces and indices:

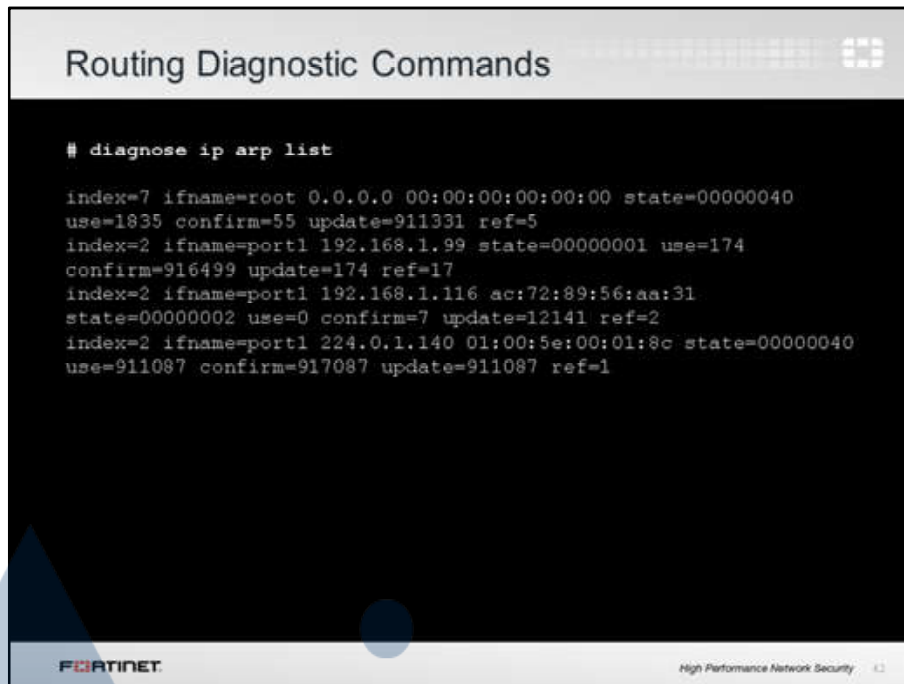
```
# diagnose ip address list

IP=192.168.12.254->192.168.12.254/255.255.255.0 index=3
  devname=wan1
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=5 devname=root
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=7 devname=ProviderA
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=9 devname=ProviderB
IP=172.16.78.254->172.16.78.254/255.255.255.0 index=12
  devname=wan2
IP=192.168.3.254->192.168.3.254/255.255.255.0 index=13
  devname=dmz
```

The FortiGate logo and "High Performance Network Security" tagline are visible at the bottom of the terminal window.

This command gives a quick list of IP addresses associated with each interface.

They can be physical, VLAN, or virtual interfaces.



```
Routing Diagnostic Commands

# diagnose ip arp list

index=7 ifname=root 0.0.0.0 00:00:00:00:00:00 state=00000040
use=1835 confirm=55 update=911331 ref=5
index=2 ifname=port1 192.168.1.99 state=00000001 use=174
confirm=916499 update=174 ref=17
index=2 ifname=port1 192.168.1.116 ac:72:89:56:aa:31
state=00000002 use=0 confirm=7 update=12141 ref=2
index=2 ifname=port1 224.0.1.140 01:00:5e:00:01:8c state=00000040
use=911087 confirm=917087 update=911087 ref=1
```

If you suspect that there is an IP address conflict, or that an IP has been assigned to the wrong interface, you may need to look at the ARP table. This command is used for that purpose. It shows the interface, IP address, and associated MAC address.




Demo Version




In this lesson, we will show how to configure virtual domains (VDOMs) and common usage examples.

