

FortiOS - New Features Guide

Version 6.4.0

FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO GUIDE
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/support-and-training/training.html

NSE INSTITUTE
https://training.fortinet.com

FORTIGUARD CENTER
https://fortiguard.com/

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com



April 13, 2020
FortiOS 6.4.0 New Features Guide
01-640-582010-20200413

TABLE OF CONTENTS

Change Log **6**
Security-driven Networking **7**
NGFW 7
Log buffer on FortiGates with an SSD disk 7
Route leaking between VRFs 10
WAD and Proxyd SSL logging improvement 12
Support SSL mirroring in proxy mode 17
SSL-based application detection over decrypted traffic in a sandwich topology 20
Consolidated IPv4 and IPv6 policy configuration 20
Allow creation of ISDB objects with regional information 23
Force HA failover for testing and demonstrations 25
IP definitions database merged into the internet service database 28
Security Profiles enhancements 30
AntiVirus uses Extended DB by default 36
Support UTM inspection on asymmetric traffic in FGSP 37
Support UTM inspection on asymmetric traffic on L3 39
Add encryption for L3 on asymmetric traffic in FGSP 41
Use anycast to communicate with FortiGuard servers 42
SD-WAN 44
IBGP and EBGP support in VRF 45
SD-WAN event log subtype 47
SD-WAN logging improvement to identify matched application 51
SD-WAN configuration portability 51
SD-WAN log format improvements 54
SD-WAN monitor on ADVPN shortcuts 59
SD-WAN GUI and monitoring enhancements 60
Enhance ADVPN to support UDP hole punching for spokes behind NAT 65
SD-WAN health check packet enhancement 68
Weighted round robin for IPsec aggregate tunnels 68
Default_DNS performance SLA profile 70
Interface speedtest 71
Support SD-WAN integration with OCVPN 73
Allow FortiClient to join OCVPN 81
Secure access 84
Switch controller - quarantine by redirect 85
Wireless IPv6 support 87
Support for spectrum analysis of FortiAP E models 93
Increase in maximum number of managed FortiAPs 99
VLAN interface templates for FortiSwitch devices 100
Improved FortiSwitch support 104
Even distribution of FortiAP reports 104
View detailed information for individual WiFi connections 107
VLAN probe report 116
FortiAP client load balancing per AP 120
Layer three ACL configurations for Wireless APs 121

Zero-trust Network Access **124**
NAC 124
IoT detection service 124
Support NAC policies on switch ports 126
Added ability in FortiSwitch to query FortiGuard IoT service for device details 130
FortiSwitch voice device detection 132
AI-driven Security Operations **137**
ATP 137
Credential phishing prevention 137
Extend ISDB to include well-known MAC address list 139
GeoIP matching by registered and physical location 141
Fabric Management Platform **143**
Single pane 143
SAML SP for VPN authentication 143
Display cloud service communications statistics 146
Confirmation prompt when creating new VDOMs 147
Admin profile option for diagnostic access 148
Override FortiAnalyzer and syslog server settings 148
FortinetOne renamed FortiCloud 152
Security Fabric automation 153
SDN connector for Cisco ACI northbound API integration 154
Support multiple SDN connector instances for Cisco ACI and Nuage 157
Automation stitches 163
Slack notification action 172
Simplify Azure Fabric connector configuration for a FortiGate-VM deployed on Azure 176
Add multifunction tooltip for Fabric connectors 178
Integrate FortiAnalyzer management into the Security Fabric using SAML SSO 180
Group address objects synchronized from FortiManager 182
Simplify FortiClient EMS setup 184
Simplify the synchronization of EMS tags and configurations 187
Allow FortiNAC to join the Security Fabric 189
Exchange Server connector with Kerberos KDC auto-discovery 192
Redesign Security Rating scorecards 193
Redesign Fortinet Fabric Connectors and Fabric setup pages 195
Display endpoints in Topology using donut chart 198
Support filtering on AWS autoscaling group for dynamic address objects 200
Support dynamic address objects in real servers under virtual server load balance 201
Consolidate Monitor and FortiView pages 202
Using the root FortiGate with disk to store historic user and device information 208
Synchronizing objects across the Security Fabric 208
Other **213**
IPv6 213
IPv6 geography-based address support 214
Support for IPv6 in central SNAT table 216
FQDN support for remote gateways 218
Redirect to WAD after handshake completion 220
No session timeout 220

UUID field added to all policy types ... 222
Use CP9/SoC3 entropy source ... 224
Authentication support for upstream proxy in transparent proxy mode ... 224
SNMP bridge MIB module support ... 225
Support SHA-2 for SNMPv3 ... 227
Set minimum RIP update timer to one second ... 228
Dynamic address support for SSL VPN policies ... 228
Configure the FortiAuthenticator ... 230
Fortinet Single Sign-On Collector Agent ... 231
Configure the FortiGate ... 231
Confirmation ... 236
SNAT support for policies with virtual wire pairs ... 237
GUI support for FortiLink groups ... 239
Increase in maximum number of VIP real servers ... 239
WAN interface bandwidth log ... 240
Source interface setting for NetFlow data ... 241
FortiSwitch link status visibility improvements ... 244
Support up to 24 interfaces on FortiGate VM ... 245
Maintain radio SSID WLAN IDs ... 247
Support for Okta RADIUS attributes filter-Id and class ... 250
FortiOS image signing and verification ... 252
ICAP response filtering ... 254
Example ... 254
Support for FAP431F and FAP433F ... 256
Enhanced autoscale clusters for FortiGate VM ... 259
SNMP traps and query for monitoring DHCP pool ... 261
Firmware upgrade notifications ... 262
Identify the XAUI link used for a specific traffic stream ... 263
DHCP client options ... 264
NAS-IP support per SSL-VPN realm ... 265
Matching multiple parameters on application control signatures ... 267
Detecting IEC 61850 MMS protocol in IPS ... 269
IP address tooltips ... 271
Interface-based traffic shaping ... 273

Change Log

Date	Change Description
2020-03-31	Initial release.
2020-04-06	Added Support filtering on AWS autoscaling group for dynamic address objects on page 200 and Support dynamic address objects in real servers under virtual server load balance on page 201.
2020-04-07	Added Allow FortiClient to join OCVPN on page 81, Redesign Fortinet Fabric Connectors and Fabric setup pages on page 195, Display endpoints in Topology using donut chart on page 198, and Consolidate Monitor and FortiView pages on page 202 .
2020-04-08	Added Added ability in FortiSwitch to query FortiGuard IoT service for device details on page 130, Redesign Security Rating scorecards on page 193, Using the root FortiGate with disk to store historic user and device information on page 208, and Support SD-WAN integration with OCVPN on page 73.
2020-04-13	Added FortiSwitch voice device detection on page 132 and Synchronizing objects across the Security Fabric on page 208.

Security-driven Networking

This section lists the new features added to FortiOS for security-driven networking:

- NGFW on page 7
- SD-WAN on page 44
- Secure access on page 84

NGFW

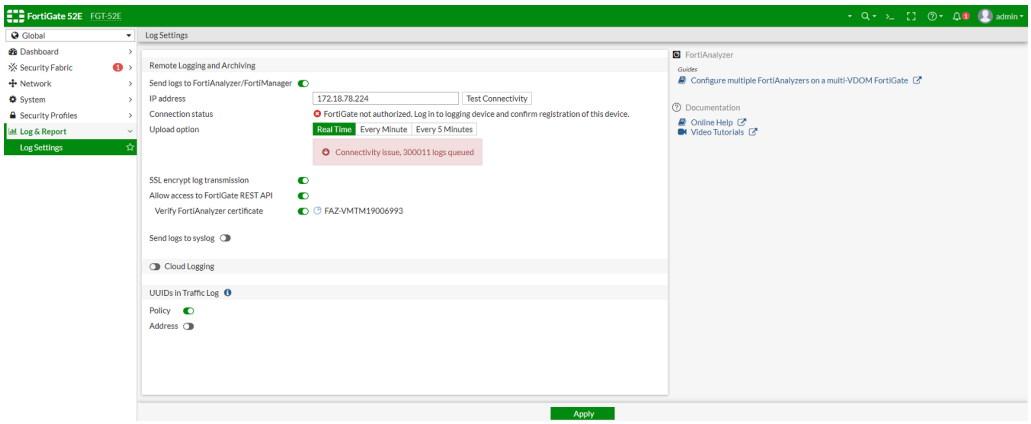
This section includes NGFW features added to FortiOS:

- Log buffer on FortiGates with an SSD disk on page 7
- Route leaking between VRFs on page 10
- WAD and Proxyd SSL logging improvement on page 12
- Support SSL mirroring in proxy mode on page 17
- SSL-based application detection over decrypted traffic in a sandwich topology on page 20
- Consolidated IPv4 and IPv6 policy configuration on page 20
- Allow creation of ISDB objects with regional information on page 23
- Force HA failover for testing and demonstrations on page 25
- IP definitions database merged into the internet service database on page 28
- Security Profiles enhancements on page 30
- AntiVirus uses Extended DB by default on page 36
- Support UTM inspection on asymmetric traffic in FGSP on page 37
- Support UTM inspection on asymmetric traffic on L3 on page 39
- Add encryption for L3 on asymmetric traffic in FGSP on page 41
- Use anycast to communicate with FortiGuard servers on page 42

Log buffer on FortiGates with an SSD disk

FortiGates with an SSD disk have a configurable log buffer. When the connection to FortiAnalyzer is unreachable, the FortiGate is able to buffer logs on disk if the memory log buffer is full. The logs queued on the disk buffer can be sent successfully once the connection to FortiAnalyzer is restored.

The number of logs queued on the disk buffer is visible in the *Log & Report* > *Log Settings* page:



The queued logs are buffered to the memory first and then disk. `Main miglogd` handles the disk buffering job, while `miglogd-children` handles the memory buffering. Disk buffer statistics only appear under `Main miglogd`, and memory buffer statistics only appears under `miglogd-children`. If the total buffer is full, new logs will overwrite the old logs.

To configure the log buffer:

1. Allocate disk space (MB) to temporarily store logs to FortiAnalyzer:

```
config system global
    set faz-disk-buffer-size 200
end
```

2. Check the `Main miglogd` and `miglogd-children` statistics. The 200 MB disk buffer has been set, and there are currently no logs buffered in memory or on disk when FortiAnalyzer is reachable:

```
# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

    memory queue:
        num:0 size:0(0MB) max:101906636(97MB) logs:0

    disk max queue size:200MB total:0MB
        totol items:0
        disk queue agents:
            devid:-1-10-0-1
            buffer path:/var/log/qbuf/10.0/1
            saved size:0MB cached size:0
            save roll:0 restore roll:0
            restore id:0 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

    memory queue:
        num:0 size:0(0MB) max:101906636(97MB) logs:0

    disk queue client:
```

```
        devid:-1-10-0-1 status:buffering
        Total in cache:0 size:0(0MB) max:4MB logs:0
```

3. Disable the connection between the FortiGate and FortiAnalyzer. For example, delete the FortiGate from the FortiAnalyzer authorized device list.
   Assuming a massive number of logs (~ 300000) are recorded during this downtime, the logs will be queued in the memory buffer first. If the memory buffer is full, then the remaining logs will be queued on the disk buffer.
4. Check the `Main miglogd` and `miglogd-children` statistics again. All 97 MB of the memory buffer is occupied, and 76 of the 200 MB has been taken from the disk buffer:

```
# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0

        disk max queue size:200MB total:76MB
                totol items:128917
                disk queue agents:
                        devid:-1-10-0-1
                        buffer path:/var/log/qbuf/10.0/1
                        saved size:76MB cached size:3324984
                        save roll:19 restore roll:0
                        restore id:0 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 165721 used: 101908358(97MB) allocated: 106449280
(101MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:165718 size:101906500(97MB) max:101906636(97MB) logs:165718

        disk queue client:
                devid:-1-10-0-1 status:restoring
                restore id:1267 space:0MB
                Total in cache:3 size:1858(0MB) max:4MB logs:3
```

The overall `miglogd` statistics shows the total cached logs is the sum of the logs buffered in memory and on disk:

```
# diagnose test application miglogd 6
mem=0, disk=11, alert=0, alarm=0, sys=0, faz=300053, faz-cloud=0, webt=0, fds=0
interface-missed=44
Queues in all miglogds: cur:165718  total-so-far:165718
global log dev statistics:
faz 0: sent=0, failed=0, cached=300053, dropped=0 , relayed=0
Num of REST URLs: 0
```

5. Enable the connection between FortiAnalyzer and the FortiGate.
6. After a while, check the `miglogd` statistics to confirm that all buffered logs are being sent to FortiAnalyzer successfully:

```
# diagnose test application miglogd 6
mem=0, disk=11, alert=0, alarm=0, sys=0, faz=300058, faz-cloud=0, webt=0, fds=0
```

```
interface-missed=44
Queues in all miglogds: cur:4294832957  total-so-far:165726
global log dev statistics:
faz 0: sent=300058, failed=0, cached=0, dropped=0 , relayed=0
Num of REST URLs: 15

# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0

        disk max queue size:200MB total:0MB
                totol items:0
                disk queue agents:
                        devid:-1-10-0-1
                        buffer path:/var/log/qbuf/10.0/1
                        saved size:0MB cached size:0
                        save roll:20 restore roll:20
                        restore id:1267 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0

        disk queue client:
                devid:-1-10-0-1 status:buffering
                Total in cache:0 size:0(0MB) max:4MB logs:0
```

Route leaking between VRFs

This feature provides generic route leaking capabilities between locally defined VRFs (VRF-lite). If VRF leaking is not configured, VRFs are isolated.

In this example, interface *npu0_vlink0* belongs to VRF 10 and is used to leak 1.2.2.2/32 from VRF10 to VRF20, and interface *npu0_vlink1* belongs to VRF 20 and is used to leak 172.28.1.0/24 from VRF20 to VRF10. So, VRF10 can see 172.28.1.0/24, and VRF20 can see 1.2.2.2/32.

To configure VRF leaking:

1. Configure the prefix list and route map to filter what will be leaked:

```
config router prefix-list
    edit "1"
        config rule
            edit 1
                set prefix 1.2.2.2 255.255.255.255
            next
```

```
        end
    next
    edit "2"
        config rule
            edit 1
                set prefix 172.28.1.0 255.255.255.0
            next
        end
    next
end

config router route-map
    edit "from10"
        config rule
            edit 1
                set match-ip-address "1"
            next
        end
    next
    edit "from20"
        config rule
            edit 1
                set match-ip-address "2"
            next
        end
    next
end
```

2. Configure the VDOM link interfaces for the leaking and routing:

```
config system interface
    edit "npu0_vlink0"
        set vdom "root"
        set vrf 10
        set ip 172.16.201.1 255.255.255.0
        set allowaccess ping https ssh snmp http
    next
    edit "npu0_vlink1"
        set vdom "root"
        set vrf 20
        set ip 172.16.201.2 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
    next
end
```

3. Configure the BGP VRF leak:

```
config router bgp
    set as 44
    set router-id 4.4.4.4
    config neighbor
        edit "172.16.200.1"
            set soft-reconfiguration enable
            set remote-as 11
            set update-source "port1"
        next
        edit "172.16.202.1"
            set soft-reconfiguration enable
```

```
            set remote-as 22
            set update-source "port3"
        next
    end
    config vrf-leak
        edit "10"
            config target
                edit "20"
                    set route-map "from10"
                    set interface "npu0_vlink0"
                next
            end
        next
        edit "20"
            config target
                edit "10"
                    set route-map "from20"
                    set interface "npu0_vlink1"
                next
            end
        next
    end
end
```

4. Confirm that the filtered routed leaked as expected:

```
# get router info routing-table all
Routing table for VRF=10
B       1.1.1.1/32 [20/0] via 172.16.200.1, port1, 01:03:16
B       1.2.2.2/32 [20/0] via 172.16.200.1, port1, 01:03:16
B       172.28.1.0/24 [20/0] via 172.16.201.2, npu0_vlink0, 00:00:17   <<<<<<<<<<<<<<Leaked
into VRF10 from VRF20

Routing table for VRF=20
B       1.2.2.2/32 [20/0] via 172.16.201.1, npu0_vlink1, 00:00:15   <<<<<<<<<<<<<<Leaked
into VRF 20 from VRF10
B       172.28.1.0/24 [20/0] via 172.16.202.1, port3, 01:03:16
B       172.28.2.0/24 [20/0] via 172.16.202.1, port3, 01:03:16
```

WAD and Proxyd SSL logging improvement

During deep inspection and certificate inspection, various logs generated from certificate issues now use a consistent log format. Additional details have also been added to these logs. A new option, `ssl-negotiation-log`, captures results of unsupported SSL negotiations.

SSL/SSH protocol options:

A new option, `set ssl-negotiation-log {enable | disable}`, was added to the option set.

```
config firewall ssl-ssh-profile
   edit "deep-inspection"
      set ssl-anomalies-log {enable | disable}
      set ssl-exemptions-log {enable | disable}
      set ssl-negotiation-log {enable | disable}
   next
end
```

To log invalid certificates:

```
config firewall ssl-ssh-profile
   edit "deep-inspection"
      set ssl-anomalies-log enable
   next
end
```

FortiGate will generate the ssl anomalies log when traffic triggers ssl certificate anomalies.

In the HTTPS and SMTPS version of the traffic and ssl utm logs:

- The `logid` and the server certificate CN are the same.
- The `msg` field in the SSL UTM logs are similar.

Log type	HTTP	SMTPS
Traffic log	1: date=2020-02-06 time=10:54:36 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581015276280004271 tz="-0800" srcip=10.1.100.66 srcport=45068 srcintf="port2" srcintfrole="undefined" dstip=172.16.200.99 dstport=443 dstintf="port3" dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved" sessionid=95917 proto=6 action="server-rst" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="HTTPS" trandisp="snat" transip=172.16.200.7 transport=45068 duration=5 sentbyte=931 rcvdbyte=6818 sentpkt=11 rcvdpkt=11 appcat="unscanned" wanin=0 wanout=0 lanin=696 lanout=696 utmaction="block" countssl=1 crscore=5 craction=262144 crlevel="low" utmref=65503-98	6: date=2020-02-06 time=11:02:57 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581015777090002933 tz="-0800" srcip=10.1.100.66 srcport=57522 srcintf="port2" srcintfrole="undefined" dstip=172.16.200.99 dstport=465 dstintf="port3" dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved" sessionid=96269 proto=6 action="close" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="SMTPS" trandisp="snat" transip=172.16.200.7 transport=57522 duration=5 sentbyte=597 rcvdbyte=216 sentpkt=6 rcvdpkt=4 appcat="unscanned" utmaction="block" countssl=1 utmref=65500-0
SSL UTM log	1: date=2020-02-06 time=10:54:31 **logid="1700062303"** type="utm" subtype="ssl" **eventtype="ssl-anomalies"** level="warning" vd="vdom1"	1: date=2020-02-06 time=11:02:52 **logid="1700062303"** type="utm" subtype="ssl" **eventtype="ssl-anomalies"** level="warning" vd="vdom1"

Log type	HTTP	SMTPS
	eventtime=1581015271212451397 tz="-0800" action="blocked" policyid=1 sessionid=95917 service="HTTPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=45068 dstip=172.16.200.99 dstport=443 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 eventsubtype="certificate-anomaly" **msg="SSL connection is blocked, certificate-status: expired." hostname="invalid.fortinet.com"**	eventtime=1581015771995913532 tz="-0800" action="blocked" policyid=1 sessionid=96269 service="SMTPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=57522 dstip=172.16.200.99 dstport=465 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 eventsubtype="certificate-anomaly" **msg="SSL connection is blocked, certificate-status: expired untrusted validation_failure." hostname="invalid.fortinet.com"**

To log SSL Exemptions based on FortiGuard categories:

```
config firewall ssl-ssh-profile
   edit "deep-inspection-clone"
      set ssl-exemptions-log enable
   next
end
```

In the HTTPS and SMTPS version of the traffic and ssl utm logs:

- The `logid` and the *msg* are the same.
- A server certificate CN is added to the log.



FortiGate records the wrong category ID and description in the HTTPS version of the ssl utm log. This is a known issue.