Objectives

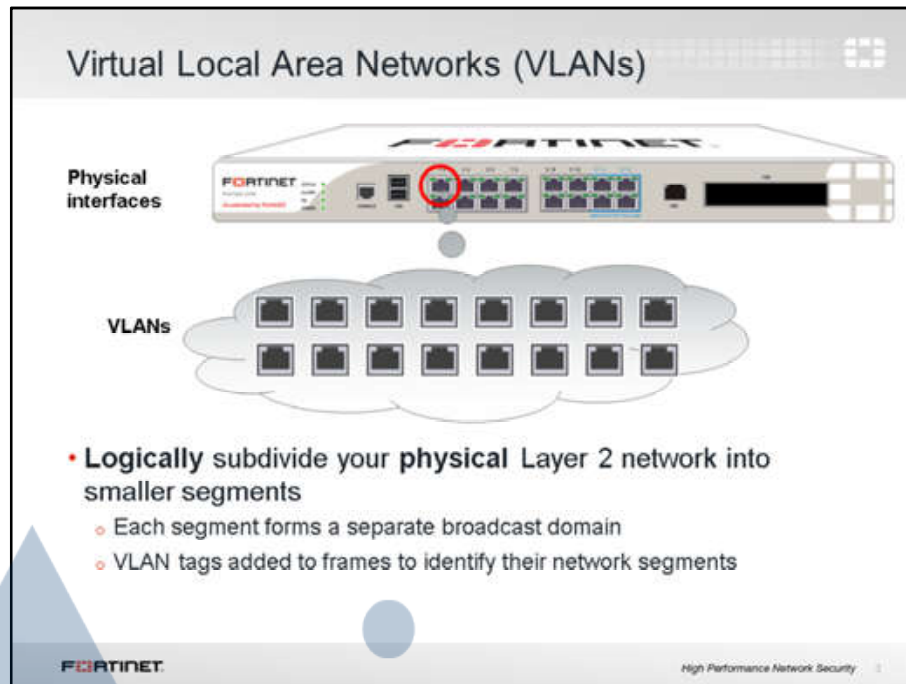
- Use VLANs to logically divide a Layer 2 network into multiple segments
- Use VDOMs to split a FortiGate into multiple virtual units
- Limit the resources allocated globally and per-VDOM
- Create administrative accounts with the access limited to one or more VDOMs
- Route traffic between VDOMs by using inter-VDOM links



High Performance Network Security

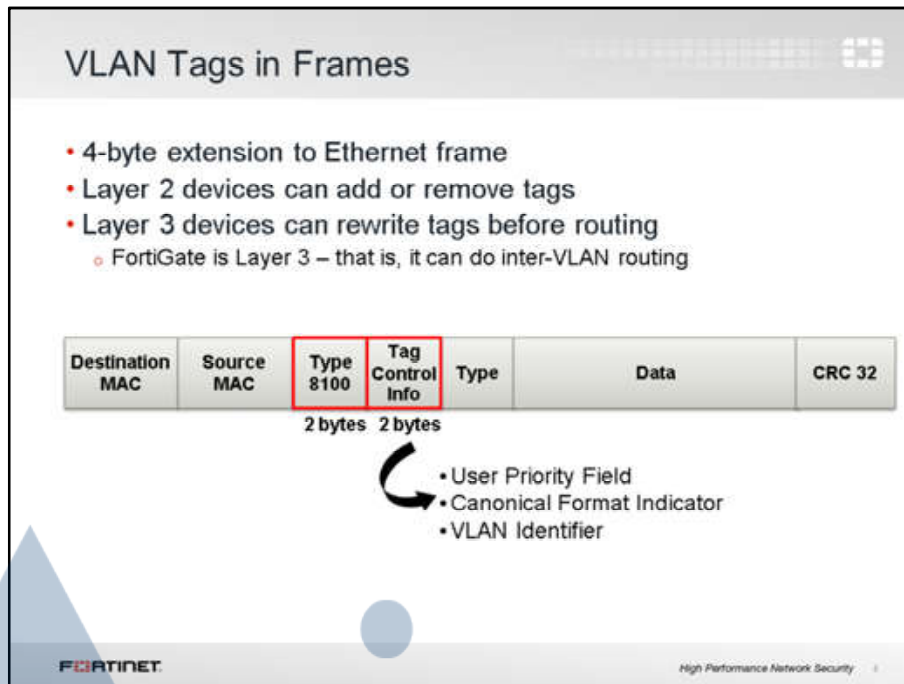
After completing this lesson, you should have these practical skills that you can use to create VDOMS and VLANs, which are commonly used logical interfaces when working with virtual domains in a FortiGate. You will also learn to limit the resources allocated to each VDOM and create per-VDOM administrative accounts. The lesson also covers inter-VDOM connectivity.

Lab exercises can help you to test and reinforce your skills.



VDOMs are a virtualization within FortiOS, providing virtual firewalls. Interfaces have VDOM membership, the interface a packet arrives on determines which VDOM will process the traffic. Interfaces can be physical or logical; IEEE 802.1Q VLANs are a logical interface commonly used with VDOMS.

VLANs splits your physical LAN into multiple logical LANs. Each VLAN forms a separated broadcast domain. In a same interface (or collision domain) multiple VLANs can coexist. In this way, a physical interface is split into two or more logical interfaces. A tag is added to each Ethernet frame to identify the VLAN that it belongs to.

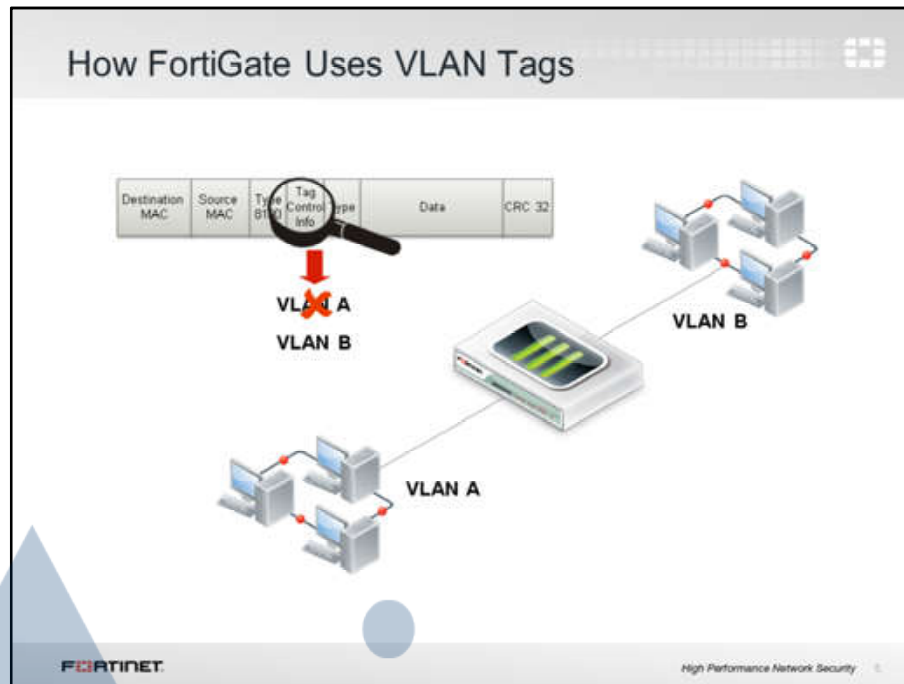


This slide shows a Ethernet frame. The frame contains the MAC addresses, the type, the data payload, and a CRC code to confirm that is not corrupted.

In the case of Ethernet frames with VLAN tagging, according with the 802.11q standard, 4 more bytes are inserted after the MAC addresses. They contain an ID number that identifies the VLAN.

An OSI Layer 2 device, such as a switch, can add or remove these tags from Ethernet frames. But it cannot change them.

A Layer 3 device, such as router or a FortiGate, can change the VLAN tag before proceeding to route the traffic. In this way, they can route traffic between VLANs.



When operating in NAT/route mode, the FortiGate device operates as a Layer 3 router in its most basic configuration. In this mode, a VLAN is an interface on the device. VLAN tags may be added on egress, removed on ingress, or rewritten based on a routing decision.