Log type	HTTPS	SMTPS
Traffic log	8: date=2020-02-06 time=15:46:10 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581032769970002679 tz="-0800" srcip=10.1.100.66 srcport=57116 srcintf="port2" srcintfrole="undefined" dstip=52.52.208.2 dstport=443 dstintf="port3"	1: date=2020-02-07 time=10:39:20 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581100760770003429 tz="-0800" srcip=10.1.100.66 srcport=42638 srcintf="port2" srcintfrole="undefined" dstip=74.125.195.109 dstport=465 dstintf="port3"

Log type	HTTPS	SMTPS
	dstintfrole="undefined" srccountry="Reserved" dstcountry="United States" sessionid=107685 proto=6 action="close" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="HTTPS" trandisp="snat" transip=172.16.200.7 transport=57116 duration=1 sentbyte=1925 rcvdbyte=7736 sentpkt=13 rcvdpkt=13 appcat="unscanned" wanin=0 wanout=0 lanin=1241 lanout=1241 utmaction="allow" countssl=1 utmref=65476-42	dstintfrole="undefined" srccountry="Reserved" dstcountry="United States" sessionid=139840 proto=6 action="close" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="SMTPS" trandisp="snat" transip=172.16.200.7 transport=42638 duration=1 sentbyte=896 rcvdbyte=3392 sentpkt=9 rcvdpkt=7 appcat="unscanned" utmaction="allow" countssl=1 utmref=65470-0
SSL UTM log	1: date=2020-02-06 time=15:46:08 **logid="1701062005"** type="utm" subtype="ssl" **eventtype="ssl-exempt"** level="notice" vd="vdom1" eventtime=1581032768540281919 tz="-0800" action="exempt" policyid=1 sessionid=107685 service="HTTPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=57116 dstip=52.52.208.2 dstport=443 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 eventsubtype="fortiguard-category" **cat=1 catdesc="Drug Abuse" hostname="www.fortinet.com" msg="SSL connection is exempted based on category rating."**	1: date=2020-02-07 time=10:39:19 **logid="1701062005"** type="utm" subtype="ssl" **eventtype="ssl-exempt"** level="notice" vd="vdom1" eventtime=1581100759642872145 tz="-0800" action="exempt" policyid=1 sessionid=139840 service="SMTPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=42638 dstip=74.125.195.109 dstport=465 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 eventsubtype="fortiguard-category" **cat=23 catdesc="Web-based Email" hostname="smtp.gmail.com" msg="SSL connection is exempted based on category rating."**

To log unsupported SSL negotiation:

```
config firewall ssl-ssh-profile
   edit "deep-inspection-clone"
      set ssl-negotiation-log enable
   next
end
```

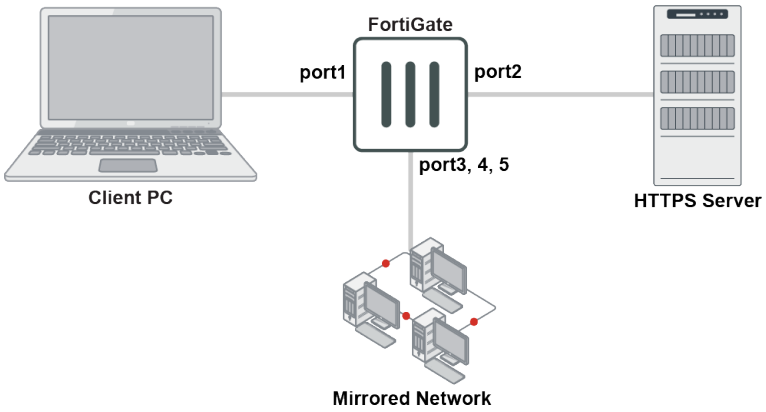
The `logid` and `msg` fields are the same in the HTTPS and IMAPS version of the traffic and ssl utm logs:

Log type	HTTPS	IMAPS
Traffic log	1: date=2020-02-07 time=11:10:59 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581102659640002285 tz="-0800" srcip=10.1.100.66 srcport=33666 srcintf="port2" srcintfrole="undefined" dstip=172.16.200.99 dstport=8080 dstintf="port3" dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved" sessionid=141224 proto=6 action="close" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="tcp/8080" trandisp="snat" transip=172.16.200.7 transport=33666 duration=1 sentbyte=216 rcvdbyte=216 sentpkt=4 rcvdpkt=4 appcat="unscanned" wanin=0 wanout=0 lanin=82 lanout=82 utmaction="block" countssl=1 utmref=65464-0	16: date=2020-02-07 time=11:06:55 **logid="0000000013"** type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1581102415810001699 tz="-0800" srcip=10.1.100.66 srcport=58162 srcintf="port2" srcintfrole="undefined" dstip=172.16.200.99 dstport=8143 dstintf="port3" dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved" sessionid=141051 proto=6 action="close" policyid=1 policytype="policy" poluuid="81d655f2-479f-51ea-d1d1-5fd661144c81" service="tcp/8143" trandisp="snat" transip=172.16.200.7 transport=58162 duration=5 sentbyte=216 rcvdbyte=164 sentpkt=4 rcvdpkt=3 appcat="unscanned" utmaction="block" countssl=1 utmref=65467-0
SSL UTM log	1: date=2020-02-07 time=11:10:58 **logid="1702062101"** type="utm" subtype="ssl" **eventtype="ssl-negotiation"** level="warning" vd="vdom1" eventtime=1581102658589415731 tz="-0800" action="blocked" policyid=1 sessionid=141224 service="HTTPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=33666 dstip=172.16.200.99 dstport=8080 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 **eventsubtype="unexpected-protocol" msg="SSL connection is blocked."**	1: date=2020-02-07 time=11:06:50 **logid="1702062101"** type="utm" subtype="ssl" **eventtype="ssl-negotiation"** level="warning" vd="vdom1" eventtime=1581102410702684472 tz="-0800" action="blocked" policyid=1 sessionid=141051 service="IMAPS" profile="deep-inspection-clone" srcip=10.1.100.66 srcport=58162 dstip=172.16.200.99 dstport=8143 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 **eventsubtype="unexpected-protocol" msg="SSL connection is blocked."**

Support SSL mirroring in proxy mode

SSL mirroring allows the FortiGate to decrypt and mirror traffic to a designated port. Previously, this was supported in flow mode. Support for proxy mode has been added. A new decrypted traffic mirror profile can be applied to IPv4, IPv6, and explicit proxy firewall policies. Full SSL inspection must be used in the policy for the traffic mirroring to occur.

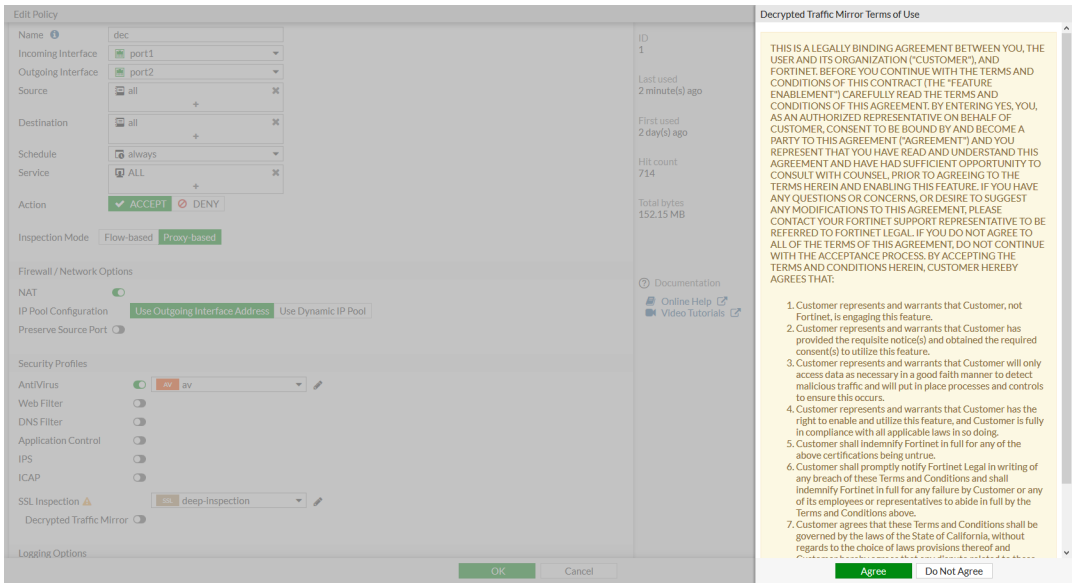
When upgrading to FortiOS 6.4.0, the `original ssl-mirror` and `ssl-mirror-intf` profiles will be replaced with a new `firewall decrypted-traffic-mirror` profile named `__upg_pol_<#>`. The default destination MAC is all FF, and the default source is client.



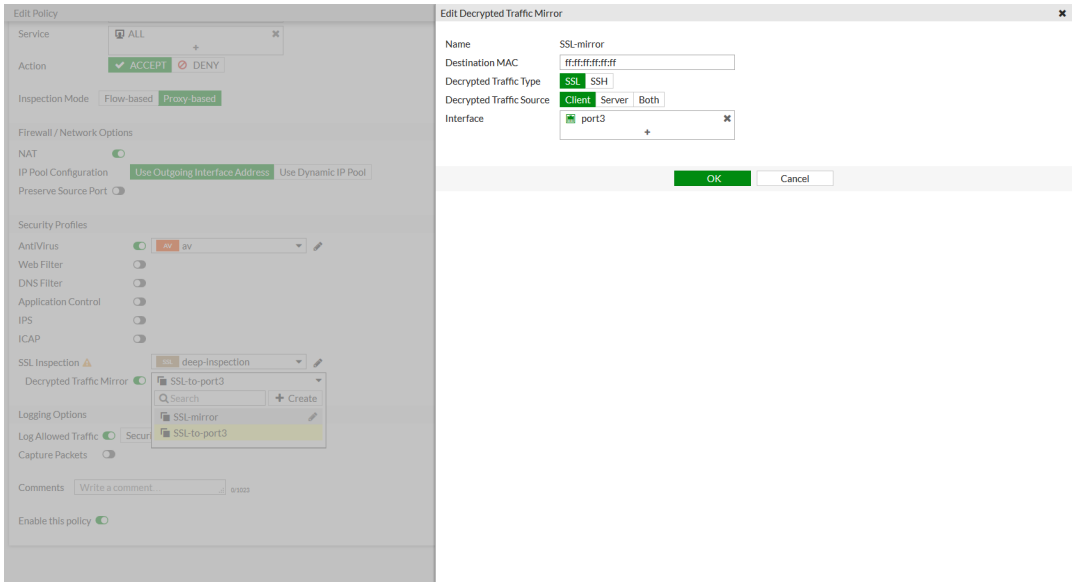
To configure SSL mirroring in proxy mode in the GUI:

1. Go to *Policy & Objects* and create a new policy, or edit an existing one. This example uses a firewall policy.
2. In the policy settings, ensure the following are configured:
   a. The *Inspection Mode* is set to *Proxy-based*.
   b. The *SSL Inspection* profile uses *Full SSL Inspection* (if needed, click the pencil icon next to the dropdown to view the inspection profile settings).
3. Enable the *Decrypted Traffic Mirror* toggle. The terms of use will appear in a separate pane.

4. Click *Agree*.



5. Beside the toggle, click *Create* to configure a new decrypted traffic mirror and adjust the settings as needed. In this example, the client is the decryted traffic source and port3 is the interface.
6. Click *OK* to save the traffic mirror settings.



7. Click *OK* to save the policy settings.

To configure SSL mirroring in proxy mode in the CLI:

1. Create the decrypted traffic mirror profile:

```
config firewall decrypted-traffic-mirror
    edit SSL-to-port3
        set dstmac ff:ff:ff:ff:ff:ff
        set traffic-type ssl
        set traffic-source client
        set interface port3
```

```
    next
end
```

2. Configure the policy to enable SSL traffic mirroring:

```
config firewall policy
    edit 1
        set inspection-mode proxy
        set ssl-ssh-profile deep-inspection
        set decrypted-traffic-mirror SSL-to-port3

        THIS IS A LEGALLY BINDING AGREEMENT BETWEEN YOU, THE USER AND ITS ORGANIZATION
("CUSTOMER"), AND FORTINET. BEFORE YOU CONTINUE WITH THE TERMS AND CONDITIONS OF THIS
CONTRACT (THE "FEATURE ENABLEMENT") CAREFULLY READ THE TERMS AND CONDITIONS OF THIS
AGREEMENT. BY ENTERING YES, YOU, AS AN AUTHORIZED REPRESENTATIVE ON BEHALF OF CUSTOMER,
CONSENT TO BE BOUND BY AND BECOME A PARTY TO THIS AGREEMENT ("AGREEMENT") AND YOU REPRESENT
THAT YOU HAVE READ AND UNDERSTAND THIS AGREEMENT AND HAVE HAD SUFFICIENT OPPORTUNITY TO
CONSULT WITH COUNSEL, PRIOR TO AGREEING TO THE TERMS HEREIN AND ENABLING THIS FEATURE. IF
YOU HAVE ANY QUESTIONS OR CONCERNS, OR DESIRE TO SUGGEST ANY MODIFICATIONS TO THIS
AGREEMENT, PLEASE CONTACT YOUR FORTINET SUPPORT REPRESENTATIVE TO BE REFERRED TO FORTINET
LEGAL. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, DO NOT CONTINUE WITH THE
ACCEPTANCE PROCESS. BY ACCEPTING THE TERMS AND CONDITIONS HEREIN, CUSTOMER HEREBY AGREES
THAT:

        1. Customer represents and warrants that Customer, not Fortinet, is engaging this
feature.

        2. Customer represents and warrants that Customer has provided the requisite notice
(s) and obtained the required consent(s) to utilize this feature.

        3. Customer represents and warrants that Customer will only access data as
necessary in a good faith manner to detect malicious traffic and will put in place
processes and controls to ensure this occurs.

        4. Customer represents and warrants that Customer has the right to enable and
utilize this feature, and Customer is fully in compliance with all applicable laws in so
doing.

        5. Customer shall indemnify Fortinet in full for any of the above certifications
being untrue.

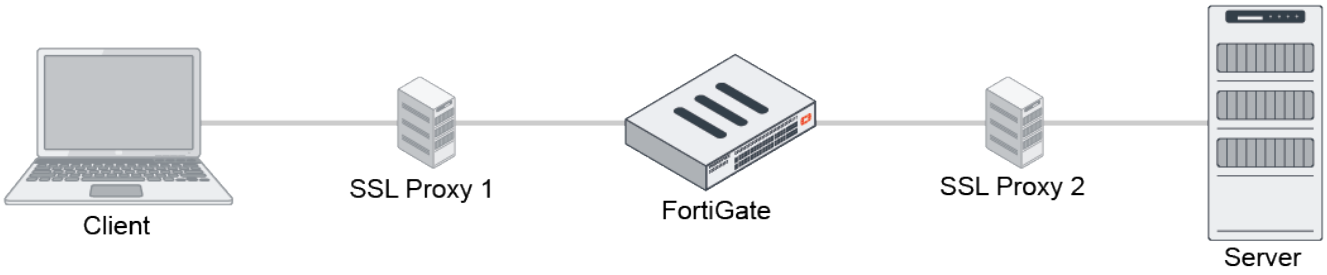
        6. Customer shall promptly notify Fortinet Legal in writing of any breach of these
Terms and Conditions and shall indemnify Fortinet in full for any failure by Customer or
any of its employees or representatives to abide in full by the Terms and Conditions above.

        7. Customer agrees that these Terms and Conditions shall be governed by the laws of
the State of California, without regards to the choice of laws provisions thereof and
Customer hereby agrees that any dispute related to these Terms and Conditions shall be
resolved in Santa Clara County, California, USA, and Customer hereby consents to personal
jurisdiction in Santa Clara County, California, USA.

        Do you want to continue? (y/n)y
    next
end
```

SSL-based application detection over decrypted traffic in a sandwich topology

When a FortiGate is sandwiched between SSL encryption and decryption devices, the FortiGate can process the decrypted traffic that passes between those devices. This feature adds support for decrypted traffic in application control. In some pre-defined signatures, the signature is pre-marked with the *require_ssl_di* tag. The `force-inclusion-ssl-di-sigs` option under `application list` allows users to control the inspection of dissected traffic. When this option is enabled, the IPS engine forces the pre-marked SSL-based signatures to be applied to the decrypted traffic of the respective applications. In the following topology, SSL Proxy 1 handles the client connection and SSL Proxy 2 handles the server connection, leaving the content unencrypted as traffic passes through the FortiGate.



To configure SSL-based application detection over decrypted traffic:

```
config application list
    edit "test"
        set force-inclusion-ssl-di-sigs {enable | disable}
    next
end
```

Example pre-marked SSL-based signature:

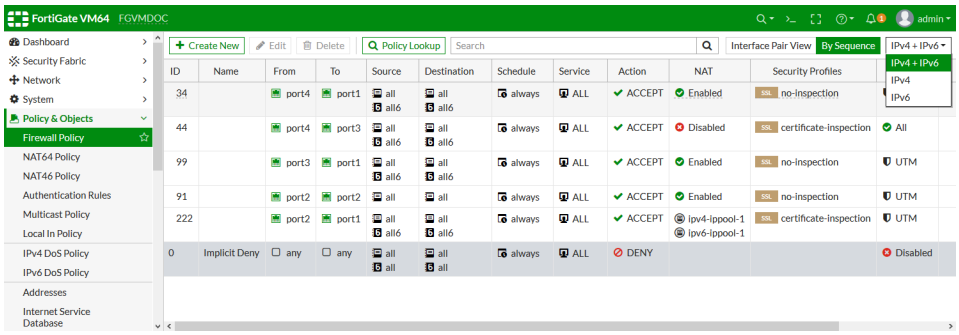
F-SBID(--vuln_id 15722; --attack_id 42985; --name "Facebook_Chat"; --group im; --protocol tcp; --default_action pass; --revision 4446; --app_cat 23; --vendor 3; --technology 1; --behavior 9; --pop 4; --risk 2; --language "Multiple"; --weight 20; --depend-on 15832; --depend-on 38468; ***--require_ssl_di "Yes"****; --casi 1; --casi 8; --parent 15832; --app_port "TCP/443"; --severity info; --status hidden; --service http; --flow from_client; --pattern "/pull?"; --context uri; --no_case; --pattern ".facebook.com"; --context host; --no_case; --tag set,Tag.Facebook.Pull; --tag quiet; --scan-range 10m,all; --date 20190301;)*

All signatures that include the *require_ssl_di* tag are pre-defined and cannot be customized.

Consolidated IPv4 and IPv6 policy configuration

IPv4 and IPv6 policy configuration are consolidated in both NGFW profile-based and NGFW policy-based modes. When creating a policy, both IPv4 and IPv6 addresses can be added as sources and destinations.

The IP version of the sources and destinations in a policy must match. For example, a policy cannot have only an IPv4 source and an IPv6 destination.



The policy list can be filtered to show policies with IPv4, IPv6, or IPv4 and IPv6 sources and destinations.

When upgrading from FortiOS 6.2.3 to 6.4.0:

- In NGFW profile-based mode, IPv4 and IPv6 policies will all be added to the Firewall Policy list, with IPv6 policies listed after IPv4 policies. If consolidated policy mode is enabled, consolidated policies will be changed to firewall policies.
- In NGFW policy-based mode, policies will be changed from consolidated policies to firewall policies in the CLI.
- The `config firewall policy6` and `config firewall consolidated policy` commands, and the `consolidated-firewall-mode` variable in the `config system settings` command, are all removed.



By default, IPv6 options are not visible. See Feature visibility for instructions on making them visible.

NGFW Profile-based mode

To configure an IPv4 and IPv6 firewall policy in the CLI:

```
config firewall policy
    edit 99
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all6"
        set dstaddr6 "all6"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
        set ippool enable
        set poolname "ipv4-ippool-1"
        set poolname6 "ipv6-ippool-1"
    next
end
```

To check the iprope lists for the policy:

```
# diagnose firewall iprope list 100004
policy index=99 uuid_idx=56 action=accept
```

```
flag (8050108): redir nat master use_src pol_stats
flag2 (4000): resolve_sso
flag3 (20): link-local
schedule(always)
cos_fwd=255  cos_rev=255
group=00100004 av=00004e20 au=00000000 split=00000000
host=1 chk_client_info=0x0 app_list=0 ips_view=0
misc=0
zone(1): 11 -> zone(1): 9
source(1): 0.0.0.0-255.255.255.255, uuid_idx=21,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=21,
service(1):
        [0:0x0:0/(0,65535)->(0,65535)] helper:auto
nat(1): flag=1 base=0.0.0.0:0 2.2.2.30-2.2.2.40(0:0)
```

```
# diagnose firewall iprope6 list 100004
policy id: 99, group: 00100004, uuid_idx=56
  action: accept, schedule: always
  cos_fwd=255 cos_rev=255
  flag (08050108): redir nat master use_src pol_stats
  flag2(00004000): resolve_sso
  shapers: / per_ip=
  sub_groups: av 00004e20 auth 00000000 split 00000000 misc 00000000
  app_list: 0 ips_view: 0
  vdom_id: 1
  zone_from(1): 11
  zone_to(1): 9
  address_src(1):
      all uuid_idx=40
  address_dst(1):
      all uuid_idx=40
  service(1):
      [0:0x0:0/(0,65535)->(0,65535)] helper:auto
  nat(1):
      flag=1 base=::(:0)
      2003::2003 - 2003::2004(0:0)
```

NGFW Policy-based mode

To configure an IPv4 and IPv6 SSL Inspection & Authentication policy in the CLI:

```
config firewall policy
    edit 2
        set srcintf "port24"
        set dstintf "port17"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set service "ALL"
        set auto-asic-offload disable
    next
end
```

To configure an IPv4 and IPv6 security policy in the CLI:

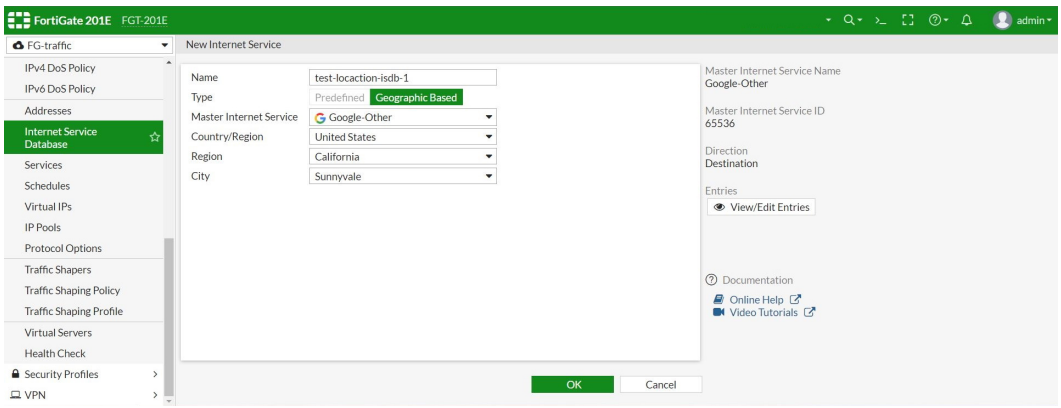
```
config firewall security-policy
    edit 1
        set comments "test"
        set srcintf "port24"
        set dstintf "port17"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set enforce-default-app-port disable
        set service "ALL"
        set action accept
        set schedule "always"
        set logtraffic all
    next
end
```

Allow creation of ISDB objects with regional information

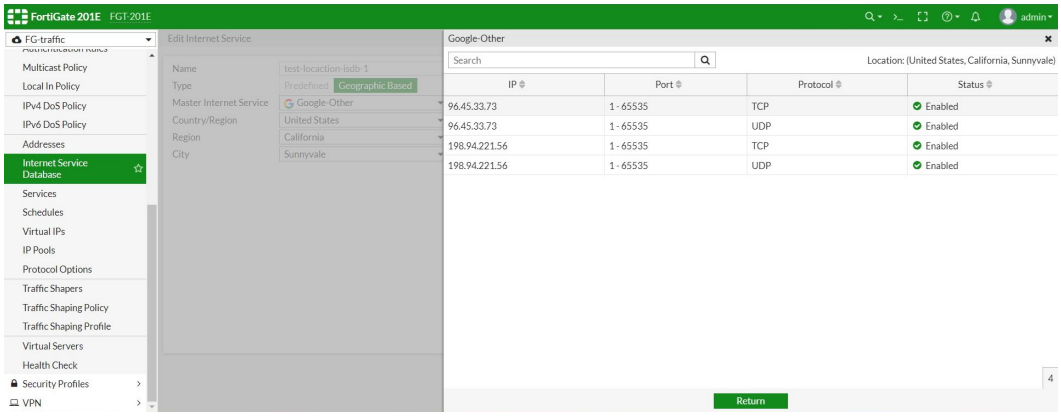
Geographic-based Internet Service Database (ISDB) objects allow users to define a country, region, and city. These objects can be used in firewall policies for more granular control over the location of the parent ISDB object. ISDB objects are now referenced in policies by name instead of ID.