When operating in Transparent mode, the FortiGate device operates as a Layer 2 bridge in its most basic configuration. In this mode, a VLAN is an identifier for identifying traffic flows. The VLAN does not exist on the FortiGate, in FortiOS the broadcast domain which is an accepted as a property of a VLAN, is defined by the virtual domain, and the broadcast domain can only be modified using forwarding domains as a sub-division. So to create a VLAN like behavior on FortiOS in transparent mode, you would need ingress and egress VLAN interfaces using the same VID, and a forwarding domain within the virtual domain containing those two interface, plus firewall policies to allow traffic.




Demo Version




In this lesson, we will show you how to configure FortiGate to operate in transparent mode, and discuss differences with NAT mode.

Objectives

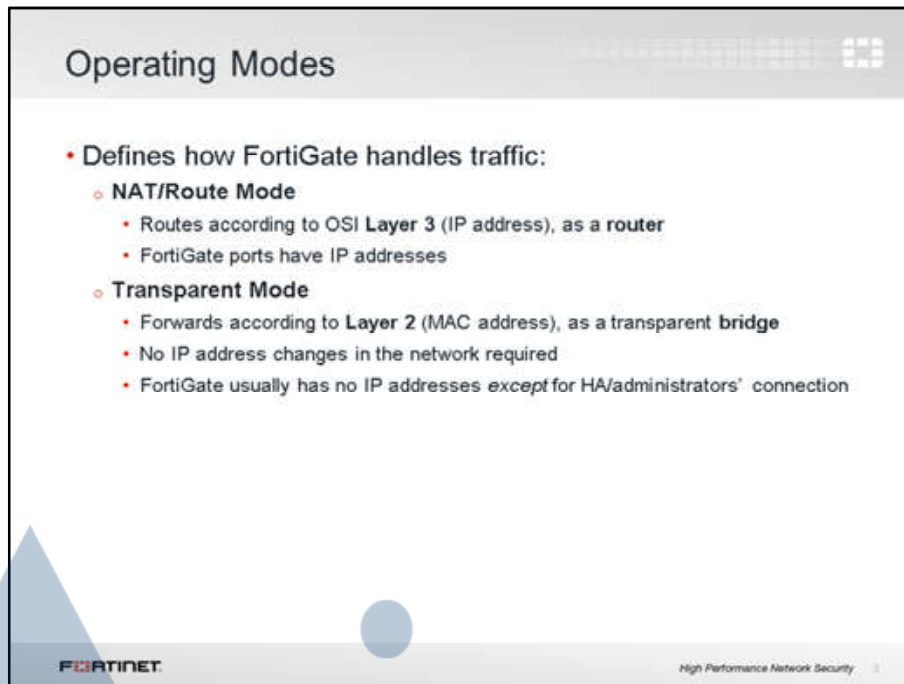
- Choose the most appropriate FortiGate operating mode
- Segment the network into multiple forwarding domains
- Prevent broadcast storms and MAC address flapping by using port pairing
- Install the FortiGate in networks running spanning tree protocol (STP)
- Monitor the MAC address table



High Performance Network Security

After completing this lesson, you should have these practical skills that you can use to configure FortiGate features that are specific to transparent mode, such as STP and port pairing.

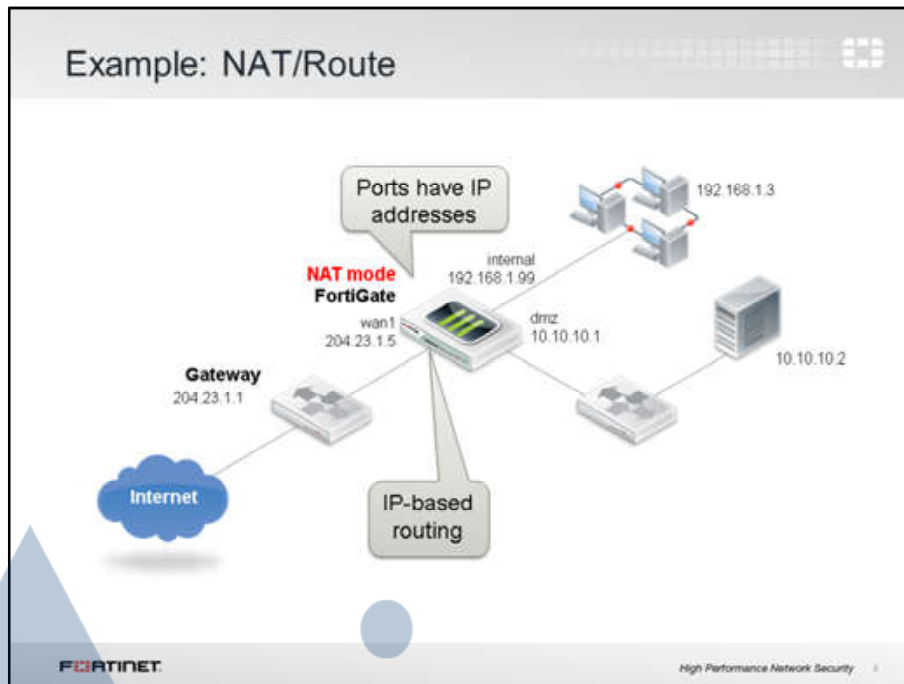
Lab exercises can help you to test and reinforce your skills.



Traditional IPv4 firewalls and NAT mode FortiGates are routers, not just switches. So, each interface has to be in different subnets and each forms different broadcast domains. The FortiGate routes IP packets based on the IP header information, overriding the source MAC address. So, if a client sends a packet to a server connected to a different FortiGate interface, the packet will arrive to the server with a FortiGate's MAC address, instead of the client's.

In the case of transparent mode, FortiGate forwards frames without changing the MAC addresses. When the client receives a packet from a server connected to a different FortiGate interface, the frame contains the server's real MAC address – FortiGate doesn't rewrite the MAC header. The FortiGate is a Layer 2 bridge or switch. So, the interfaces do not have IP addresses and all belong (by default) to the same broadcast domain.

This means that a transparent mode FortiGate can be installed in a customer network without changing the customer's IP address plan. Some customers, specially large organizations, don't want to reconfigure thousands of devices to define a new "internal" vs. "external" network.



Here is an example showing NAT mode.

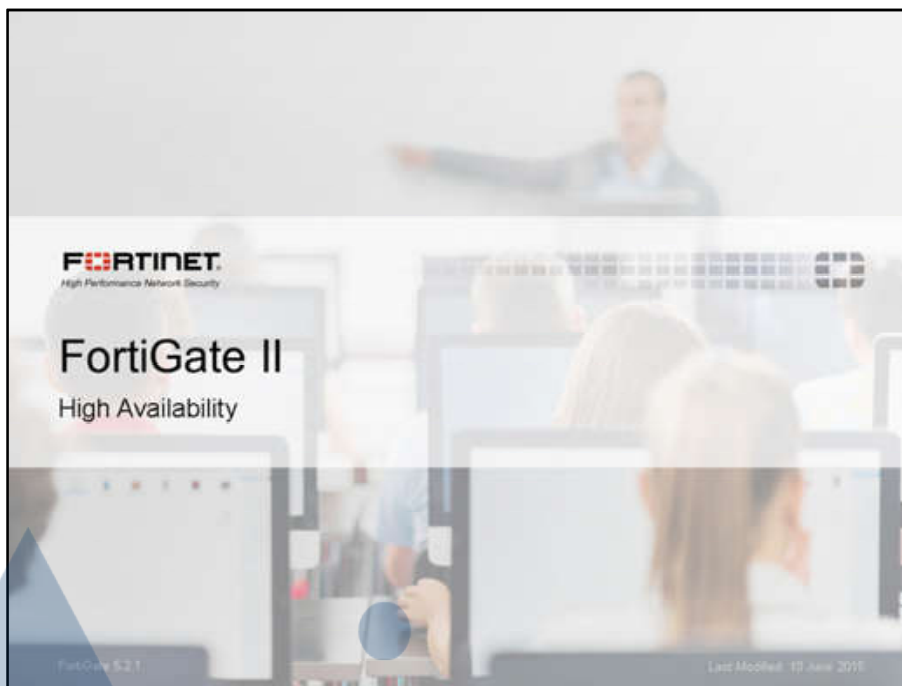
FortiGate has 3 connected ports, each with separate IP subnets. All interfaces on the FortiGate have IP addresses, and, in this case, NAT translates between networks. Firewall policies allow traffic to flow between networks.