To apply a location-based ISDB object to a policy in the GUI:

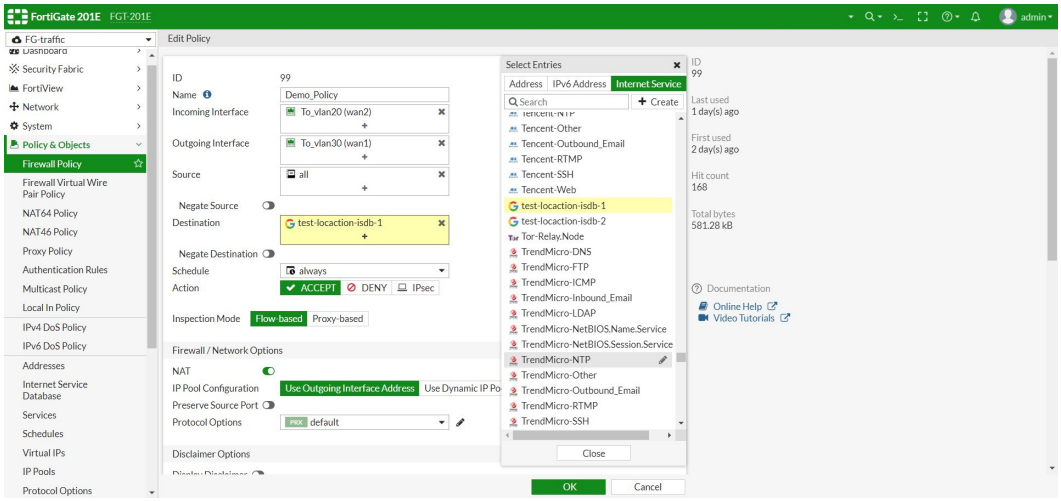
1. Create the ISDB object:
   a. Go to *Policy & Objects* > *Internet Service Database* > *Create New*.
   b. For *Type*, select *Geographic Based*, and configure the other settings as needed.
   c. Click *OK*.



2. View the IP ranges in the location-based internet service:
   a. Go to *Policy & Objects* > *Internet Service Database* .
   b. In the table, hover over the object created in step 1 and click *View/Edit Entries*. The list of IPs is displayed:



   c. Click *Return*.
3. Add the ISDB object to a policy:
   a. Go to *Policy & Objects* > *Firewall Policy*. Create a new policy or edit an existing policy.
   b. For *Destination*, click *Internet Service* and select the ISDB object created in step 1.
   c. Configure the other settings as needed.



   d. Click *OK*.

To apply a location-based ISDB object to a policy in the CLI:

1. Create the ISDB object:

```
config firewall internet-service-name
    edit "test-locaction-isdb-1"
        set type location
        set internet-service-id 65536
        set country-id 840
        set region-id 283
        set city-id 23352
    next
end
```

2. View the IP ranges in the location-based internet service:

```
# diagnose internet-service id 65536 | grep "country(840) region(283) city(23352)"
96.45.33.73-96.45.33.73 country(840) region(283) city(23352) blacklist(0x0) reputation(4),
domain(5) popularity(0) botnet(0) proto(6) port(1-65535)
96.45.33.73-96.45.33.73 country(840) region(283) city(23352) blacklist(0x0) reputation(4),
domain(5) popularity(0) botnet(0) proto(17) port(1-65535)
198.94.221.56-198.94.221.56 country(840) region(283) city(23352) blacklist(0x0) reputation
(4), domain(5) popularity(4) botnet(0) proto(6) port(1-65535)
198.94.221.56-198.94.221.56 country(840) region(283) city(23352) blacklist(0x0) reputation
(4), domain(5) popularity(4) botnet(0) proto(17) port(1-65535)
```

3. Add the ISDB object to a policy:

```
config firewall policy
    edit 99
        set name "Demo_Policy"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set internet-service enable
        set internet-service-name "test-locaction-isdb-1"
        set action accept
        set schedule "always"
        set logtraffic all
        set logtraffic-start enable
        set auto-asic-offload disable
        set comments "1"
        set nat enable
    next
end
```

Force HA failover for testing and demonstrations

HA failover can be forced on an HA master device. The device will stay in a failover state regardless of the conditions. The only way to remove the failover status is by manually turning it off.



This command should only be used for testing, troubleshooting, and demonstrations. Do not use it in a production environment.

Syntax

```
execute ha failover set <custer_id>
execute ha failover unset <custer_id>
```

Variable	Description
<custer_id>	The cluster ID is 1 for any cluster that is not is virtual cluster mode, and can be 1 or 2 if virtual cluster mode is enabled.

Example

To manually force an HA failover:

```
# execute ha failover set 1
Caution: This command will trigger an HA failover.
It is intended for testing purposes.
Do you want to continue? (y/n)y
```

To view the failover status:

```
# execute ha failover status
failover status: set
```

To view the system status of a device in forced HA failover:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-300D
Mode: HA A-P
Group: 240
Debug: 0
Cluster Uptime: 0 days 2:11:46
Cluster state change time: 2020-03-12 17:38:04
Master selected using:
    <2020/03/12 17:38:04> FGT3HD3914800153 is selected as the master because it has EXE_FAIL_
OVER flag set.
    <2020/03/12 15:27:26> FGT3HD3914800069 is selected as the master because it has the
largest value of override priority.
ses_pickup: disable
override: enable
Configuration Status:
    FGT3HD3914800069(updated 4 seconds ago): in-sync
    FGT3HD3914800153(updated 3 seconds ago): in-sync
System Usage stats:
    FGT3HD3914800069(updated 4 seconds ago):
        sessions=5, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
    FGT3HD3914800153(updated 3 seconds ago):
        sessions=41, average-cpu-user/nice/system/idle=0%/0%/0%/99%, memory=30%
HBDEV stats:
    FGT3HD3914800069(updated 4 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=15914162/42929/0/0,
tx=15681840/39505/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=17670346/52854/0/0,
tx=20198409/54692/0/0
    FGT3HD3914800153(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=16636700/45544/0/0,
tx=15529791/39512/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=20199928/54699/0/0,
tx=17672146/52862/0/0
Slave : FortiGate-300D  , FGT3HD3914800069, HA cluster index = 1
Master: FortiGate-300D  , FGT3HD3914800153, HA cluster index = 0
number of vcluster: 1
vcluster 1: standby 169.254.0.1
```

```
Slave : FGT3HD3914800069, HA operating index = 1
Master: FGT3HD3914800153, HA operating index = 0
```

To stop the failover status:

```
# execute ha failover unset 1
Caution: This command may trigger an HA failover.
It is intended for testing purposes.
Do you want to continue? (y/n)y
```

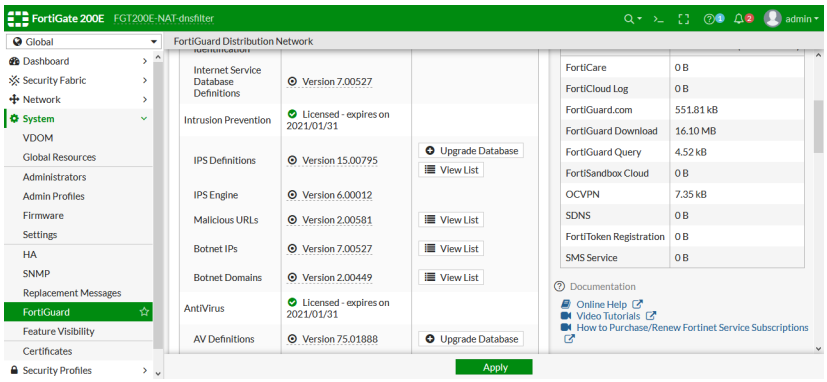
To view the system status of a device after forced HA failover is disabled:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-300D
Mode: HA A-P
Group: 240
Debug: 0
Cluster Uptime: 0 days 2:14:55
Cluster state change time: 2020-03-12 17:42:17
Master selected using:
    <2020/03/12 17:42:17> FGT3HD3914800069 is selected as the master because it has the
largest value of override priority.
    <2020/03/12 17:38:04> FGT3HD3914800153 is selected as the master because it has EXE_FAIL_
OVER flag set.
    <2020/03/12 15:27:26> FGT3HD3914800069 is selected as the master because it has the
largest value of override priority.
ses_pickup: disable
override: enable
Configuration Status:
    FGT3HD3914800069(updated 3 seconds ago): in-sync
    FGT3HD3914800153(updated 2 seconds ago): in-sync
System Usage stats:
    FGT3HD3914800069(updated 3 seconds ago):
        sessions=0, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
    FGT3HD3914800153(updated 2 seconds ago):
        sessions=38, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
HBDEV stats:
    FGT3HD3914800069(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=16302442/43964/0/0,
tx=16053848/40454/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=18161941/54088/0/0,
tx=20615650/55877/0/0
    FGT3HD3914800153(updated 2 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=17033009/46641/0/0,
tx=15907891/40462/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=20617180/55881/0/0,
tx=18163135/54091/0/0
Master: FortiGate-300D  , FGT3HD3914800069, HA cluster index = 1
Slave : FortiGate-300D  , FGT3HD3914800153, HA cluster index = 0
number of vcluster: 1
vcluster 1: work 169.254.0.2
Master: FGT3HD3914800069, HA operating index = 0
Slave : FGT3HD3914800153, HA operating index = 1
```

IP definitions database merged into the internet service database

The IP definitions database (IBDB, previously known as the IRDB) is merged into the internet service database (ISDB, also known as FFDB). Botnet C&C IP blocking now uses the ISDB as a source.

In the *License Information* table at *System* > *FortiGuard*, *Botnet IPs* and *Internet Service Database Definitions* have the same database version.



Updating object versions

When updating object versions in the CLI, Botnet IPs is not listed. Internet-service Database Apps and Internet-service Database Maps are listed, and show the version for *Botnet IPs* and *Internet Service Database Definitions*.

```
# diagnose autoupdate version

......

Internet-service Database Apps
---------
Version: 7.00528
Contract Expiry Date: n/a
Last Updated using scheduled update on Fri Mar 13 12:48:18 2020
Last Update Attempt: Fri Mar 13 16:48:10 2020
Result: No Updates

Internet-service Database Maps
---------
Version: 7.00528
Contract Expiry Date: n/a
Last Updated using scheduled update on Fri Mar 13 12:48:18 2020
Last Update Attempt: Fri Mar 13 16:48:10 2020
Result: No Updates
......
```

Update debug messages

In FortiOS 6.4 update debug messages, there is no query for the IBDB object:

6.4.0:

```
pack_obj[196]-Packing obj=Protocol=3.2|Command=Update|Firmware=FG200E-FW-6.04-1565|Seri-
alNumber=FG200E4Q17900126|UpdateMethod=0|AcceptDelta=1|DataItem=06004000APDB00105-00015.00795-
```

```
2003120019*06004000AVDB00201-00075.01892-2003131320*06004000AVDB00701-00075.01892-
2003131320*06004000MMDB00101-00075.01916-2003131321*06004000FLDB00201-00075.01893-
2003131325*06004000DBDB00100-00002.00450-2003131322*06004000NIDS02505-00015.00795-
2003120019*06004000ISDB00105-00000.00000-0101010000*06004000MUDB00103-00002.00581-
2003130417*06004000CIDB00000-00001.00096-
2003131527*06004000IPGO000000304920031221l1*00000000FCNI00000-00000.00000-
0000000000*00000000FDNI00000-00000.00000-0000000000*01000000FSCI00100-00000.00000-
0000000000*06004000AVEN02800-00006.00144-2002220146*06004000FLEN06700-00006.00012-
2003110118*06004000FLEN05000-00001.00009-1906061402*06004000FFDB00307-00007.00528-
2003131142*06004000FFDB00407-00007.00528-2003131142*06004000UWDB00100-00002.00709-
2003131105*06004000CRDB00000-00001.00015-1907031016*06004000SFAS00000-00003.00000-
2002130915*06004000MCDB00100-00001.00254-2003091200*02000000FNSD00000-00000.00008-0000000000
```

6.2.3:

```
pack_obj[192]-Packing obj=Protocol=3.2|Command=Update|Firmware=FG200E-FW-6.02-1093|Seri-
alNumber=FG200E4Q17904482|UpdateMethod=0|AcceptDelta=1|DataItem=06002000APDB00104-00015.00795-
2003120019*06002000AVDB00201-00075.02861-2003120945*06002000MMDB00101-00075.01920-
2003131421*06002000IBDB00101-00004.00634-2003111709*06002000DBDB00100-00002.00450-
2003131322*06002000NIDS02504-00015.00795-2003120019*06002000ISDB00104-00015.00795-
2003120019*06002000MUDB00103-00002.00581-2003130417*06002000CIDB00000-00001.00097-
2003091749*06002000IPGO000000304920031221l1*00000000FCNI00000-00000.00000-
0000000000*00000000FDNI00000-00000.00000-0000000000*01000000FSCI00100-00000.00000-
0000000000*06002000AVEN02800-00006.00144-2002220146*06002000FLEN07300-00005.00203-
2002242346*06002000FLEN05000-00001.00009-1906061402*06002000FFDB00306-00007.00528-
2003131137*06002000FFDB00406-00007.00528-2003131137*06002000UWDB00100-00002.00709-
2003131105*06002000CRDB00000-00001.00015-1907031016*06002000SFAS00000-00002.00033-
1911121935*06002000MCDB00100-0
```

Diagnosing botnet IPs

Botnet IPs can be diagnosed with the following CLI command:

```
# diagnose sys botnet-ip {hit | list | find | flush}
```

Command	Description
`hit`	Show botnet IP entry hit count data.
`list`	List botnet IP entries.
`find <ip> <port> <protocol>`	Find botnet IP entries. Enter the IP address, port number, and protocol number to search the entries.
`flush`	Flush botnet IP entry hit count data.

Security Profiles enhancements

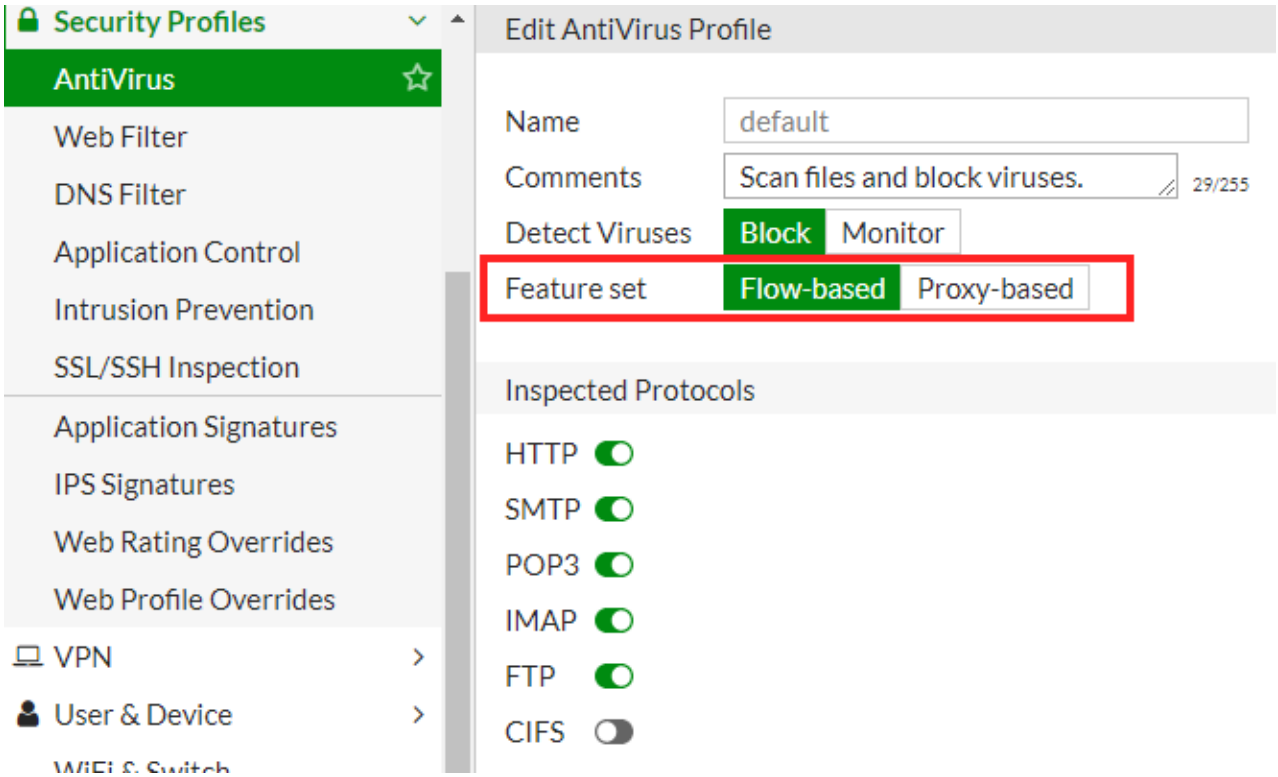
Feature set option

To more clearly show the features specific to proxy-based mode, use the new *Feature set* option to select *Flow-based* or *Proxy-based*. When you select *Flow-based* or *Proxy-based*, only the features for that mode are available.

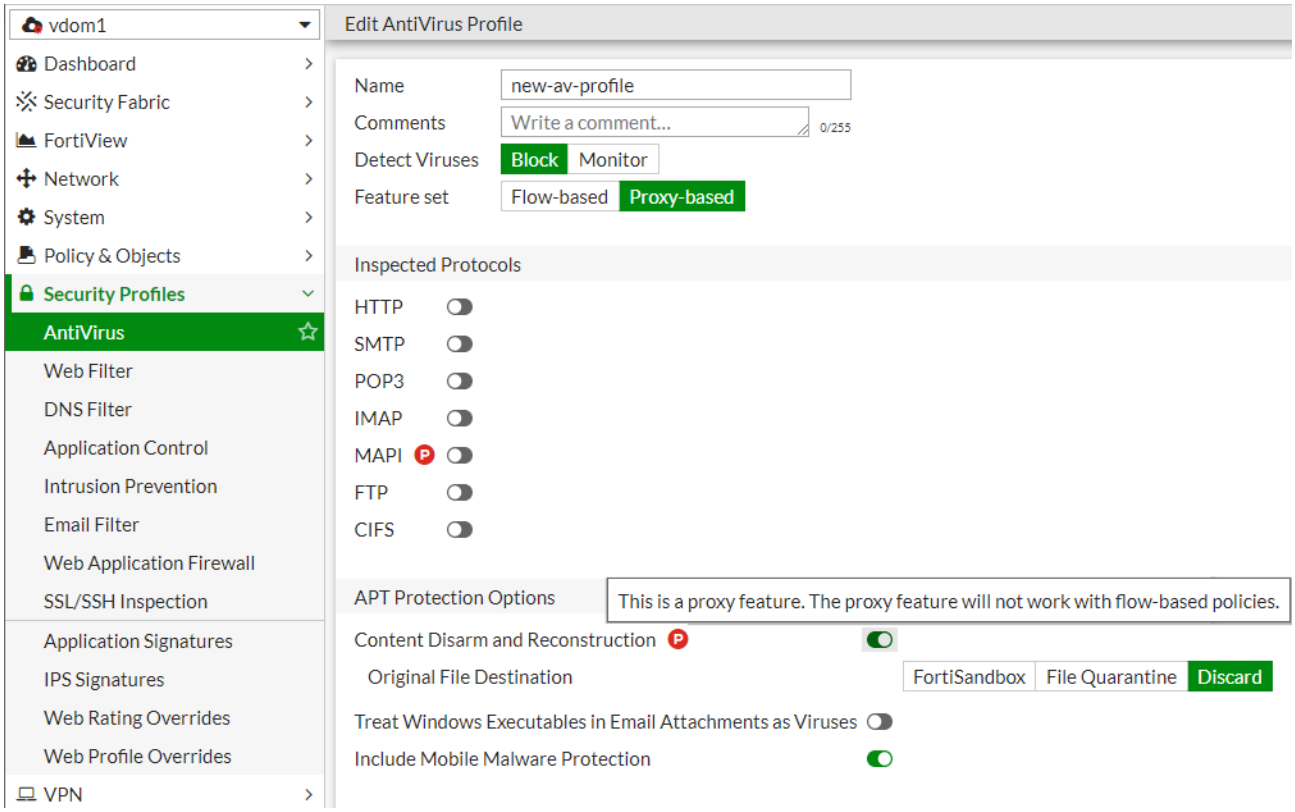
The following pages have the *Feature set* option:

- *Security Profiles* > *AntiVirus*
- *Security Profiles* > *Web Filter*
- *Security Profiles* > *Email Filter*
- *Security Profiles* > *Data Leak* (CLI only)
- *Policy & Objects* > *Protocol Options*

Example of the *Feature set* option in *Security Profiles* > *AntiVirus*:



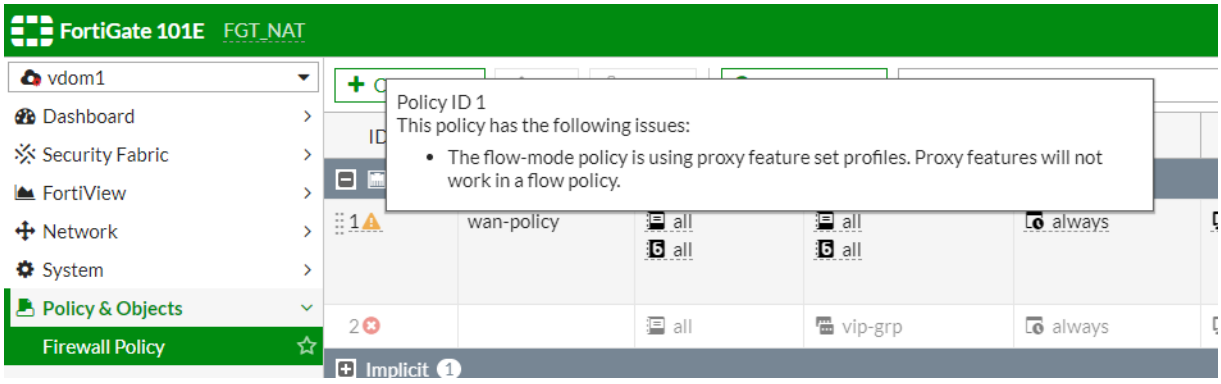
If you select *Proxy-based*, a red P icon indicates the proxy-only features. FortiOS.



When you configure firewall policies:

- If the inspection mode is flow-based, dropdown menus only display profiles with flow-based feature sets.
- If the inspection mode is proxy-based, dropdown menus display profiles with flow-based or proxy-based feature sets.

If a flow-based inspection policy has a proxy-based profile assigned, a warning icon and tooltip informs you that proxy features do not work in a flow-based policy. This warning also appears when you use the CLI to assign security profiles.



Upgrade support

Upgrading from 6.2.x to 6.4.0 causes the following changes to security profiles.

Upgrade scenario	Result after upgrade
Profile was assigned exclusively to flow-base firewall policies in 6.2.x.	feature-set = flow
Profile was assigned exclusively to proxy-base firewall policies in 6.2.x.	feature-set = proxy
Profile was assigned to both flow-base and proxy-base firewall policies in 6.2.x.	feature-set = proxy
Profile was not assigned to any firewall policies in 6.2.x.	feature-set = flow

Configure security profiles using CLI

To configure the AntiVirus security profile using the CLI:

```
FGT_NAT (vdom1) # config antivirus profile
FGT_NAT (profile) # edit new-av-profile

FGT_NAT (new-av-profile) # set ?
comment              Comment.
replacemsg-group     Replacement message group customized for this profile.
feature-set          Flow/proxy feature set.
mobile-malware-db    Enable/disable using the mobile malware signature database.
av-virus-log         Enable/disable AntiVirus logging.
av-block-log         Enable/disable logging for AntiVirus file blocking.
extended-log         Enable/disable extended logging for antivirus.

FGT_NAT (new-av-profile) # set feature-set ?
flow     Flow feature set.
proxy    Proxy feature set.

FGT_NAT (new-av-profile) # set feature-set proxy

FGT_NAT (new-av-profile) # show
config antivirus profile
    edit "new-av-profile"
        set feature-set proxy
    next
end
```

To configure the Web Filter security profile using the CLI:

```
FGT_NAT (vdom1) # config webfilter profile
FGT_NAT (profile) # edit new-wf-profile

FGT_NAT (new-wf-profile) # set ?
comment                         Optional comments.
feature-set                     Flow/proxy feature set.
replacemsg-group                Replacement message group.
options                         Options.
...
```

```
FGT_NAT (new-wf-profile) # set feature-set ?
flow     Flow feature set.
proxy    Proxy feature set.

FGT_NAT (new-wf-profile) # set feature-set proxy

FGT_NAT (new-wf-profile) # show
config webfilter profile
    edit "new-wf-profile"
        set feature-set proxy
        config ftgd-wf
            unset options
            config filters
                ...
            end
        end
    next
end
```

To configure the Email Filter security profile using the CLI:

```
FGT_NAT (vdom1) # config emailfilter profile
FGT_NAT (profile) # edit new-ef-profile

FGT_NAT (new-ef-profile) # set ?
comment                          Comment.
feature-set                      Flow/proxy feature set.
replacemsg-group                 Replacement message group.
spam-log                         Enable/disable spam logging for email filtering.
spam-log-fortiguard-response     Enable/disable logging FortiGuard spam response.
spam-filtering                   Enable/disable spam filtering.
external                         Enable/disable external Email inspection.
options                          Options.
spam-bword-threshold             Spam banned word threshold.
spam-bword-table                 Anti-spam banned word table ID.
spam-bwl-table                   Anti-spam black/white list table ID.
spam-mheader-table               Anti-spam MIME header table ID.
spam-rbl-table                   Anti-spam DNSBL table ID.
spam-iptrust-table               Anti-spam IP trust table ID.

FGT_NAT (new-ef-profile) # set feature-set ?
flow     Flow feature set.
proxy    Proxy feature set.

FGT_NAT (new-ef-profile) # set feature-set proxy

FGT_NAT (new-ef-profile) # show
config emailfilter profile
    edit "new-ef-profile"
        set feature-set proxy
    next
end
```

To configure the DLP security profile using the CLI:

```
FGT_NAT (vdom1) # config dlp sensor
FGT_NAT (sensor) # edit new-dlp-profile

FGT_NAT (new-dlp-profile) # set ?
comment               Comment.
feature-set           Flow/proxy feature set.
replacemsg-group      Replacement message group used by this DLP sensor.
dlp-log               Enable/disable DLP logging.
extended-log          Enable/disable extended logging for data leak prevention.
nac-quar-log          Enable/disable NAC quarantine logging.
full-archive-proto    Protocols to always content archive.
summary-proto         Protocols to always log summary.

FGT_NAT (new-dlp-profile) # set feature-set ?
flow     Flow feature set.
proxy    Proxy feature set.

FGT_NAT (new-dlp-profile) # set feature-set proxy

FGT_NAT (new-dlp-profile) # show
config dlp sensor
    edit "new-dlp-profile"
        set feature-set proxy
    next
end
```

To configure Protocol Options in Policy & Objects using the CLI:

```
FGT_NAT (vdom1) # config firewall profile-protocol-options
FGT_NAT (profile-protocol~ons) # edit new-protocol-options

FGT_NAT (new-protocol-options) # set ?
comment                   Optional comments.
feature-set               Flow/proxy feature set.
replacemsg-group          Name of the replacement message group to be used
oversize-log              Enable/disable logging for antivirus oversize file blocking.
switching-protocols-log   Enable/disable logging for HTTP/HTTPS switching protocols.
rpc-over-http             Enable/disable inspection of RPC over HTTP.

FGT_NAT (new-protocol-options) # set feature-set ?
flow     Flow feature set.
proxy    Proxy feature set.

FGT_NAT (new-protocol-options) # set feature-set proxy

FGT_NAT (new-protocol-options) # show
config firewall profile-protocol-options
    edit "new-protocol-options"
        set feature-set proxy
        config http
            set ports 80
            unset options
            unset post-lang
        end
```

```
        config ftp
            set ports 21
            set options splice
        end
        config imap
            set ports 143
            set options fragmail
        end
        ...
    next
end
```

AntiVirus profiles use hybrid scanning as default

In flow-based AntiVirus profiles, the scan-mode option is removed. Flow-based AntiVirus profiles use the default hybrid scanning method to process traffic. Legacy mode is available for diagnostics only.



When upgrading from 6.2.x to 6.4.0, AntiVirus profiles assigned to flow-based firewall policies only operate in the default hybrid mode regardless of the previous scan-mode setting.

In CLI, `scan-mode` options are only available for proxy-based AntiVirus profiles. The `scan-mode` options are not available for flow-based AntiVirus profiles.

```
config antivirus profile
    edit "new-av-profile"
        set comment ''
        set replacemsg-group ''
        set feature-set proxy
        set mobile-malware-db enable
        config http
            unset options
            unset archive-block
            unset archive-log
            set emulator enable
            set outbreak-prevention disabled
        end
        ...
        set av-virus-log enable
        set av-block-log enable
        set extended-log disable
        set scan-mode default
    next
end

set ?
comment              Comment.
replacemsg-group     Replacement message group customized for this profile.
feature-set          Flow/proxy feature set.
mobile-malware-db    Enable/disable using the mobile malware signature database.
av-virus-log         Enable/disable AntiVirus logging.
av-block-log         Enable/disable logging for AntiVirus file blocking.
extended-log         Enable/disable extended logging for antivirus.
scan-mode            Choose between default scan mode and legacy scan mode.
```

Diagnostics

The following diagnostic commands are meant for troubleshooting only.

```
diagnose ips av mode ?
    hybrid    Enable/disable hybrid scan mode.
    show      Show status of hybrid scan mode.
```

To check flow-base AV scan mode status:

```
diagnose ips av mode show
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
```

To disable hybrid scan for flow-base AV and enable full scan:



This command does not persist over a reboot. `Flow-av hybrid scan` is enabled by default.

```
diagnose ips av mode hybrid disable

diagnose ips av mode show
    Flow-av hybrid scan: Disabled
    Flow-av hybrid scan: Disabled
    Flow-av hybrid scan: Disabled
    Flow-av hybrid scan: Disabled
```

To enable hybrid scan for flow-base AV and disable full scan to go back to default:

```
diagnose ips av mode hybrid enable

diagnose ips av mode show
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
    Flow-av hybrid scan: Enabled
```

AntiVirus uses Extended DB by default

Starting with this version, FortiGate uses Extended DB as its default AntiVirus DB. The Normal DB option is no longer supported. For FortiGate models that support Extreme DB, you have the option to choose Extended DB or Extreme DB.

Under `config antivirus settings`, the `default-db` parameter has been removed.

FortiGate models that support extreme set database have a new `use-extreme-db` parameter.

By default, `use-extreme-db` is disabled so that FortiGate uses its normal and extended set databases. When you enable `use-extreme-db`, FortiGate uses the extreme set database.

Upgrade support

Upgrading from 6.2.x to 6.4.0 causes the following changes.

Before upgrade	After upgrade
default-db = normal	use-extreme-db = disable (hidden on low-end models)
default-db = extended	use-extreme-db = disable (hidden on low-end models)
default-db = extreme	use-extreme-db = enable

AntiVirus settings in the CLI

On low end models, `use-extreme-db` is hidden. This example shows the CLI captured on FGT-101E.

```
FGT_NAT (settings) # show full-configuration
config antivirus settings
    set grayware enable
    set override-timeout 0
end
```

On higher end models, `use-extreme-db` is available. This example shows the CLI captured on FGT-600D.

```
FGT_NAT (settings) # show full-configuration
config antivirus settings
    set use-extreme-db enable
    set grayware enable
    set override-timeout 0
end

FGT_NAT (settings) # set use-extreme-db ?
enable     Enable extreme AVDB.
disable    Disable extreme AVDB.
```

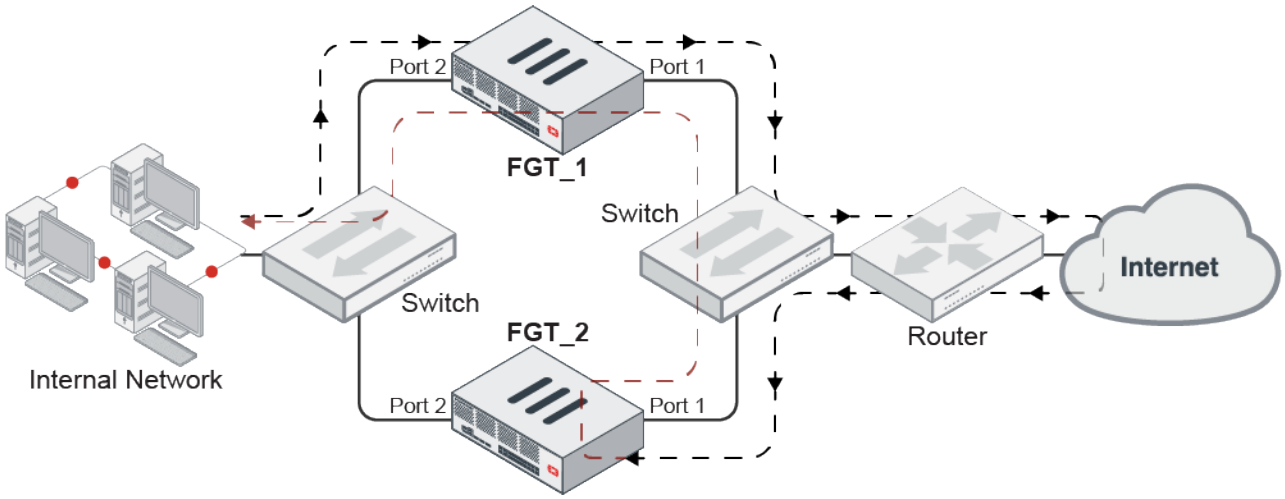
Support UTM inspection on asymmetric traffic in FGSP

When traffic passes asymmetrically through FGSP peers, UTM inspection can be supported by always forwarding traffic back to the session owner for processing. The session owner is the FortiGate that receives the first packet of the session.

In this example, traffic from the internal network first hits FGT_1, but the return traffic is routed to FGT_2. Consequently, traffic bounces from FGT_2 port1 to FGT_1 port1 using FGT_1's MAC address. Traffic is then inspected by FGT_1.

This example requires the following settings:

- Internal and outgoing interfaces of both FortiGates in the FGSP pair are in the same subnet.
- Both peers have layer 2 access with each other.



To configure FTG_1:

1. Configure the cluster, setting the peer IP to the IP address of FGT_2:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.2
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 0
    set layer2-connection available
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

To configure FTG_2:

1. Configure the cluster, setting the peer IP to the IP address of FGT_1:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.1
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 1
    set layer2-connection available
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

Results

Capture packets on FGT_2 to see that traffic bounced from FGT_2 to FGT_1 over the traffic interface.

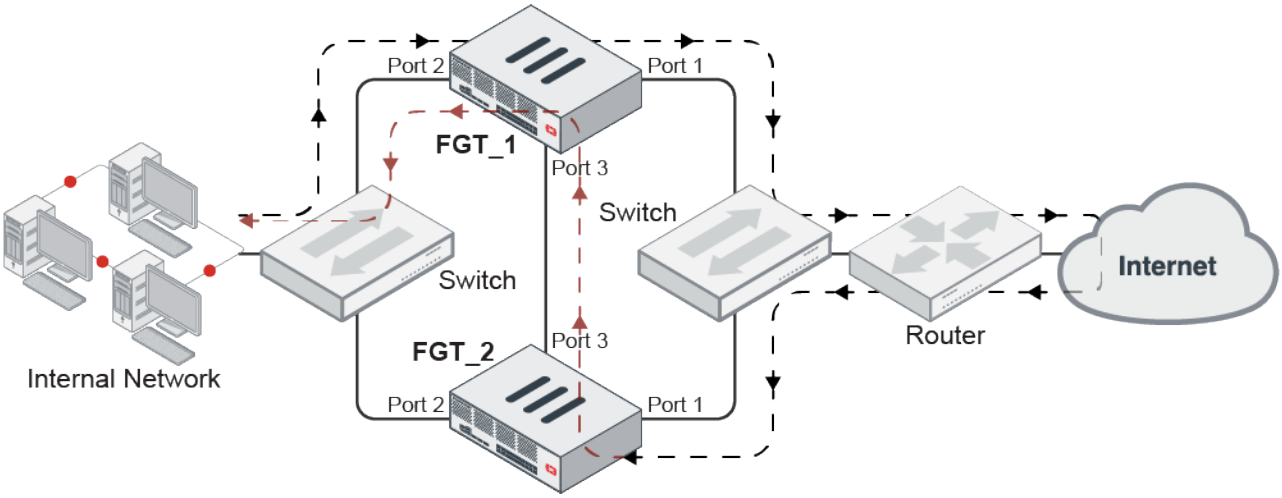
```
FGT_2 # diagnose sniffer packet any 'host 10.1.100.15 and host 172.6.200.55' 4
interfaces=[any]
filters=[host 10.1.100.15 and host 172.16.200.55]
91.803816 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800480 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800486 port1 out 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800816 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800818 port1 out 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
```

Support UTM inspection on asymmetric traffic on L3

When traffic passes asymmetrically through FGSP peers, UTM inspection can be supported by always forwarding traffic back to the session owner for processing. The session owner is the FortiGate that receives the first packet of the session.

For networks where L2 connectivity is not available, such as cloud environments, traffic bound for the session owner are forwarded through the peer interface using a TCP connection.

In this example, traffic from the internal network first hits FGT_1, but the return traffic is routed to FGT_2. Consequently, return traffic is packed and sent from FGT_2 to FGT_1 using UDP encapsulation between two peer interfaces (port 3). Traffic is then inspected by FGT_1.



To configure FTG_1:

1. Configure the cluster, setting the peer IP to the IP address of FGT_2:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.2
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 0
    set layer2-connection unavailable
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
```

```
        set nat enable
    next
end
```

To configure FTG_2:

1. Configure the cluster, setting the peer IP to the IP address of FGT_1:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.1
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 1
    set layer2-connection unavailable
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

Add encryption for L3 on asymmetric traffic in FGSP

In scenarios where asymmetric routing between FGSP members occurs, the return traffic can be routed back to the session owner in Layer 3 (L3). This L3 traffic can now be encrypted.

To encrypt L3 traffic in FGSP:

1. Run the following on both FortiGates:

```
config system standalone-cluster
    set encryption enable
    set psksecret xxxxxxxxx
end
```

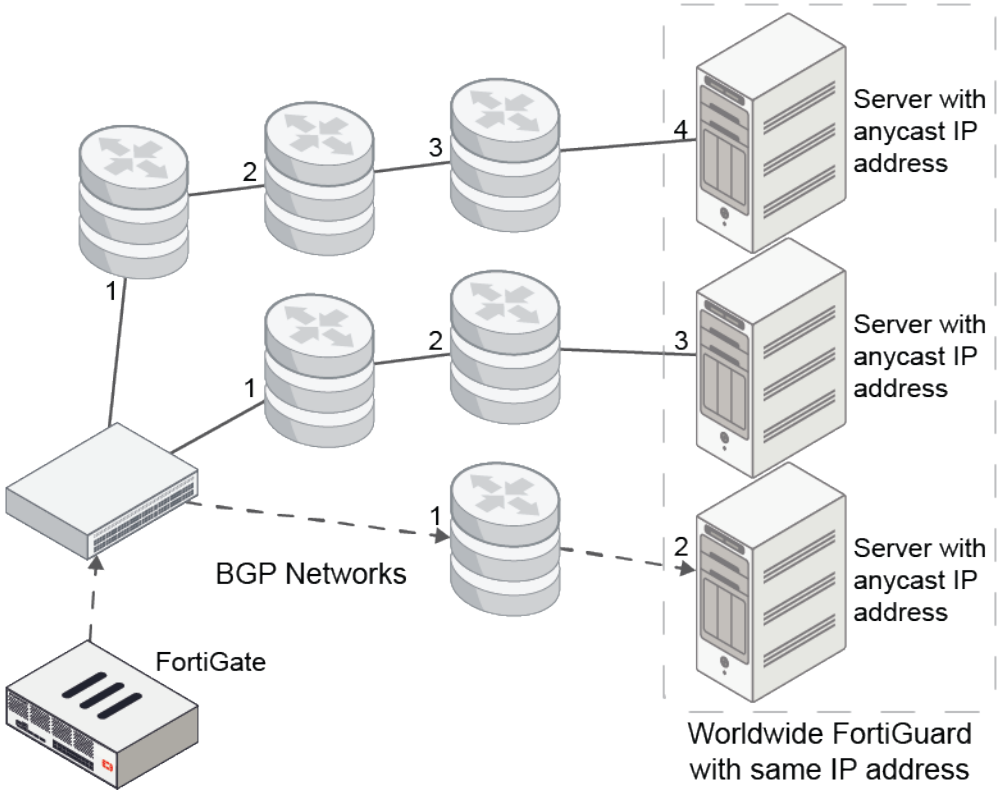
Use anycast to communicate with FortiGuard servers

Third party certificate verification and OCSP stapling check is implemented for all FortiGuard servers that are connected to from FortiOS. The default FortiGuard access mode is anycast.

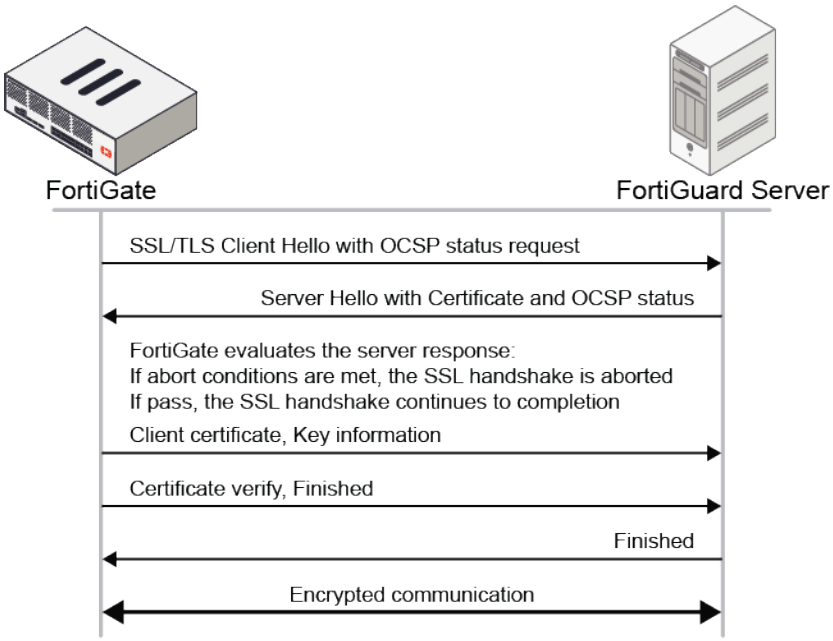
FortiGuard represents all cloud based servers, including those for:

Server	Domain name and IP address
Object download	globalupdate.fortinet.net - 173.243.140.6
Querying service (web-filtering, anti-spam)	globalguardservice.fortinet.net - 173.243.140.16
Querying service (device-query)	
FortiGate Cloud logging	globallogctrl.fortinet.net - 173.243.132.25
FortiGate Cloud management	globalmgrctrl.fortinet.net - 173.243.132.26
FortiGate Cloud messaging	globalmsgctrl.fortinet.net - 173.243.132.27
FortiGate Cloud sandbox	globalaptctrl.fortinet.net - 184.94.112.22
The productapi used by OCVPN registration and GUI icon download	globalproductapi.fortinet.net - 66.35.17.252
FortiCare registration and tokens	
Secure DNS	

The anycast server has one IP address to match its domain name. The FortiGate connects with a single server address, regardless of where the FortiGate is located.



The following process is used to connect to an anycast server:

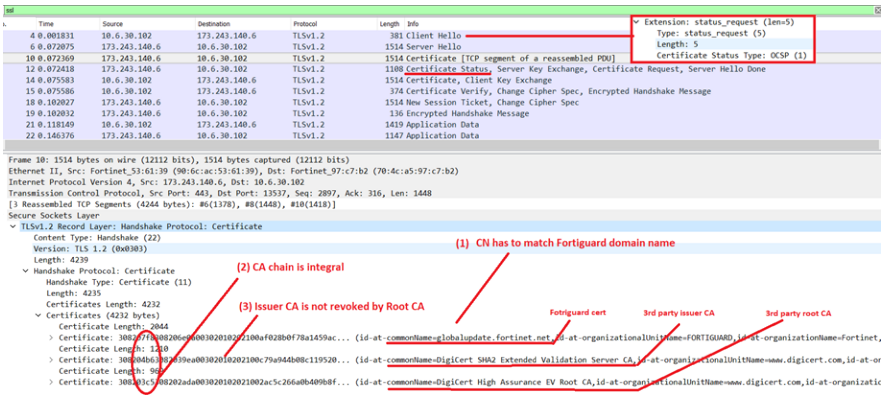


Abort conditions include:

- The CN in the server's certificate does not match the domain name resolved from the DNS.
- The OCSP status is not good.
- The issuer-CA is revoked by the root-CA.

Once the SSL handshake is established, the FortiGate can engage the server.

Example wireshark PCAP:



To enable anycast FortiGuard access mode:

```
config system fortiguard
    set fortigaurd-anycast enable
    set fortiguard-anycast-source fortinet
end
```

SD-WAN

This section includes SD-WAN features added to FortiOS:

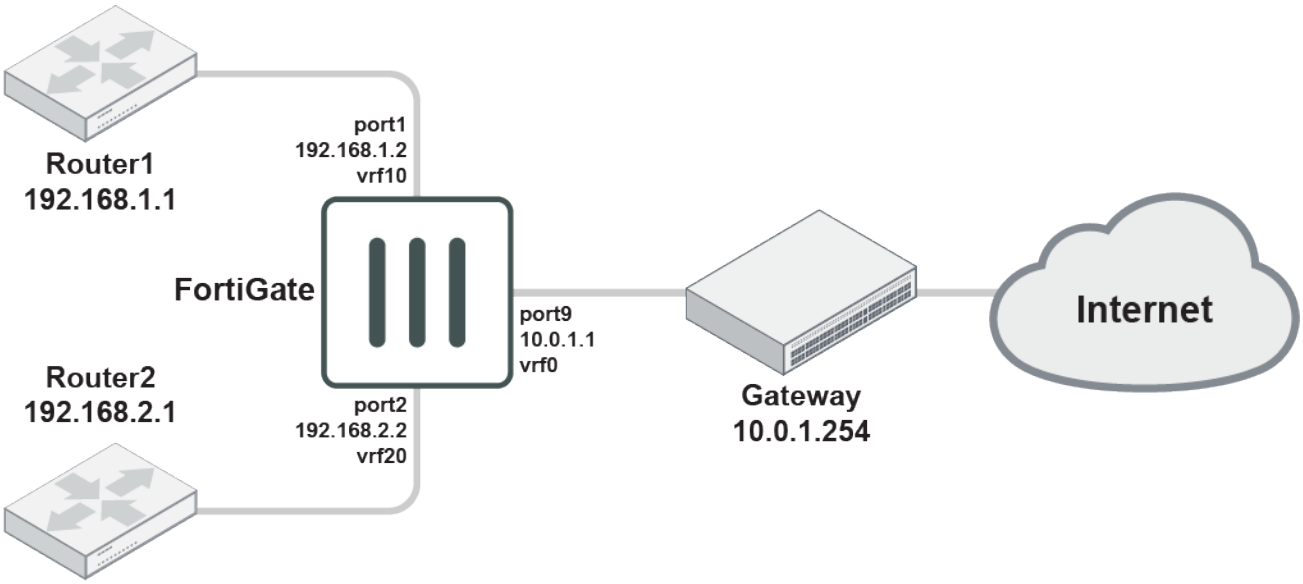
- IBGP and EBGP support in VRF on page 45
- SD-WAN event log subtype on page 47
- SD-WAN logging improvement to identify matched application on page 51
- SD-WAN configuration portability on page 51
- SD-WAN log format improvements on page 54
- SD-WAN monitor on ADVPN shortcuts on page 59
- SD-WAN GUI and monitoring enhancements on page 60
- Enhance ADVPN to support UDP hole punching for spokes behind NAT on page 65
- SD-WAN health check packet enhancement on page 68
- Weighted round robin for IPsec aggregate tunnels on page 68
- Default_DNS performance SLA profile on page 70
- Interface speedtest on page 71
- Support SD-WAN integration with OCVPN on page 73
- Allow FortiClient to join OCVPN on page 81

IBGP and EBGP support in VRF

Support is included for internal and external border gateway protocols (IBGP and EBGP) in virtual routing and forwarding (VRF).

FortiGate can establish neighbor connections with other FortiGates or routers, and the learned routes are put into different VRF tables according to the neighbor's settings.

This example uses the following topology:



- BGP routes learned from the Router1 neighbor are put into vrf10.
- BGP routes learned from the Router2 neighbor are put into vrf20.

To configure this example:

```
config system interface
    edit port1
        set vrf 10
    next
    edit port2
        set vrf 20
    next
end
```

```
config router bgp
    config neighbor
        edit "192.168.1.1"
            set update-source port1
        next
        edit "192.168.2.1"
            set interface port2
        next
    end
end
```

Results

Using the above topology:

- Both Router1 and Router2 establish OSPF and BGP neighbor with the FortiGate.
- Router1 advertises 10.10.1.0/24 into OSPF and 10.10.2.0/24 into BGP.
- Router2 advertises 20.20.1.0/24 into OSPF and 20.20.2.0/24 into BGP.

When port1 and port2 have not set VRF, all of the routing is in VRF=0:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.0.1.254, port9
C       10.0.1.0/24 is directly connected, port9
O       10.10.1.0/24 [110/10] via 192.168.1.1, port1, 00:18:31
B       10.10.2.0/24 [20/200] via 192.168.1.1, port1, 00:01:31
O       20.20.1.0/22 [110/10] via 192.168.2.1, port2, 00:19:05
B       20.20.2.0/24 [20/200] via 192.168.2.1, port2, 00:01:31
C       192.168.1.0/24 is directly connected, port1
C       192.168.2.0/24 is directly connected, port2
```

After VRF is set for BGP, BGP routes are added to the VRF tables along with OSPF and connected routes:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.0.1.254, port9
C       10.0.1.0/24 is directly connected, port9

Routing table for VRF=10
O       10.10.1.0/24 [110/10] via 192.168.1.1, port1, 00:18:31
B       10.10.2.0/24 [20/200] via 192.168.1.1, port1, 00:01:31
C       192.168.1.0/24 is directly connected, port1

Routing table for VRF=20
O       20.20.1.0/22 [110/10] via 192.168.2.1, port2, 00:19:05
B       20.20.2.0/24 [20/200] via 192.168.2.1, port2, 00:01:31
C       192.168.2.0/24 is directly connected, port2
```

BGP neighbor groups

This feature is also supported in the BGP neighbor groups. For example:

```
config router bgp
    config neighbor-group
        edit "FGT"
            set update-source "port1"
        next
    end
    config neighbor-range
        edit 1
            set prefix 172.16.201.0 255.255.255.0
            set neighbor-group "FGT"
        next
    end
end
```

Note that the `set interface` command is not supported.

SD-WAN event log subtype

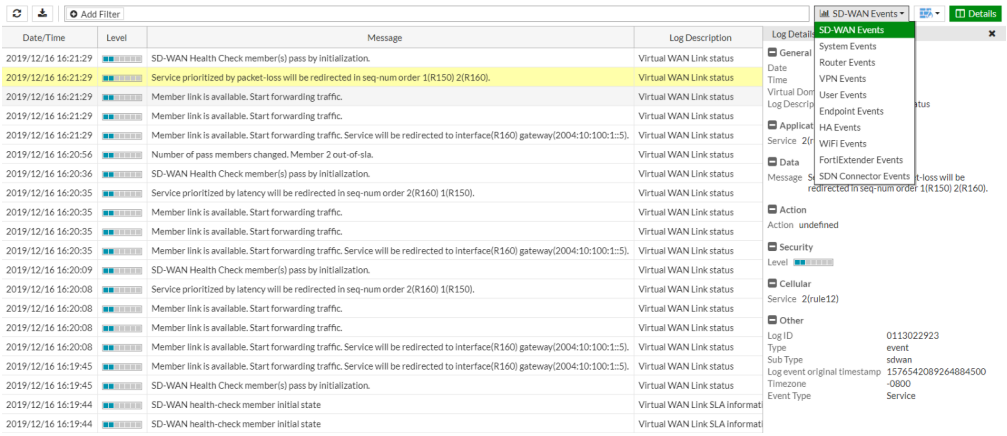
A separate log subtype, *SD-WAN*, has been added to *Event* logs. It consists of seven log IDs:

Log ID	Log description
22923	Virtual WAN link status
22924	Virtual WAN link volume status
22925	Virtual WAN link SLA information
22926	Virtual WAN link neighbor status
22927	Virtual WAN link neighbor standalone
22928	Virtual WAN link neighbor primary
22929	Virtual WAN link neighbor secondary

To filter event logs to show SD-WAN events:

1. Go to *Log & Report* > *Events*.
2. In the toolbar, click the event dropdown button and select *SD-WAN Events*. The filtered list of SD-WAN event logs appears, including the *Log Description*.

3. Select an entry and click the *Details* button to view more information about the log.



Sample SD-WAN event logs

Virtual WAN Link status

Event type = HEALTH CHECK

```
date=2020-03-29 time=16:36:55 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525015062338339 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Health Check" healthcheck="ping1" slatargetid=1 numpassmember=2 msg="SD-WAN Health Check
member(s) pass."
```

```
date=2020-03-29 time=16:41:30 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525290513555981 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Health Check" healthcheck="ping1" slatargetid=1 oldvalue="1" newvalue="2" msg="Number of
pass member changed."
```

```
date=2020-03-29 time=16:41:30 logid="0113022923" type="event" subtype="sdwan" level-
l="information" vd="root" eventtime=1585525290513553153 tz="-0700" logdesc="Virtual WAN Link
status" eventtype="Health Check" healthcheck="ping1" slatargetid=1 member="2" msg="Member
status changed. Member in sla."
```

```
date=2020-03-29 time=16:40:33 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525232970358654 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Health Check" healthcheck="ping1" slatargetid=1 member="2" msg="Member status changed.
Member out-of-sla."
```

Event type = SERVICE

```
date=2020-03-29 time=17:20:13 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585527613995020448 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" serviceid=1 service="" metric="latency" seq="1,2" msg="Service prioritized by
performance metric will be redirected in sequence order."
```

```
date=2020-03-29 time=17:20:13 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585527613995017084 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" interface="R160" member="2" serviceid=1 service="" gateway="10.100.1.5" msg-
g="Member link is available. Start forwarding traffic. "
```

```
date=2020-03-29 time=17:33:25 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585528405170900938 tz="-0700" logdesc="Virtual WAN Link status"
```

```
eventtype="Service" serviceid=1 service="service1" msg="Service disabled caused by no outgoing
path."

date=2020-03-29 time=17:33:25 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585528405170876948 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" serviceid=1 service="service1" msg="Service failover to other available inter-
face(s)."

date=2020-03-29 time=17:33:25 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585528405170874263 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" interface="R150" member="1" serviceid=1 service="service1" gateway="10.100.1.1"
msg="Member link is unreachable or miss threshold. Stop forwarding traffic. "

date=2020-03-29 time=18:05:14 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585530314708843222 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" interface="R150" member="1" serviceid=1 service="service1" gateway="10.100.1.1"
metric="packet-loss" oldvalue="1" newvalue="2" msg="The member order changed by performance
metric."

date=2020-03-29 time=19:25:40 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585535140122779004 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" serviceid=1 service="service1" seq="1,2" msg="Service prioritized by SLA will
be redirected in sequence order."

date=2020-03-29 time=19:27:02 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585535222140485480 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" interface="R150" member="1" serviceid=1 service="service1" gateway="10.100.1.1"
oldvalue="1" newvalue="2" msg="The member SLA order changed."

date=2020-03-29 time=19:38:33 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585535913042763548 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" serviceid=1 service="service1" member="1(R150),2(R160)" msg="Service will be
load balanced among members with available routing."

date=2020-03-29 time=20:58:50 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585540730662430230 tz="-0700" logdesc="Virtual WAN Link status" event-
type="Service" serviceid=1 service="service1" msg="Service disabled caused by role mismatch."
```

Virtual WAN Link volume status

Event type = VOLUME

```
date=2020-03-29 time=20:46:19 logid="0113022924" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585539979756723714 tz="-0700" logdesc="Virtual WAN Link volume status"
eventtype="Volume" interface="port12" member="2" msg="Member enters into conservative status
with limited ablity to receive new sessions for too much traffic."

date=2020-03-29 time=20:46:19 logid="0113022924" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585539979756723714 tz="-0700" logdesc="Virtual WAN Link volume status"
eventtype="Volume" interface="wan1" member="2" msg="Member resumes normal status to receive
new sessions for internal adjustment."
```

Virtual WAN Link SLA information

Event type = SLA

```
date=2020-03-29 time=16:51:27 logid="0113022925" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525888177637570 tz="-0700" logdesc="Virtual WAN Link SLA information"
```

```
eventtype="SLA" healthcheck="ping1" slatargetid=1 interface="R150" status="up" latency="0.013"
jitter="0.001" packetloss="100.000%" inbandwidth="0kbps" outbandwidth="0kbps" bib-
andwidth="0kbps" slamap="0x0" metric="packetloss" msg="Health Check SLA status. SLA failed due
to being over the performance metric threshold."

date=2020-03-29 time=16:51:21 logid="0113022925" type="event" subtype="sdwan" level-
l="information" vd="root" eventtime=1585525881177944788 tz="-0700" logdesc="Virtual WAN Link
SLA information" eventtype="SLA" healthcheck="ping1" slatargetid=1 interface="R160" status-
s="up" latency="0.010" jitter="0.001" packetloss="0.000%" inbandwidth="0kbps" out-
bandwidth="0kbps" bibandwidth="0kbps" slamap="0x1" msg="Health Check SLA status."
```

Event type = HEALTH CHECK

```
date=2020-03-29 time=16:36:54 logid="0113022925" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525014564428201 tz="-0700" logdesc="Virtual WAN Link SLA information"
eventtype="Health Check" healthcheck="ping1" interface="R160" probeproto="ping" oldvalue=""
newvalue="alive" msg="SD-WAN health-check member initial state."

date=2020-03-29 time=16:55:18 logid="0113022925" type="event" subtype="sdwan" level="warning"
vd="root" eventtime=1585526118334582737 tz="-0700" logdesc="Virtual WAN Link SLA information"
eventtype="Health Check" healthcheck="ping1" interface="R150" probeproto="ping" old-
value="alive" newvalue="die" msg="SD-WAN health-check member changed state."

date=2020-03-29 time=16:54:35 logid="0113022925" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585526075811696627 tz="-0700" logdesc="Virtual WAN Link SLA information"
eventtype="Health Check" healthcheck="ping1" interface="R150" probeproto="ping" oldvalue="die"
newvalue="alive" msg="SD-WAN health-check member changed state."
```

Virtual WAN Link Neighbor

Event type = NEIGHBOR

```
date=2020-03-29 time=20:57:36 logid="0113022926" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585540656722222479 tz="-0700" logdesc="Virtual WAN Link Neighbor status"
eventtype="Neighbor" neighbor="10.100.1.1" member="1" msg="Neighbor(10.100.1.1) for member(1)
is unselected forcefully due to configuration change."

date=2020-03-29 time=20:58:51 logid="0113022928" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585540731163096946 tz="-0700" logdesc="Virtual WAN Link Neighbor primary"
eventtype="Neighbor" oldvalue="standalone" newvalue="primary" msg="Selected role is changed."

date=2020-03-29 time=20:58:51 logid="0113022926" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585540731163094572 tz="-0700" logdesc="Virtual WAN Link Neighbor status"
eventtype="Neighbor" neighbor="10.100.1.1" member="1" msg="Neighbor(10.100.1.1) for member(1)
is selected."

date=2020-03-29 time=21:01:01 logid="0113022929" type="event" subtype="sdwan" level="warning"
vd="root" eventtime=1585540861280903746 tz="-0700" logdesc="Virtual WAN Link Neighbor sec-
ondary" eventtype="Neighbor" oldvalue="primary" newvalue="secondary" msg="Selected role is
changed."

date=2020-03-29 time=21:01:01 logid="0113022926" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585540861280842811 tz="-0700" logdesc="Virtual WAN Link Neighbor status"
eventtype="Neighbor" neighbor="10.100.1.1" member="1" msg="Neighbor(10.100.1.1) for member(1)
is unselected."
```

SD-WAN logging improvement to identify matched application

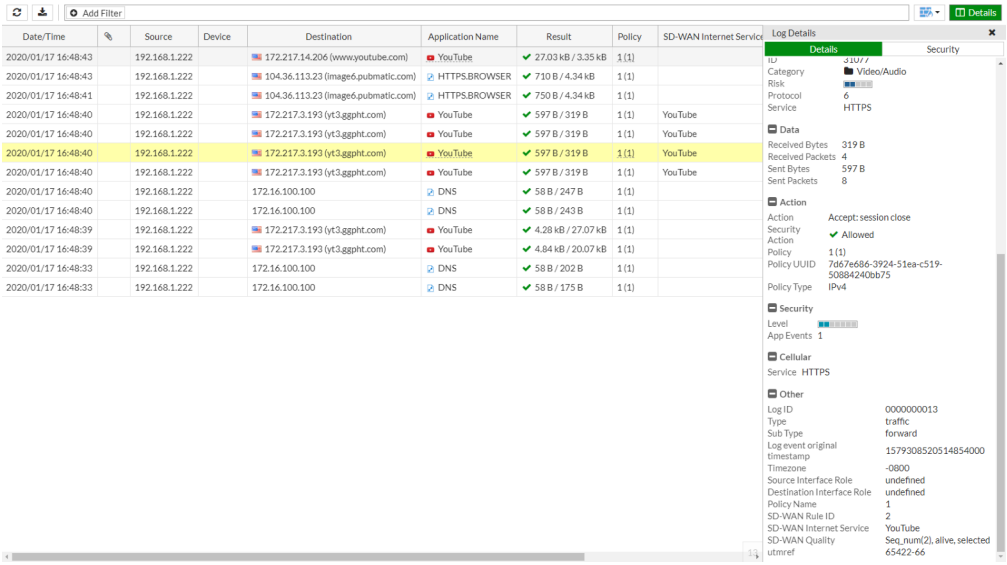
In SD-WAN rules, users can define destinations based on applications. With this enhancement, the `vwlservice` field in the forward traffic log has been updated to include the matched application.

Sample log

```
183: date=2020-01-17 time=16:48:40 logid="0000000013" type="traffic" subtype="forward" level-
l="notice" vd="root" eventtime=1579308520544853557 tz="-0800" srcip=192.168.1.222 src-
port=51530 srcintf="port10" srcintfrole="undefined" dstip=172.217.3.193 dstport=443
dstintf="port9" dstintfrole="undefined" sessionid=12654 proto=6 action="close" policyid=1 poli-
cytype="policy" poluuid="7d67e686-3924-51ea-c519-50884240bb75" policyname="1" service="HTTPS"
dstcountry="United States" srccountry="Reserved" trandisp="snat" transip=172.16.200.1 trans-
port=51530 appid=31077 app="YouTube" appcat="Video/Audio" apprisk="elevated" applist="g-wifi-
default" duration=1 sentbyte=597 rcvdbyte=319 sentpkt=8 rcvdpkt=4 vwlid=2 vwlservice="YouTube"
vwlquality="Seq_num(2), alive, selected" utmaction="allow" countapp=1 utmref=65422-94
```

To view SD-WAN logs in the GUI:

1. Go to *Log & Report* > *Forward Traffic*. The *SD-WAN Internet Service* column displays the application.
2. Select a log entry to view the details.

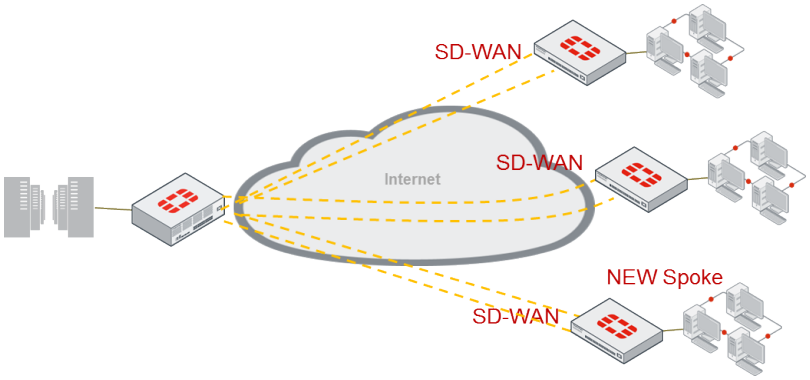


SD-WAN configuration portability

When configuring SD-WAN, adding interfaces to members is optional. This allows a configuration to be copied directly from one device to another, without requiring the devices to have interfaces with the same names.

After the configuration is pasted to the new device, add the interfaces to the new device to make it fully functional.

Example



To copy the SD-WAN configuration from an existing spoke to a new spoke:

1. Copy the configuration from the configured spoke:

```
config system virtual-wan-link
    set status enable
    config members
        edit 1
            set interface "_OCVPN3-0.0"
        next
        edit 2
            set interface "_OCVPN3-0.1"
        next
    end
    config health-check
        edit "office"
            set server "office365.com"
            set protocol http
            set sla-fail-log-period 300
            set sla-pass-log-period 300
            set members 2 1
            config sla
                edit 1
                    set latency-threshold 300
                    set jitter-threshold 200
                next
                edit 2
                    set link-cost-factor latency
                    set latency-threshold 20
                next
            end
        next
        ...
    end
    config service
        edit 2
            set name "Office365"
            set mode sla
            set internet-service enable
            set internet-service-app-ctrl 327782
```

```
            config sla
                edit "office"
                    set id 1
                next
            end
            set priority-members 2 1
        next
        ...
    end
end
```

2. Paste the configuration onto the new spoke:

```
config system virtual-wan-link
    set status enable
    config members
        edit 1
        next
        edit 2
        next
    end
    config health-check
        edit "office"
            set server "office365.com"
            set protocol http
            set sla-fail-log-period 300
            set sla-pass-log-period 300
            set members 2 1
            config sla
                edit 1
                    set latency-threshold 300
                    set jitter-threshold 200
                next
                edit 2
                    set link-cost-factor latency
                    set latency-threshold 20
                next
            end
        next
        ...
    end
    config service
        edit 2
            set name "Office365"
            set mode sla
            set internet-service enable
            set internet-service-app-ctrl 327782
            config sla
                edit "office"
                    set id 1
                next
            end
            set priority-members 2 1
        next
        ...
```

```
    end
end
```

The member interfaces are not copied over. Already configured interfaces are not unset. The member is disabled until an interface is configured.

3. Configure the member interfaces on the new spoke:

```
config system virtual-wan-link
    config members
        edit 1
            set interface "_OCVPN4-0.0"
        next
        edit 2
            set interface "_OCVPN4-0.1"
        next
    end
end
```

After the interfaces are configured, the new spoke will function like the other spokes.

SD-WAN log format improvements

The SD-WAN log format has been improved for better reporting and event handler creation on FortiAnalzyer.

Sample logs

The following sample logs identify where the improvements were made to the log format.

Service field

The `service` field only includes the service name. The service ID was removed from the `service` field and added to a new field named `serviceid`.

Old Format

```
date=2019-11-05 time=12:11:16 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569438676644424772 tz="-0700"
logdesc="Virtual WAN Link status" eventtype="Service" service="1(gmail)"
msg="Service prioritized by latency will be redirected in seq-num order 1(lan2) 2
(wan1)"
```

New format

```
date=2020-02-04 time=15:24:23 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580858663336645512 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" serviceid=1 service="gmail"
metric="latency" seq="2,1" msg="Service prioritized by performance metric will be
redirected in sequence order."
```

Name field

The `name` field has been replaced with the more specific `healthcheck` field to be consistent with other logs and to reduce confusion.

Old format

```
date=2019-11-03 time=17:13:28 logid="0113022925" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569284008246643001 tz="-0700"
logdesc="Virtual WAN Link SLA information" eventtype="SLA" name="test"
interface="lan2" status="down" latency="0.000" jitter="0.000"
packetloss="85.000%" inbandwidth="0kbps" outbandwidth="0kbps" bibandwidth="0kbps"
slamap="0x0" msg="Health Check SLA status. SLA 1 failed due to being over the
packetloss threshold "
```

New format

```
date=2020-02-04 time=14:39:08 logid="0113022925" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580855948407682526 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="SLA" healthcheck="ping1"
slatargetid=1 interface="R160" status="up" latency="0.010" jitter="0.000"
packetloss="21.000%" inbandwidth="0kbps" outbandwidth="0kbps" bibandwidth="0kbps"
slamap="0x0" metric="packetloss" msg="Health Check SLA status. SLA failed due to
being over the performance metric threshold."
```

Old format