FortiGate handles packets according to their routes, which are in most of the cases based on the destination IP address (at Layer 3 of the OSI model).

Clients on each subnet send frames that are destined for a FortiGate MAC address – not the real MAC address of the server.




Demo Version





In this lesson, you will learn about FortiGate high availability (HA).

Objectives

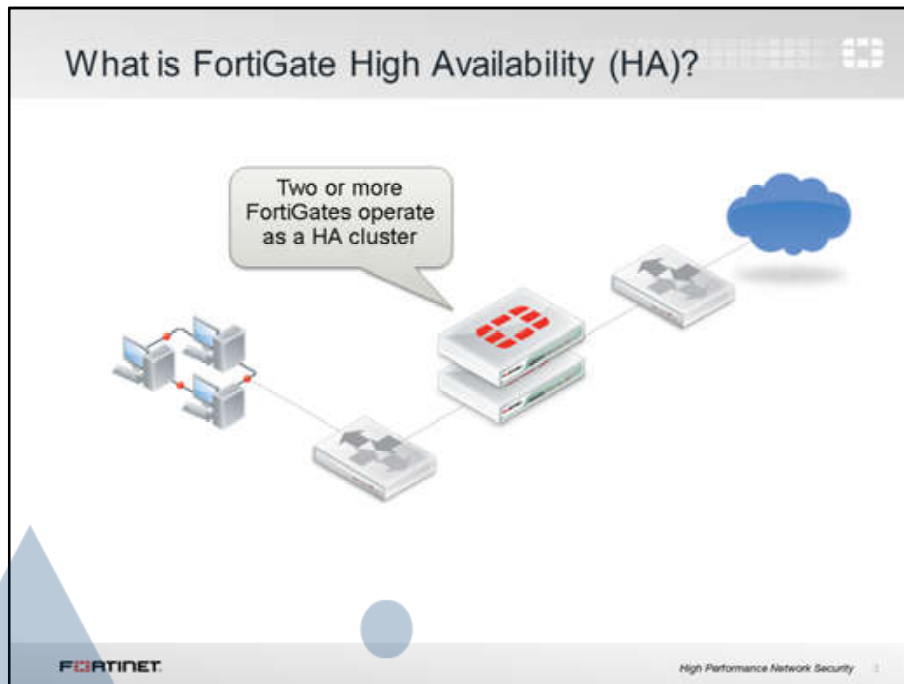
- Choose the right high availability (HA) operation mode
- Implement and configure a HA solution
- Configure session synchronization for seamless failover
- Set up FortiGate session life support protocol (FGSP)
- Upgrade a HA cluster's firmware
- Use virtual clustering for per-VDOM high availability
- Verify the normal operation of a HA cluster





When you've completed this lesson, you should be able to configure, operate, and monitor a FortiGate HA cluster.

Lab exercises can help you to test and reinforce your skills.