```
date=2019-11-06 time=17:26:12 logid="0113022925" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569543972762277480 tz="-0700"
logdesc="Virtual WAN Link SLA information" eventtype="Health Check" name="test-1-
VIRTUAL_WAN_LINK-1" interface="lan2" probeproto="ping" oldstate="die"
newstate="alive" msg="SD-WAN health-check member changed state"
```

New format

```
date=2020-02-04 time=14:14:42 logid="0113022925" type="event" subtype="sdwan"
level="warning" vd="root" eventtime=1580854483005525076 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="Health Check"
healthcheck="ping1-2-VIRTUAL_WAN_LINK-2" interface="R160" probeproto="ping"
oldvalue="alive" newvalue="die" msg="SD-WAN health-check member changed state."
```

SLA field

The `sla` field has been replaced with the more specific `slatargetid` field.

Old format

```
date=2019-11-06 time=16:51:05 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1573087865540014616 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping2"
sla="22" oldpassmember="2" newpassmember="1" msg="Number of pass members changed.
Member 2 out-of-sla"
```

New format

```
date=2020-02-04 time=14:38:16 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580855896895319923 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 oldvalue="2" newvalue="1" msg="Number of pass member changed."
date=2020-02-04 time=14:38:16 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580855896895316020 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 member="2" msg="Member status changed. Member out-of-sla."
```

SLA ID

The SLA ID was moved from the `msg` field and added to the new `slatargetid` field. The SLA values were also removed from the `msg` field. There is now one log for each SLA failure.

Old format

```
date=2019-12-06 time=10:19:53 logid="0113022925" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1575656393121996604 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="SLA" name="1"
interface="port1" status="up" latency="0.092" jitter="0.006" packetloss="0.000%"
inbandwidth="99.98Mbps" outbandwidth="99.99Mbps" bibandwidth="199.97Mbps"
slamap="0x0" msg="Health Check SLA status. SLA 1 failed due to being over the
latency threshold SLA 2 failed due to being over the jitter threshold"
```

New format

```
date=2020-02-05 time=15:56:27 logid="0113022925" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580946987433804652 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="SLA" healthcheck="ping1"
slatargetid=2 interface="R160" status="up" latency="3.012" jitter="1.002"
packetloss="43.000%" inbandwidth="0kbps" outbandwidth="0kbps" bibandwidth="0kbps"
slamap="0x0" metric="latency" msg="Health Check SLA status. SLA failed due to
being over the performance metric threshold." date=2020-02-05 time=15:56:27
logid="0113022925" type="event" subtype="sdwan" level="notice" vd="root"
eventtime=1580946987433799366 tz="-0800" logdesc="Virtual WAN Link SLA
information" eventtype="SLA" healthcheck="ping1" slatargetid=1 interface="R160"
status="up" latency="3.012" jitter="1.002" packetloss="43.000%"
inbandwidth="0kbps" outbandwidth="0kbps" bibandwidth="0kbps" slamap="0x0"
metric="jitter" msg="Health Check SLA status. SLA failed due to being over the
performance metric threshold."
```

Old and new value fields

The *old* and *new* value field types have been replaced with the `oldvalue` and `newvalue` fields since the field values are meaningful enough to cover different log types.

Old format

```
date=2019-11-05 time=15:08:07 logid="0113022927" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569449287861854771 tz="-0700"
logdesc="Virtual WAN Link Neighbor standalone" eventtype="Neighbor"
oldselectedrole="primary" newselectedrole="standalone" msg="Selected role is
changed."
```

New format

```
date=2020-02-04 time=17:09:57 logid="0113022927" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580864997042080958 tz="-0800"
logdesc="Virtual WAN Link Neighbor standalone" eventtype="Neighbor"
oldvalue="primary" newvalue="standalone" msg="Selected role is changed."
```

Old format

```
date=2019-11-06 time=16:51:05 logid="0113022925" type="event" subtype="sdwan"
level="warning" vd="root" eventtime=1573087865315149386 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="Health Check" name="test2-
2-VIRTUAL_WAN_LINK-2" interface="port15" probeproto="ping" oldstate="alive"
newstate="die" msg="SD-WAN health-check member changed state"
```

New format

```
date=2020-02-04 time=14:14:42 logid="0113022925" type="event" subtype="sdwan"
level="warning" vd="root" eventtime=1580854483005525076 tz="-0800"
logdesc="Virtual WAN Link SLA information" eventtype="Health Check"
healthcheck="ping1-2-VIRTUAL_WAN_LINK-2" interface="R160" probeproto="ping"
oldvalue="alive" newvalue="die" msg="SD-WAN health-check member changed state."
```

Old format

```
date=2019-11-04 time=17:28:11 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569371291676959641 tz="-0700"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="test"
sla="1" oldpassmember="1" newpassmember="0" msg="Number of pass members changed.
Member 2 out-of-sla"
```

New format

```
date=2020-02-04 time=17:17:34 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580865454841077461 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 oldvalue="2" newvalue="1" msg="Number of pass member changed."
date=2020-02-04 time=17:17:34 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580865454841074245 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 member="1" msg="Member status changed. Member out-of-sla."
```

Network performance metrics

The latency, jitter, and packet loss network performance metrics were removed from the `msg` field and moved to a new field named `metric`. The cause factor has also been removed from the `msg` field.

Old format

```
date=2019-11-05 time=16:22:00 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1572999721054428968 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" service="2(rule12)"
msg="Service prioritized by latency will be redirected in seq-num order 2(port15)
1(port13)."
```

New format

```
date=2020-02-04 time=15:24:23 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580858663336645512 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" serviceid=1 service="gmail"
metric="latency" seq="2,1" msg="Service prioritized by performance metric will be
redirected in sequence order."
```

Old format

```
date=2019-11-05 time=12:11:16 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569438676644407292 tz="-0700"
logdesc="Virtual WAN Link status" eventtype="Service" interface="wan1" member="2"
service="1(gmail)" msg="The member link quality latency order changed from 1 to
2."
```

New format

```
date=2020-02-04 time=15:40:48 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580859648553624138 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" interface="R160" member="2"
serviceid=1 service="gmail" gateway="10.100.1.5" metric="packet-loss"
oldvalue="1" newvalue="2" msg="The member order changed by performance metric."
```

Gateway address

The `gateway` address has been removed from the `msg` field and added to a new field named `gateway`.

Old format

```
date=2019-11-07 time=07:49:58 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1569595798367258194 tz="-0700"
logdesc="Virtual WAN Link status" eventtype="Service" interface="wan1" member="2"
service="2(is)" msg="Member link is available. Start forwarding traffic. Service
will be redirected to interface(wan1) gateway(172.18.45.1)"
```

New format

```
date=2020-02-04 time=15:39:04 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580859544464985538 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" interface="R160" member="2"
serviceid=2 service="google-isdb" gateway="10.100.1.5" msg="Member link is
available. Start forwarding traffic. "
```

Seq-num order

The `seq-num` order was removed from the `msg` field and added to the new field named `seq`. The order values were also removed from the `msg` field.

Old format

```
date=2019-11-05 time=16:22:00 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1572999721054428968 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" service="2(rule12)"
msg="Service prioritized by latency will be redirected in seq-num order 2(port15)
1(port13)"
```

New format

```
date=2020-02-04 time=15:39:04 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580859544464944421 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Service" serviceid=1 service="gmail"
metric="latency" seq="2,1" msg="Service prioritized by performance metric will be
redirected in sequence order."
```

Member value

The member value was removed from the `msg` field and added to the `member` field. There is now one log for each changed member and another log for how the pass member changed.

Old format

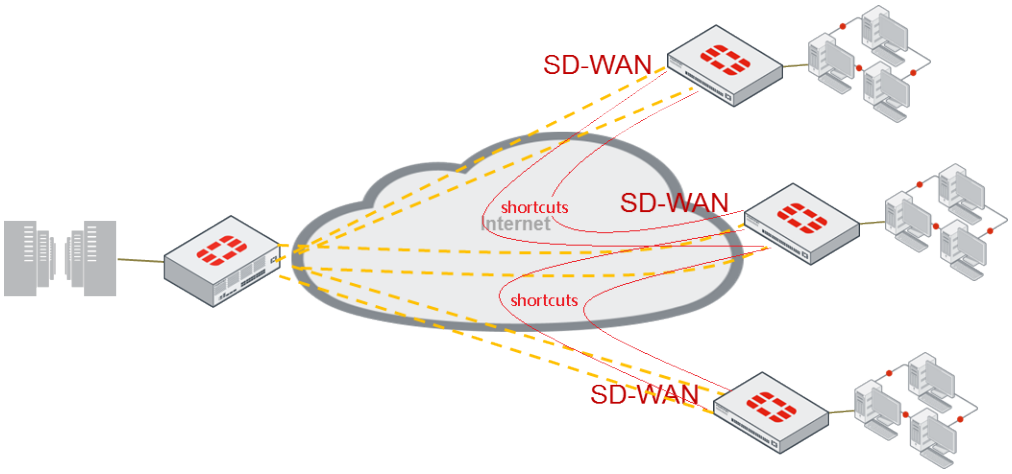
```
date=2019-12-11 time=14:47:40 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1576104460831070527 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="test"
sla="22" oldpassmember="2" newpassmember="1" msg="Number of pass members changed.
Member 2 out-of-sla"
```

New format

```
date=2020-02-04 time=17:17:34 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580865454841077461 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 oldvalue="2" newvalue="1" msg="Number of pass member changed."
date=2020-02-04 time=17:17:34 logid="0113022923" type="event" subtype="sdwan"
level="notice" vd="root" eventtime=1580865454841074245 tz="-0800"
logdesc="Virtual WAN Link status" eventtype="Health Check" healthcheck="ping1"
slatargetid=1 member="1" msg="Member status changed. Member out-of-sla."
```

SD-WAN monitor on ADVPN shortcuts

SD-WAN can monitor ADVPN shortcuts link quality by dynamically creating link monitors for each ADVPN link. The dynamic link monitor on the spoke will use ICMP probes and the IP address of the gateway as the monitored server. These ICMP probes will not be counted as actual user traffic that keeps the spoke-to-spoke tunnel alive.



- When no shortcut is established:

```
# diagnose sys virtual-wan-link health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.038), jitter(0.006) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.035), jitter(0.004) sla_
map=0x3
```

- When one shortcut is established:

```
# diagnose sys virtual-wan-link health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.039), jitter(0.003) sla_
map=0x3
Seq(1 tunnel-1_0): state(alive), packet-loss(0.000%) latency(0.060), jitter(0.023) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.035), jitter(0.002) sla_
map=0x3
```

- When more than one shortcut is established:

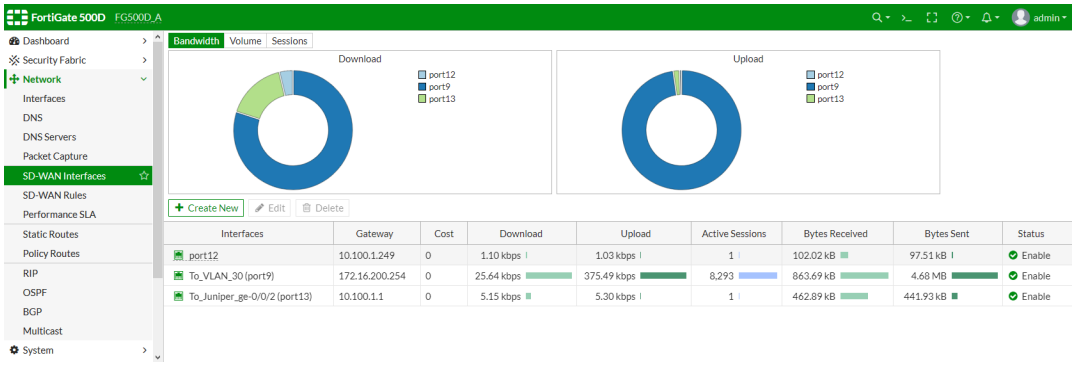
```
# diagnose sys virtual-wan-link health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.036), jitter(0.004) sla_
map=0x3
Seq(1 tunnel-1_0): state(alive), packet-loss(0.000%) latency(0.041), jitter(0.009) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.030), jitter(0.005) sla_
map=0x3
Seq(2 tunnel-2_0): state(alive), packet-loss(0.000%) latency(0.031), jitter(0.004) sla_
map=0x3
```

SD-WAN GUI and monitoring enhancements

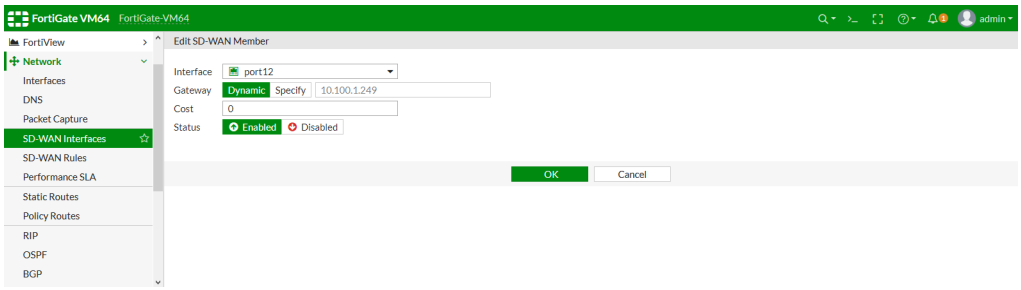
The SD-WAN pages in the GUI are updated to simplify SD-WAN configuration. New charts and monitoring capabilities are also added. DNS is now a supported protocol in performance SLA.

SD-WAN interfaces

The SD-WAN interface list shows pie charts at the top of the list, and includes more information about each interface in the table, such as the number of sessions and the bytes sent and received.



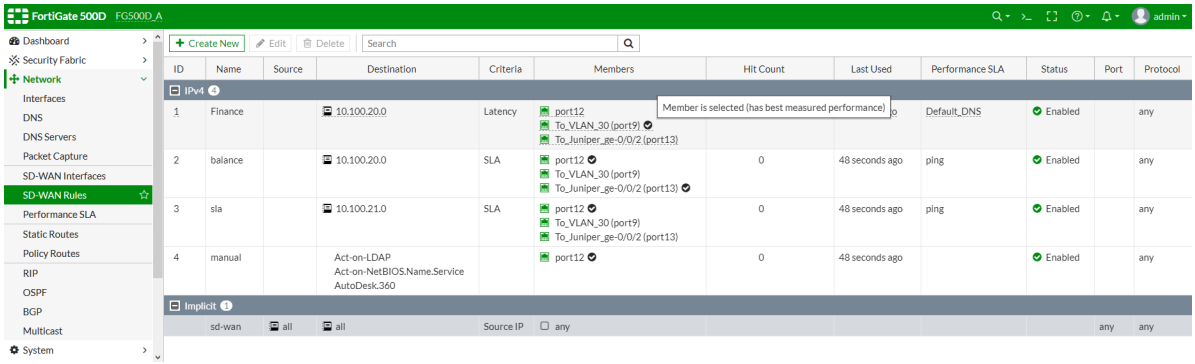
The gateway configuration for an SD-WAN interface that is using DHCP is simplified.



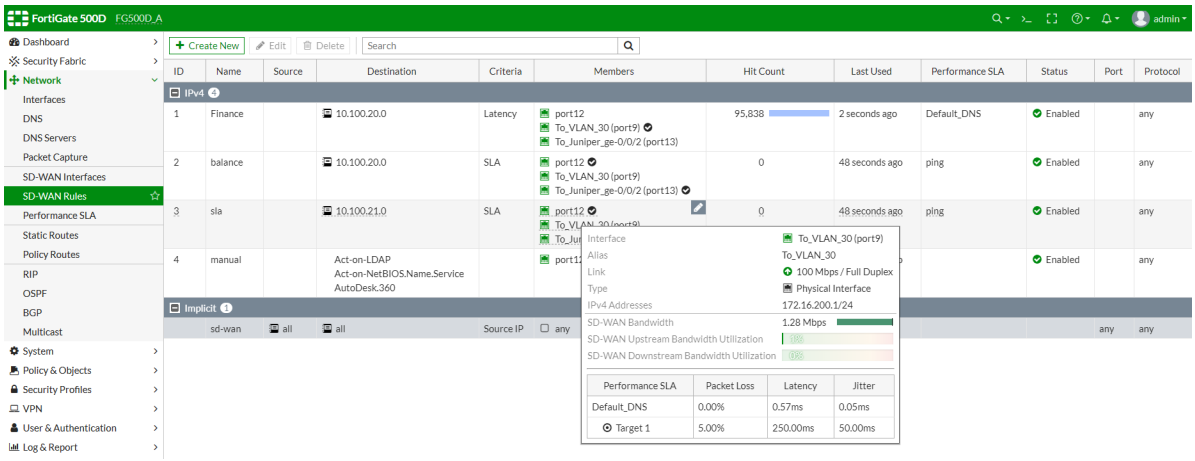
SD-WAN rules

In the SD-WAN rules list, the interface that is currently selected by the rule has a checkmark next to its name in the *Members* column. Hover the cursor over the checkmark to open a tooltip that gives the reason why that member is selected, such as *has best measured performance*. Even if multiple members are selected, only the highest ranked member is highlighted, unless the mode is *Maximize Bandwidth (SLA)* (`load-balance`).

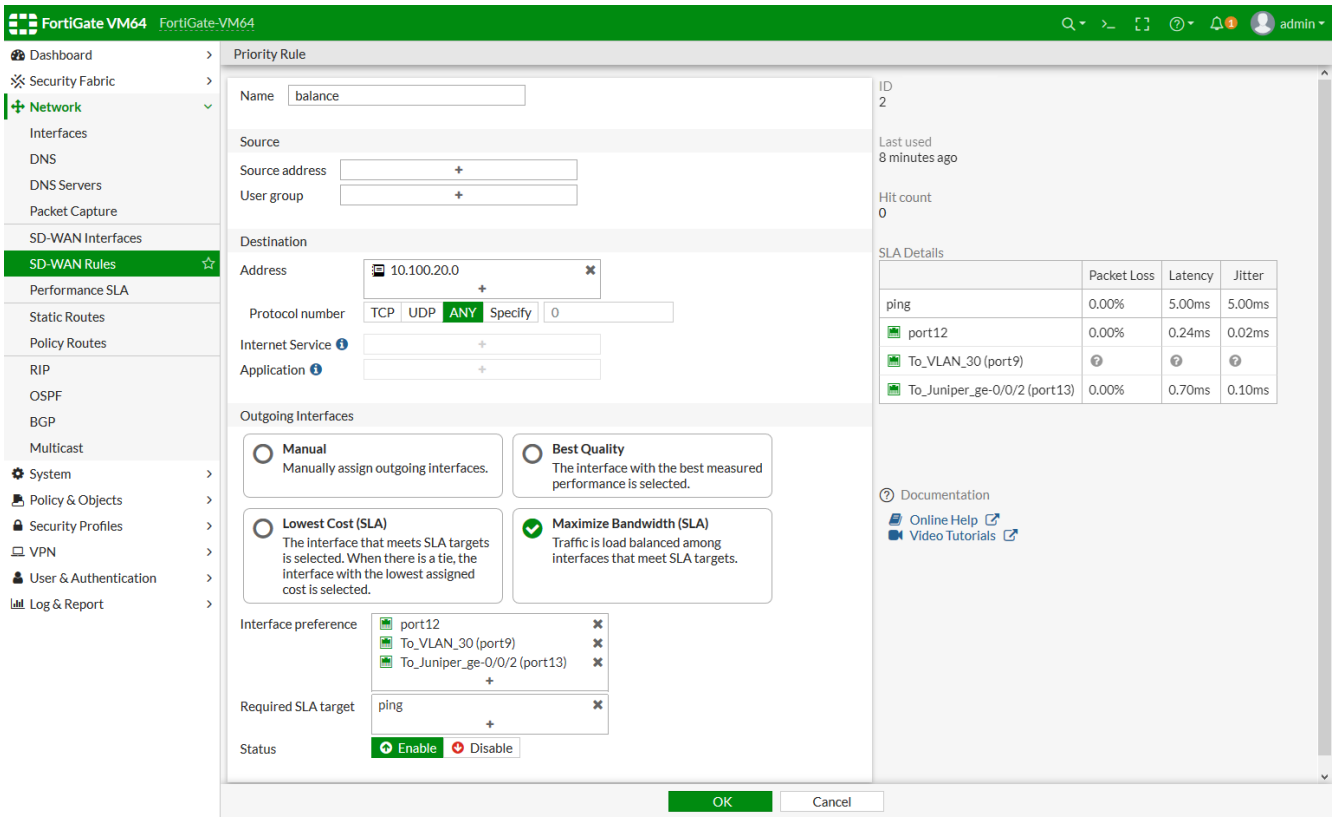
Hit Count and *Last Used* columns are also added to the table.



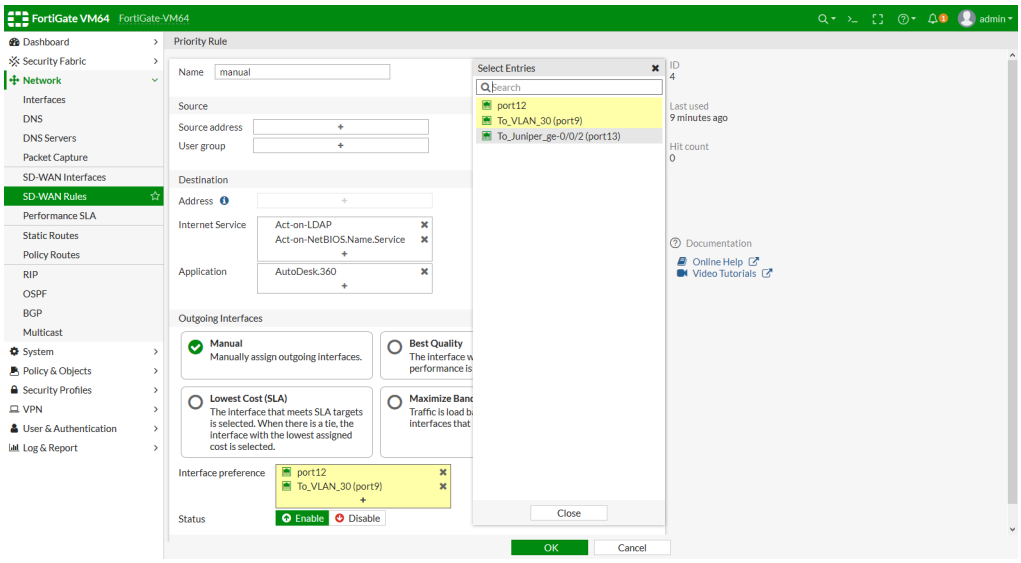
Hover over a member name to open the SD-WAN member tooltip. It includes health check and SLA statistics tables.



When editing an SD-WAN rule, the strategies are listed on cards that include a brief description of that strategy. The gutter on the right side of the page includes the hit count for the rule, when it was last used, and a table showing statistics for the currently selected interfaces and SLA targets (depending on the selected strategy).



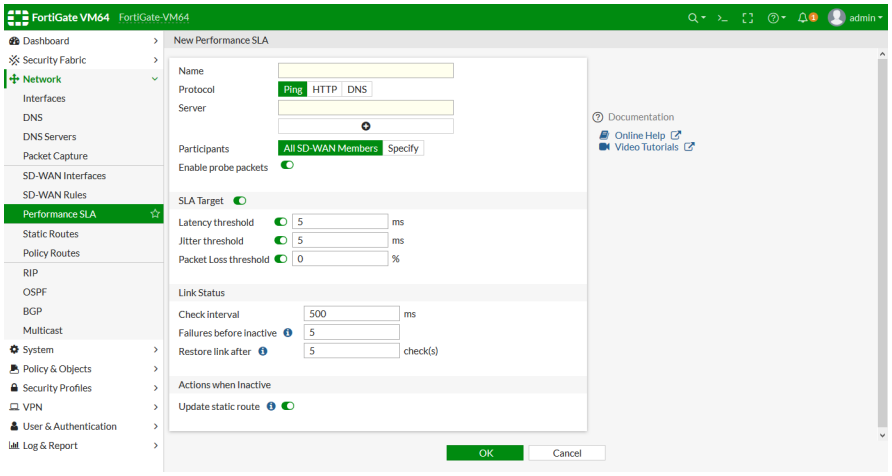
When *Manual* mode is selected, multiple members can be selected.



Performance SLA

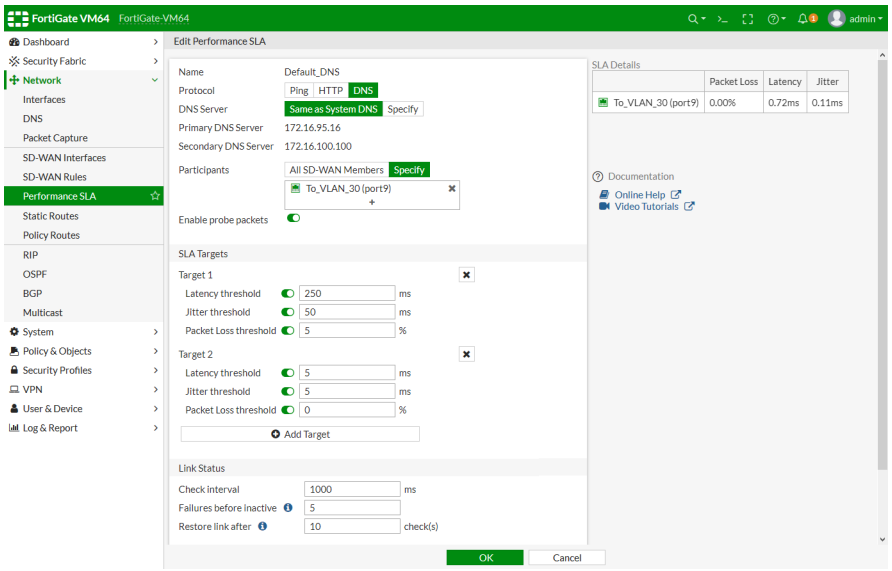
SLA targets

When configuring a performance SLA, by default, only one SLA target can be configured. Additional targets can be created in the CLI, after which they will also be available from the GUI.

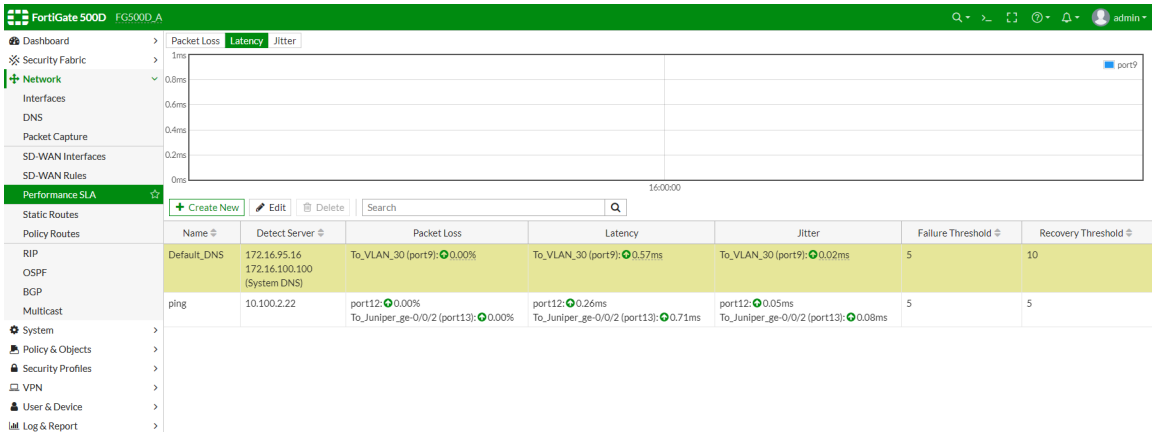


DNS protocol

The IPv4 DNS protocol can be selected, and the system DNS servers can be used.

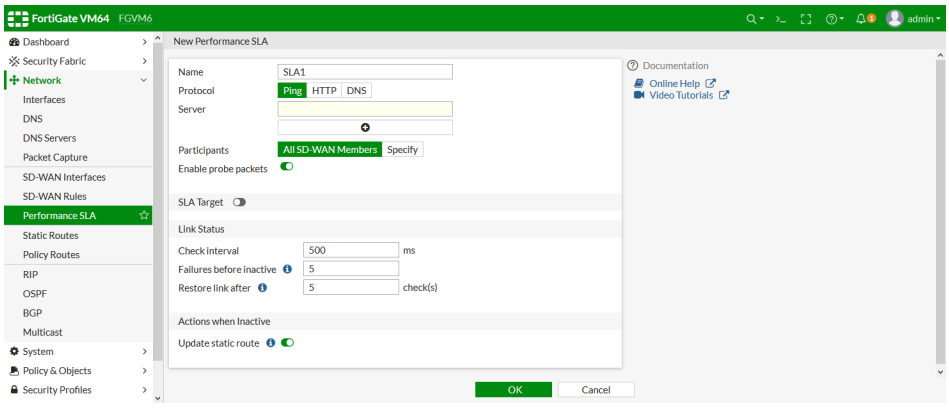


In the Performance SLA table, the *Detect Server* column will show that the system DNS servers are used.



Participants

When adding a new performance SLA, by default, all SD-WAN members are included as participants.



In the CLI, `member` is set to zero to include all participants.