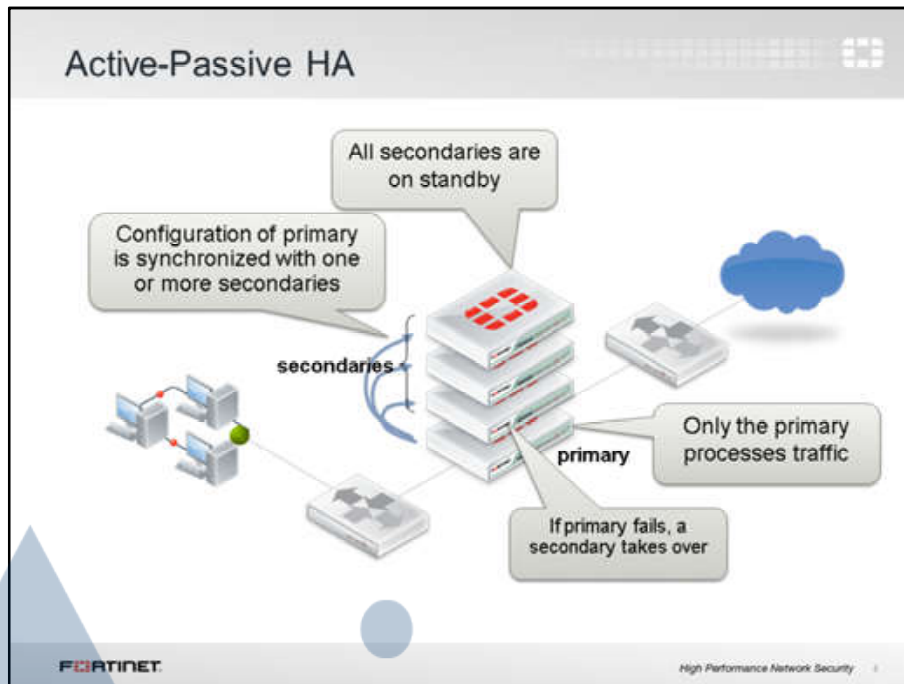
The idea of HA is simple. HA links and synchronizes 2 or more devices.

Like HA you may have seen on other vendors' products, one FortiGate device acts as the primary appliance (also called the active FortiGate): it synchronizes its configuration to the other devices. The other FortiGates are called secondary or standby devices.

A heartbeat link among all the appliances is used to detect when any unit becomes unresponsive.

What is synchronized among the units? Are all FortiGate devices processing traffic? Does HA literally improve availability, or does it improve throughput?

The answers vary depending on the HA mode. There are currently two HA modes available: active-active, and active-passive. Let's examine the differences.



(slide contains animation)

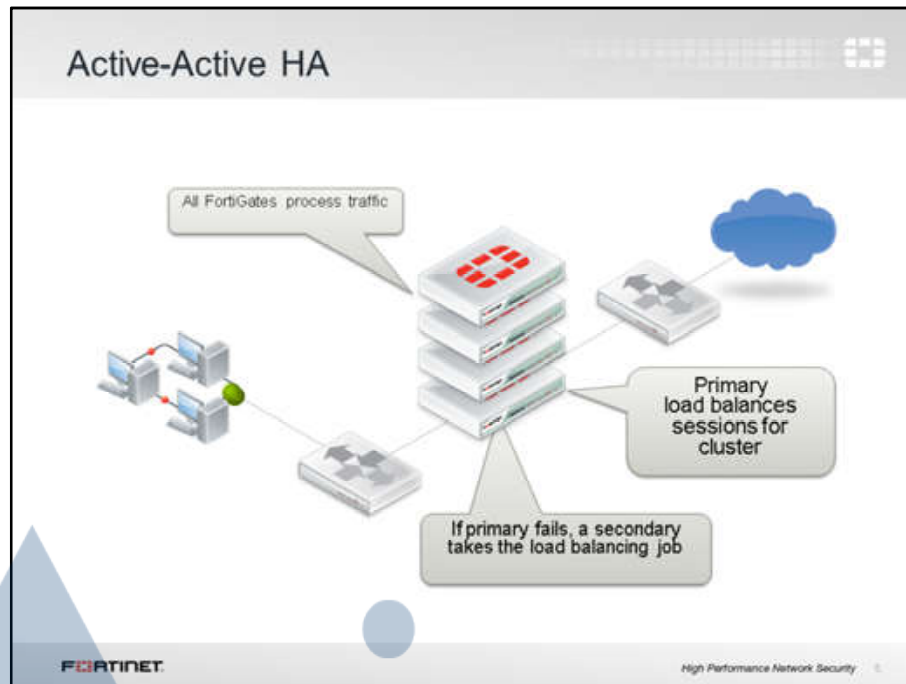
Let's examine first the active-passive mode. In any of the two HA operation modes, the configuration of the secondary FortiGates are synchronized with the configuration in the primary device.

(click)

In the case of the active-passive mode, the primary FortiGate is the only FortiGate device that actively processes traffic. secondary FortiGates remains in passive mode monitoring the status of the primary device.

(click)

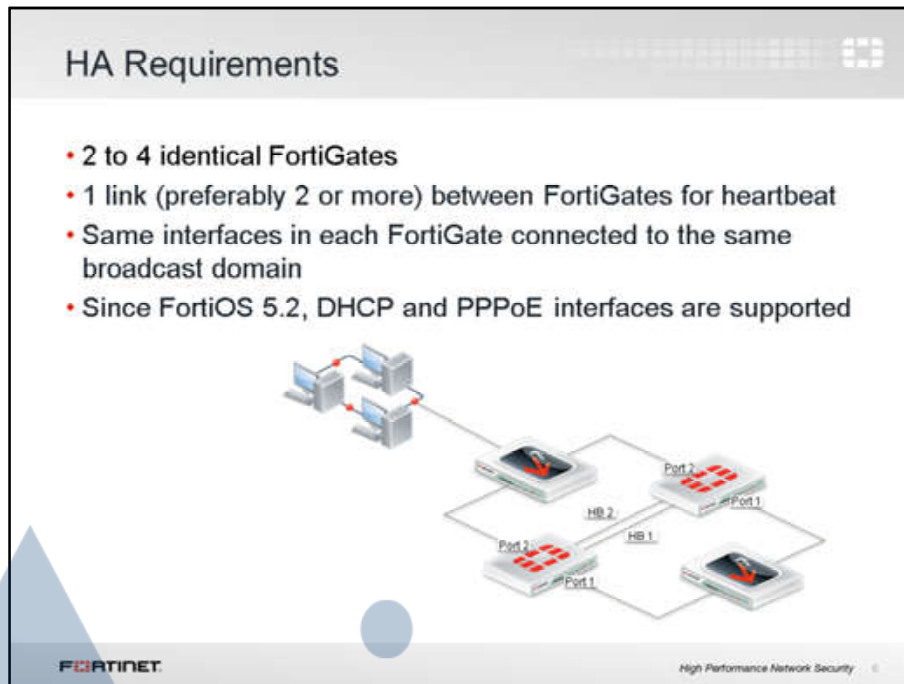
If a problem is detected in the primary FortiGate, one of the secondary devices will take over the primary role. This event is what we call HA failover.



The other HA mode is called “active-active.”

Like with active-passive HA, in active-active, all FortiGates' configurations are synchronized. Also, if a problem is detected with the primary device, one of the secondaries will take over the role of primary traffic processing.

However, one of the main differences with active-passive mode is that in active-active mode all of the FortiGates are processing traffic. As we will see later, one of the tasks of a primary FortiGate in active-active mode is to balance some of the traffic among all the secondary devices.



(slide contains animation)
FortiGate HA requires...

First, at least 2, but up to 4, FortiGate devices with the same:

- Firmware
- Hardware model and VM license
- Hard drive capacity and partitions
- Operating mode (transparent or NAT)

(click)

Second, at least 1 link between the FortiGate units for the HA communication, which is called heartbeat traffic. For redundancy, up to 8 heartbeat interfaces can be used. If one link fails, HA will use the next one by priority and position in the heartbeat interface list.

(click)

Third, the same interfaces on each FortiGate unit have to be connected to the same switch or LAN segment. Notice that in this illustration, the FortiGate units are redundant to mitigate failure. But the switches and their links still are a single point of failure. As we will see later, you can also have redundancy in the network switches and links.

(click)

One important change in FortiOS 5.2, related with HA, is that now the cluster can include interfaces whose IP addresses are assigned dynamically, via either DHCP or PPPoE. Prior to FortiOS 5.2, a HA cluster could only contain interfaces with static IP addresses.



Demo Version