```
config system virtual-wan-link
    config health-check
        edit "SLA1"
            set system-dns enable
            set members 0
        next
    end
end
```

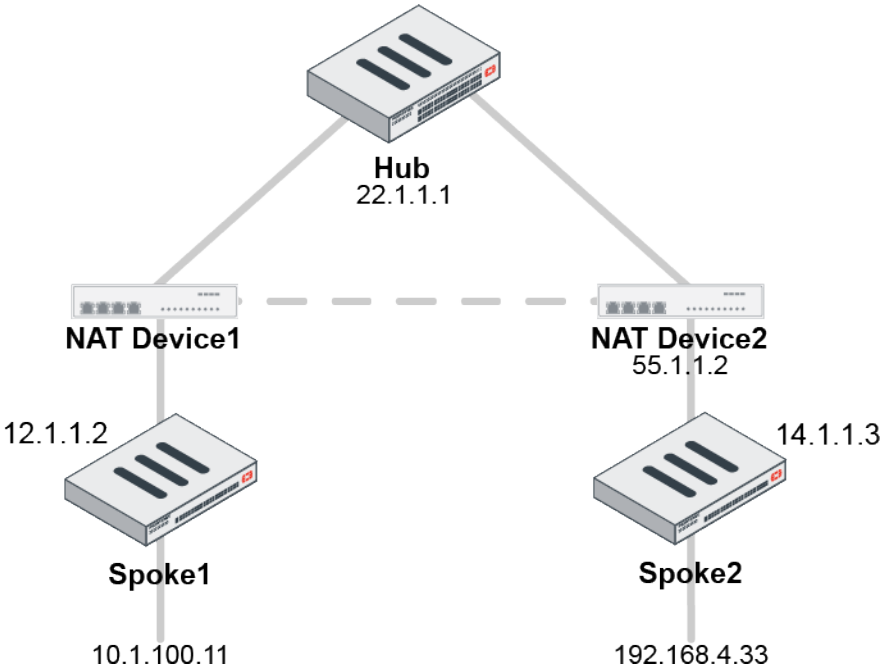
Routing monitor

Hit Count and *Last Used* columns are added to the *Network* > *Policy Routes* page and *Policy Routing* widget.

Enhance ADVPN to support UDP hole punching for spokes behind NAT

Previously, spokes behind NAT devices could only create shortcuts if DNAT was used on the NAT devices. This feature adds UDP hole punching capability, which allows ADVPN shortcuts to be established through a UDP hole on a NAT device. The NAT device must support RFC 4787 Endpoint-Independent Mapping.

In the following example, device 10.1.100.11 behind Spoke1 needs to reach device 192.168.4.33 behind Spoke2. Spoke1 and Spoke2 are behind NAT devices and have established IPsec tunnels to the Hub. The hole punching creates a shortcut between Spoke1 and Spoke2 that bypasses the Hub.



To verify the ADVPN shortcut is established between both spokes behind NAT:

```
# diagnose debug enable
# diagnose debug application ike -1
ike 0: comes 22.1.1.1:4500->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Informational id=3c10fb6a76f1e264/6c7b397100dffc63:58ac7c02 len=204
ike 0:toHub1:35: notify msg received: SHORTCUT-OFFER
```

```
ike 0:toHub1: shortcut-offer 10.1.100.11->192.168.4.33 psk 64 ppk 0 ver 1 mode 0
ike 0 looking up shortcut by addr 192.168.4.33, name toHub1
ike 0:toHub1: send shortcut-query 1438189781753480593 d3fdd1bfbc94caee/0000000000000000
12.1.1.2 10.1.100.11->192.168.4.33 psk 64 ttl 32 nat 1 ver 1 mode 0
ike 0:toHub1:35: sent IKE msg (SHORTCUT-QUERY): 12.1.1.2:4500->22.1.1.1:4500, len=236, id=3c10-
0fb6a76f1e264/6c7b397100dffc63:12e263f7
ike 0: comes 22.1.1.1:4500->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Informational id=3c10fb6a76f1e264/6c7b397100dffc63:4976e1ac len=236
ike 0:toHub1:35: notify msg received: SHORTCUT-REPLY
ike 0:toHub1: recv shortcut-reply 1438189781753480593 d3fdd1bfbc94caee/16a1eb5b0f37ee23
14.1.1.3 to 10.1.100.11 psk 64 ppk 0 ver 1 mode 0 nat 55.1.1.2:64916
ike 0:toHub1: iif 22 192.168.4.33->10.1.100.11 route lookup oif 21
ike 0:toHub1: shortcut-reply received from 55.1.1.2:64916, local-nat=yes, peer-nat=yes
ike 0:toHub1: NAT hole punching to peer at 55.1.1.2:64916
ike 0:toHub1: created connection: 0x5e71f58 6 12.1.1.2->55.1.1.2:64916.     <==55.1.1.2:64916
this is UDP hole of NAT device
ike 0:toHub1: adding new dynamic tunnel for 55.1.1.2:64916
ike 0:toHub1_0: added new dynamic tunnel for 55.1.1.2:64916
ike 0:toHub1_0:48: initiator: main mode is sending 1st message...
ike 0:toHub1_0:48: cookie d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0:toHub1_0:48: sent IKE msg (ident_i1send): 12.1.1.2:4500->55.1.1.2:64916, len=632, id=d3-
3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=252
ike 0:toHub1_0:48: initiator: main mode get 1st response...
…
…
ike 0:toHub1_0:48: negotiation result
ike 0:toHub1_0:48: proposal id = 1:
ike 0:toHub1_0:48:   protocol id = ISAKMP:
ike 0:toHub1_0:48:      trans_id = KEY_IKE.
ike 0:toHub1_0:48:      encapsulation = IKE/none
ike 0:toHub1_0:48:         type=OAKLEY_ENCRYPT_ALG, val=AES_CBC, key-len=128
ike 0:toHub1_0:48:         type=OAKLEY_HASH_ALG, val=SHA2_256.
ike 0:toHub1_0:48:         type=AUTH_METHOD, val=PRESHARED_KEY.
ike 0:toHub1_0:48:         type=OAKLEY_GROUP, val=MODP2048.
ike 0:toHub1_0:48: ISAKMP SA lifetime=86400
ike 0:toHub1_0:48: sent IKE msg (ident_i2send): 12.1.1.2:4500->55.1.1.2:64916, len=380, id=d3-
3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=380
ike 0:toHub1_0:48: initiator: main mode get 2nd response...
…
…
ike 0:toHub1_0:48: add INITIAL-CONTACT
ike 0:toHub1_0:48: add INTERFACE-ADDR4 10.10.1.100
ike 0:toHub1_0:48: sent IKE msg (ident_i3send): 12.1.1.2:4500->55.1.1.2:64916, len=140, id=d3-
3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=124
ike 0:toHub1_0:48: initiator: main mode get 3rd response...
ike 0:toHub1_0:48: received p1 notify type INTERFACE-ADDR4
ike 0:toHub1_0:48: INTERFACE-ADDR4 10.10.1.102
ike 0:toHub1_0:48: peer identifier IPV4_ADDR 14.1.1.3
ike 0:toHub1_0:48: PSK authentication succeeded
ike 0:toHub1_0:48: authentication OK
```

```
ike 0:toHub1_0:48: established IKE SA d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0:toHub1_0:48: auto-discovery receiver
ike 0:toHub1_0:48: auto-discovery 2
ike 0:toHub1_0: add R/32 route 10.10.1.102 via 10.10.1.102, intf=toHub1(22)
ike 0:toHub1_0: add peer route 10.10.1.102
ike 0:toHub1: schedule auto-negotiate
ike 0:toHub1_0:48: no pending Quick-Mode negotiations
ike 0:toHub1_0:toHub1: IPsec SA connect 6 12.1.1.2->55.1.1.2:64916
ike 0:toHub1_0:toHub1: using existing connection
ike 0:toHub1_0:toHub1: traffic triggered, serial=1 1:10.1.100.11:2048->1:192.168.4.33:0
ike 0:toHub1:toHub1: config found
ike 0:toHub1_0:toHub1: IPsec SA connect 6 12.1.1.2->55.1.1.2:64916 negotiating
ike 0:toHub1_0:48: cookie d3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467
ike 0:toHub1_0:48:toHub1:109: natt flags 0x1f, encmode 1->3
ike 0:toHub1_0:48:toHub1:109: initiator selectors 0 0:0.0.0.0/0.0.0.0:0:0-
>0:0.0.0.0/0.0.0.0:0:0
ike 0:toHub1_0:48: sent IKE msg (quick_i1send): 12.1.1.2:4500->55.1.1.2:64916, len=620, id=d3-
3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Quick id=d3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467 len=444
ike 0:toHub1_0:48:toHub1:109: responder selectors 0:0.0.0.0/0.0.0.0:0->0:0.0.0.0/0.0.0.0:0
ike 0:toHub1_0:48:toHub1:109: my proposal:
…
…
ike 0:toHub1_0:48:toHub1:109: add IPsec SA: SPIs=79654cf1/5e9936a5
ike 0:toHub1_0:48:toHub1:109: IPsec SA dec spi 79654cf1 key
16:5E21180992B8892DE5142E1F53ABD29E auth 20:49AA4AE14994A39A138392AC517B6E79D98CA673
ike 0:toHub1_0:48:toHub1:109: IPsec SA enc spi 5e9936a5 key
16:BE16B8EF4E75F7B3CF97A1D58D996890 auth 20:2F46B57CAC6F3185BB182F9280312263325F6BAF
ike 0:toHub1_0:48:toHub1:109: added IPsec SA: SPIs=79654cf1/5e9936a5
ike 0:toHub1_0:48:toHub1:109: sending SNMP tunnel UP trapp
```

To verify the spoke-to-spoke IPsec phase 1 tunnel shortcut is established:

```
# diagnose vpn ike gateway list
vd: root/0
name: toHub1
version: 1
interface: wan2 6
addr: 12.1.1.2:4500 -> 22.1.1.1:4500
created: 503s ago
assigned IPv4 address: 10.10.1.100/255.255.255.0
nat: me
auto-discovery: 2 receiver
IKE SA: created 1/1  established 1/1  time 0/0/0 ms
IPsec SA: created 1/3  established 1/3  time 0/0/0 ms

  id/spi: 35 3c10fb6a76f1e264/6c7b397100dffc63
  direction: initiator
  status: established 503-503s ago = 0ms
  proposal: aes128-sha256
  key: 7fca86063ea2e72f-4efea6f1bec23948
  lifetime/rekey: 86400/85596
  DPD sent/recv: 00000000/00000000

vd: root/0
```

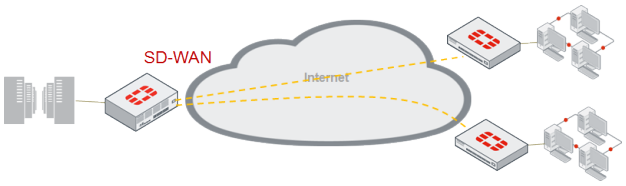
```
name: toHub1_0
version: 1
interface: wan2 6
addr: 12.1.1.2:4500 -> 55.1.1.2:64916
created: 208s ago
nat: me peer
auto-discovery: 2 receiver
IKE SA: created 1/1  established 1/1  time 20/20/20 ms
IPsec SA: created 1/1  established 1/1  time 10/10/10 ms

  id/spi: 48 d3fdd1bfbc94caee/16a1eb5b0f37ee23
  direction: initiator
  status: established 208-208s ago = 20ms
  proposal: aes128-sha256
  key: 9bcac400d8e14e11-fffde33eaa3a8263
  lifetime/rekey: 86400/85891
  DPD sent/recv: 0000000a/00000000
```

SD-WAN health check packet enhancement

SD-WAN health check probe packets now support Differentiated Services Code Point (DSCP) markers for accurate evaluation of the link performance for high priority applications by upstream devices.

When the SD-WAN health check packet is sent out, the DSCP can be set with a CLI command.



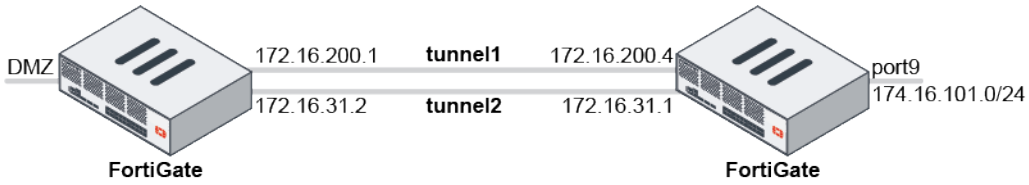
To mark health-check packets with DSCP:

```
config system virtual-wan-link
   config health-check
      edit <name>
         set diffservcode <6 bits binary, range 000000-111111>
      nexy
   end
end
```

Weighted round robin for IPsec aggregate tunnels

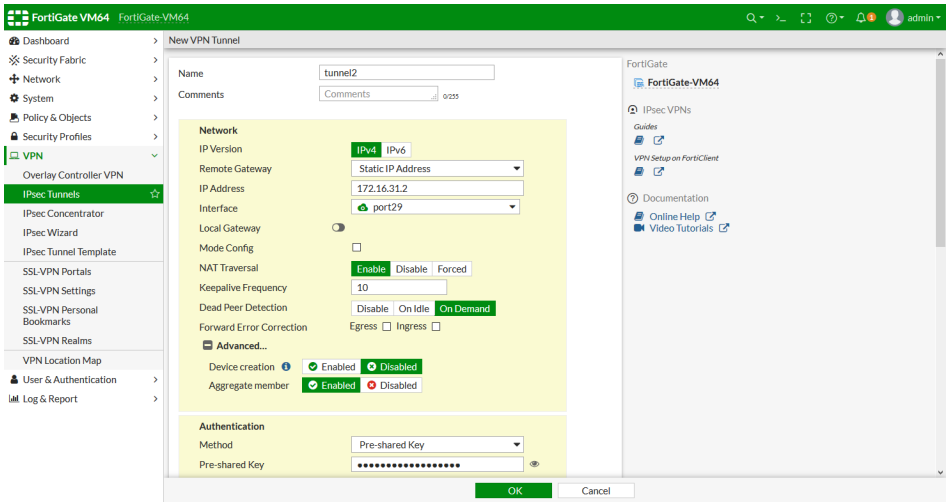
A weighted round robin algorithm can be used for IPsec aggregate tunnels to distribute traffic by the weight of each member tunnel.

In this example, the FortiGate has two IPsec tunnels put into IPsec aggregate. Traffic is distributed among the members, with one third over *tunnel1*, and two thirds over *tunnel2*. To achieve this, the weighted round robin algorithm is selected, *tunnel1* is assigned a weight of 10, and *tunnel2* is assigned a weight of 20.

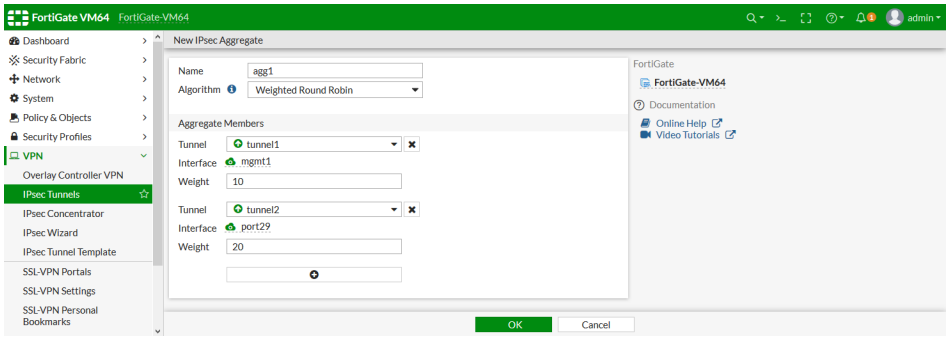


To create the IPsec aggregate in the GUI:

1. Create the *tunnel1* and *tunnel2* custom IPsec tunnels. Ensure that *Aggregate member* is *Enabled* for each tunnel.

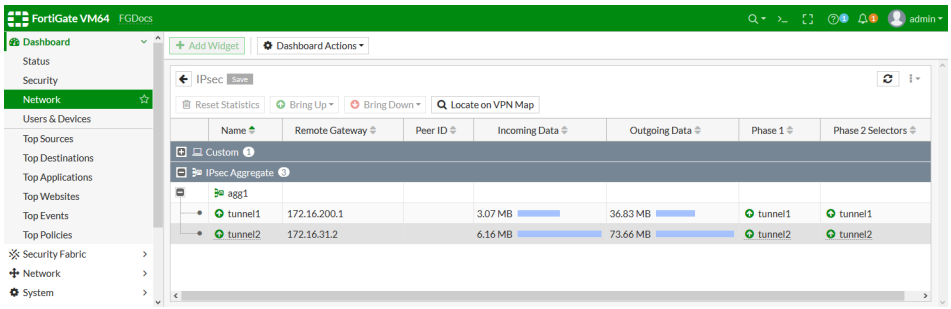


2. Go to *VPN* > *IPsec Tunnels* and click *Create New* > *IPsec Aggregate*.
3. Enter a name for the aggregate, such as *agg1*, and ensure that *Algorithm* is *Weighted Round Robin*.
4. Add *tunnel1* as an aggregate members, and set *Weight* to *10*.
5. Add *tunnel2* as a second aggregate members, and set its *Weight* to *20*.



6. Click *OK*.

7. To view and monitor the aggregate tunnel statistics, go to the *IPsec* widget on the *Network* dashboard.



To create the IPsec aggregate in the CLI:

1. Create the *tunnel1* and *tunnel2* custom IPsec tunnels with aggregate-member enabled and aggregate-weight set for both tunnels:

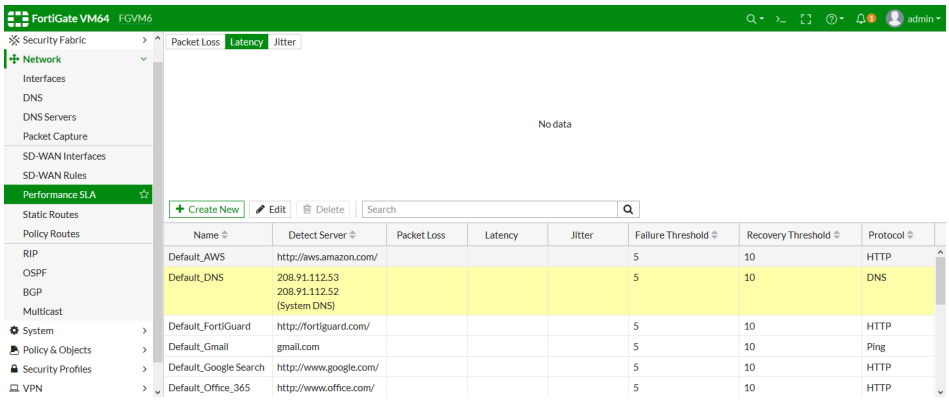
```
config vpn ipsec phase1-interface
    edit "tunnel1"
        ...
        set aggregate-member enable
        set aggregate-weight 10
        ...
    next
    edit "tunnel2"
        ...
        set aggregate-member enable
        set aggregate-weight 20
        ...
    next
end
```

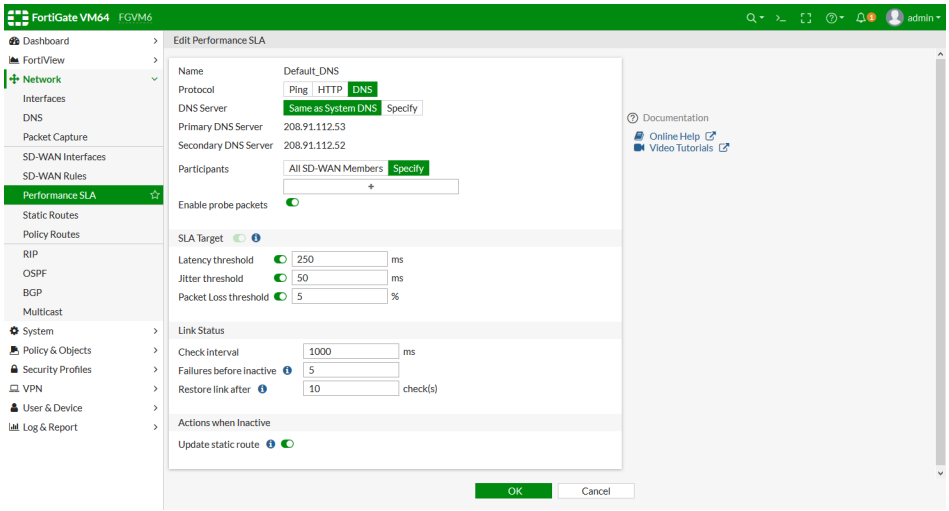
2. Create the IPsec aggregate:

```
config system ipsec-aggregate
    edit "agg1"
        set member "tunnel1" "tunnel2"
        set algorithm weighted-round-robin
    next
end
```

Default_DNS performance SLA profile

A new Default_DNS performance SLA is added after performing a factory reset.



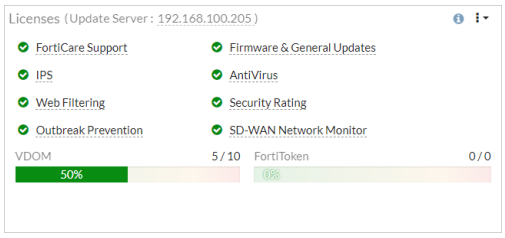


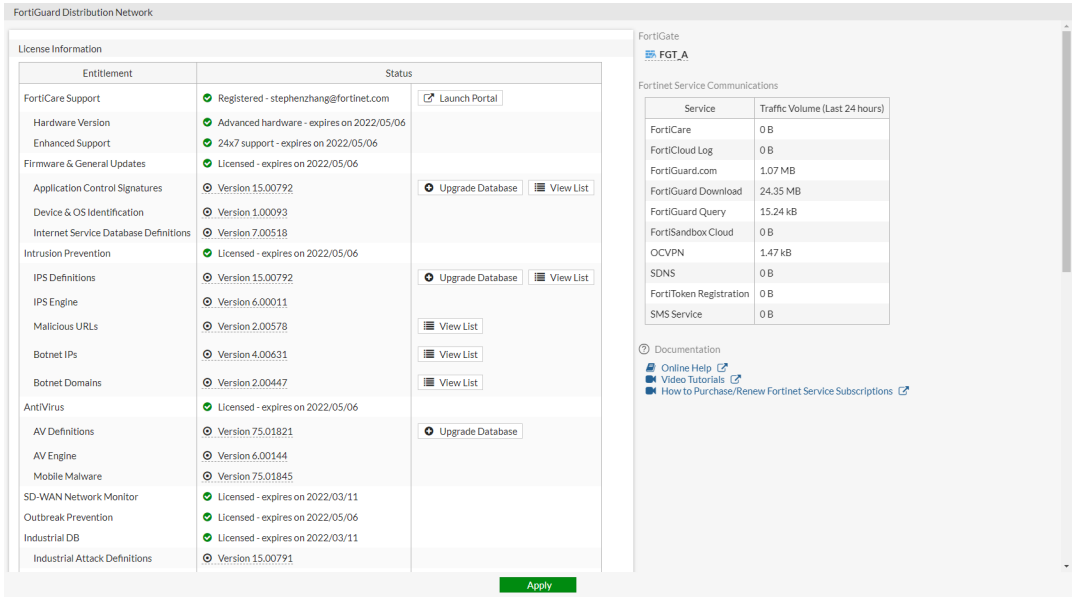

```
config system virtual-wan-link
    config health-check
        edit "Default_DNS"
            set system-dns enable
            set interval 1000
            set probe-timeout 1000
            set recoverytime 10
            config sla
                edit 1
                    set latency-threshold 250
                    set jitter-threshold 50
                    set packetloss-threshold 5
                next
            end
        next
    end
end
```

Interface speedtest

An interface speedtest can be performed on WAN interfaces in the GUI. The results of the test can be added to the interface's *Estimated bandwidth*. An SD-WAN Network Monitor license is required.

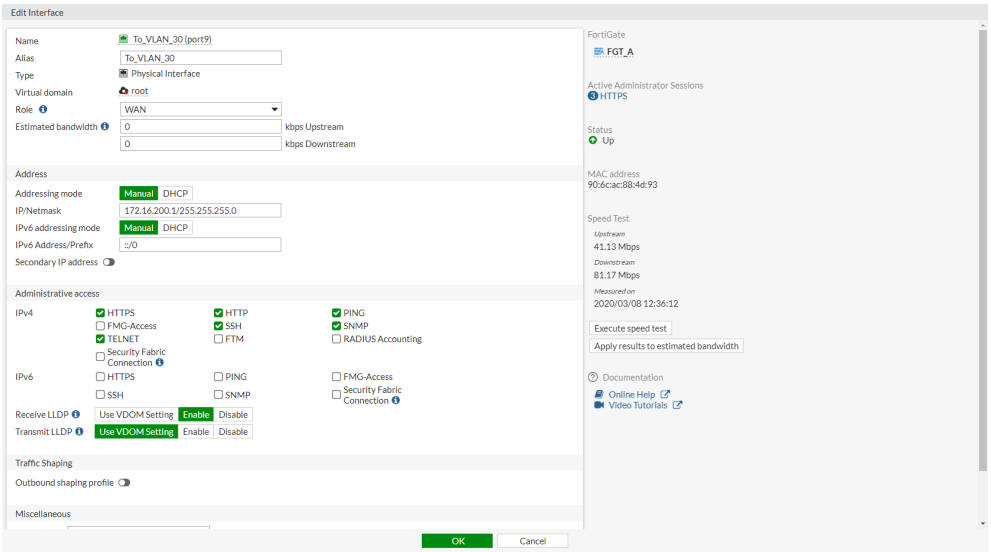
The *License* widget and the *System* > *FortiGuard* page display the SD-WAN Network Monitor license status.



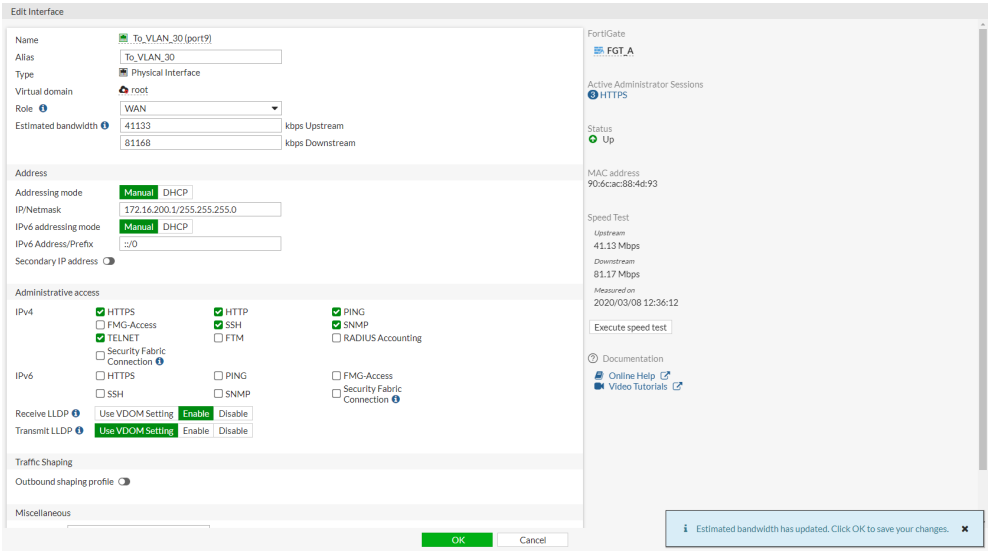


To run an interface speedtest in the GUI:

1. Go to *Network* > *Interfaces*.
2. Edit a WAN interface. The interfaces can be grouped by role using the grouping dropdown on the right side of the toolbar.
3. Click *Execute speed test* in the right pane.



4. When the test completes, click *Apply results to estimated bandwidth*.



The speedtest results are used to populate the *Estimated bandwidth* fields.

5. Click *OK*.

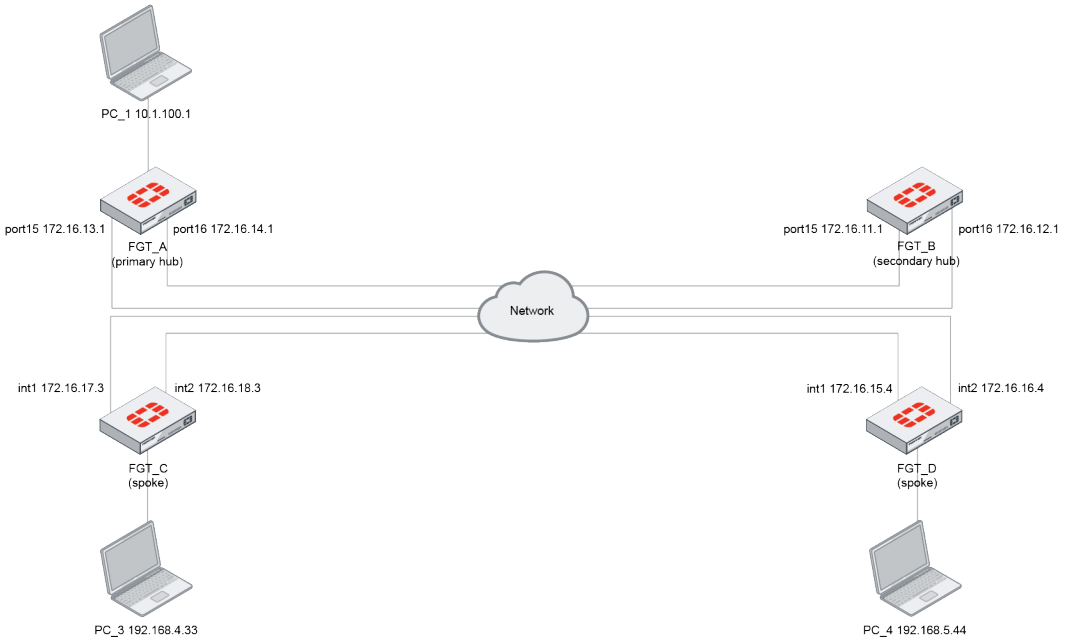


The FortiGate must be connected to FortiGuard, and able to reach either the AWS or Google speedtest servers.

Support SD-WAN integration with OCVPN

OCVPN now has the capability to enable SD-WAN in order to dynamically add its tunnel interfaces as SD-WAN members. Users can configure SD-WAN health checks and service rules to direct traffic over the OCVPN tunnels.

The following example uses a dual hub and spoke topology. Each hub and spoke has two WAN link connections to the ISP. The spokes generate two IPsec tunnels to each hub (four tunnels in total). BGP neighbors are established over each tunnel and routes from the hubs and other spokes learned from all neighbors, which forms an ECMP scenario. All tunnels are placed as SD-WAN members, so traffic can be distributed across tunnels based on the configured SD-WAN service rules.

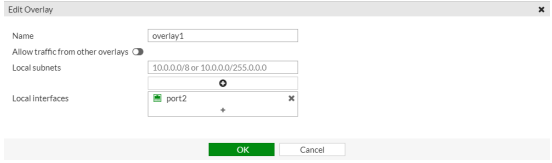


To integrate SD-WAN with OCVPN in the GUI:

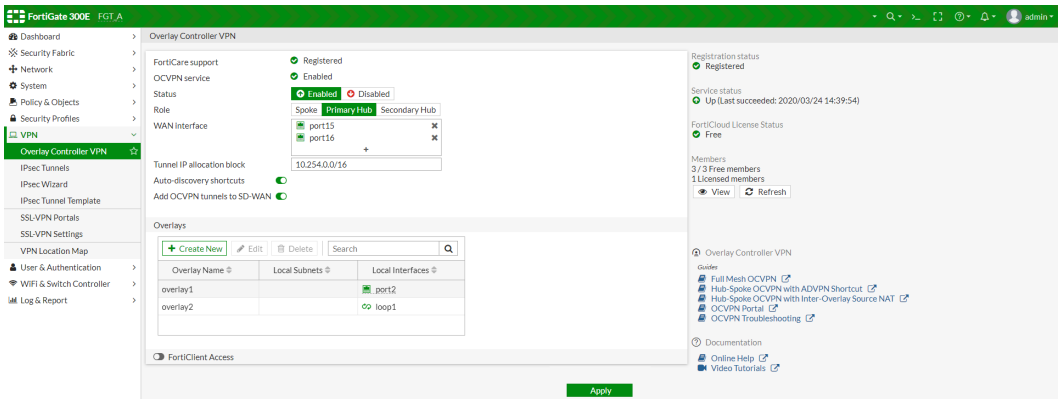
1. Configure the primary hub:
   a. Go to *VPN > Overlay Controller VPN* and set the *Status* to *Enable*.
   b. For *Role*, select *Primary Hub*.
   c. Enter the WAN interfaces (*port15* and *port16*) and tunnel IP allocation block (*10.254.0.0/16*).

   The WAN interface is position sensitive, meaning a tunnel will be created with the first position interface on the hub to the first position interface on the spoke, and so on. In this example, FGT_A (primary hub) will create two tunnels with FGT_C (spoke):
   - FGT_A port15 <==> FGT_C internal1
   - FGT_A port16 <==> FGT_C internal2

   d. Enable *Auto-discovery shortcuts*.
   e. Enable *Add OCVPN tunnels to SD-WAN*. The IPsec tunnels will be added automatically to the SD-WAN members if SD-WAN is enabled.
2. Configure the overlays on the primary hub:
   a. In the *Overlays* section, click *Create New*.
   b. Enter a name and add the local interface (*port2*). Note the overlay is either based on local subnets or local interfaces, but not both.
   By default, inter-overlay traffic is not enabled. Toggle *Allow traffic from other overlays* to enable it.
   c. Click *OK* and repeat these steps to create the second overlay (*loop1*).



d. Click *Apply*.



3. Configure the secondary hub with the same settings as the primary hub.

4. Configure the spoke:

a. Go to *VPN* > *Overlay Controller VPN* and set the *Status* to *Enable*.

b. For *Role*, select *Spoke*.

c. Enter the WAN interfaces (*internal1* and *internal2*).

d. Enable *Auto-discovery shortcuts*.

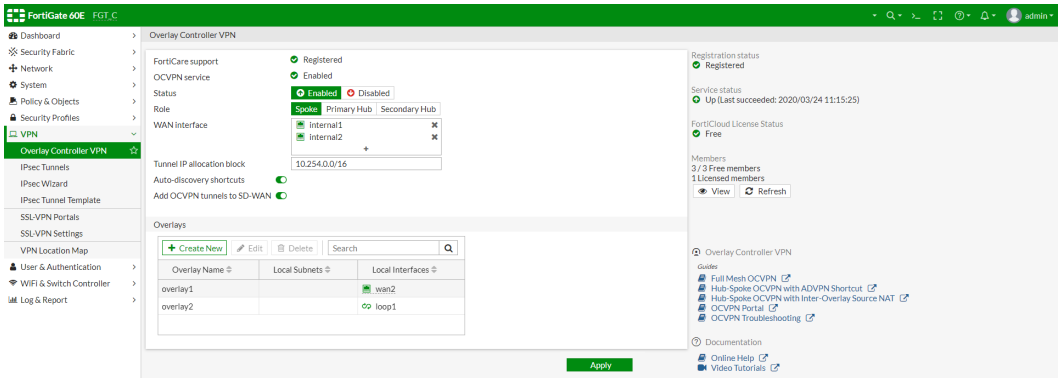
e. Enable *Add OCVPN tunnels to SD-WAN*. The IPsec tunnels will be added automatically to the SD-WAN members if SD-WAN is enabled.

f. Configure the overlays.



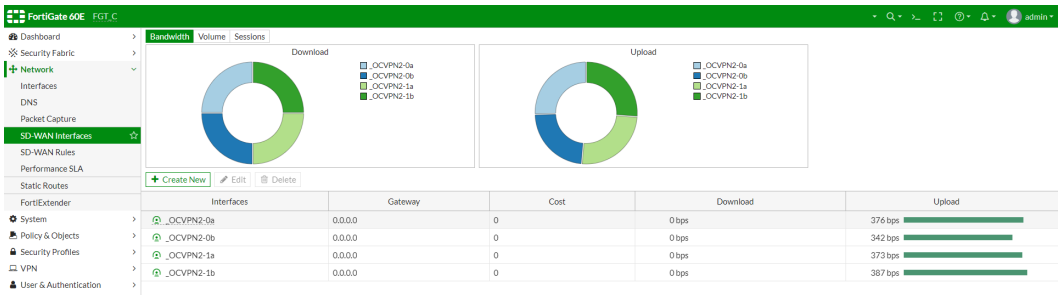
The overlay names on the spokes must match the hub for the traffic to be allowed through the same overlay.

g. Click *Apply*.



5. Configure the other spoke with the same settings.

6. On a spoke, go to *Network* > *SD-WAN Interfaces* to view the configuration generated by OCVPN.



Firewall policies will be automatically generated by OCVPN between the local interfaces and the SD-WAN interface. Each policy will define the proper local and remote networks for its source and destination addresses.

To integrate SD-WAN with OCVPN in the CLI:

1. Configure the primary hub:

```
config vpn ocvpn
    set role primary-hub
    set sdwan enable
    set wan-interface "port15" "port16"
    set ip-allocation-block 10.254.0.0 255.255.0.0
    config overlays
        edit "overlay1"
            config subnets
                edit 1
                    set type interface
                    set interface "port2"
                next
            end
        next
        edit "overlay2"
            config subnets
                edit 1
                    set type interface
                    set interface "loop1"
                next
            end
        next
    end
end
```

2. Configure the secondary hub with the same settings as the primary hub.
3. Configure the spoke:

```
config vpn ocvpn
    set status enable
    set sdwan enable
    set wan-interface "internal1" "internal2"
    config overlays
        edit "overlay1"
            config subnets
                edit 1
                    set type interface
                    set interface "wan2"
```

```
                next
            end
        next
        edit "overlay2"
            config subnets
                edit 1
                    set type interface
                    set interface "loop1"
                next
            end
        next
    end
end
```

4. Configure the other spoke with the same settings.
5. Configure SD-WAN:

```
config system virtual-wan-link
    set status enable
    config members
        edit 1
            set interface "_OCVPN2-0a"
        next
        edit 2
            set interface "_OCVPN2-0b"
        next
        edit 3
            set interface "_OCVPN2-1a"
        next
        edit 4
            set interface "_OCVPN2-1b"
        next
    end
end
```

Firewall policies will be automatically generated by OCVPN between the local interfaces and the SD-WAN interface. Each policy will define the proper local and remote networks for its source and destination addresses.

To verify the integration is working after the ADVPN shortcut is triggered:

1. Check the routing table on the spoke:

```
FGT_C # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 172.16.17.2, internal1
                  [10/0] via 172.16.18.2, internal2
B       10.1.100.0/24 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:24
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:24
B       10.1.200.0/24 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:24
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:24
```

```
B       10.2.100.0/24 [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
B       10.2.200.0/24 [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
B       10.254.0.0/16 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:15
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:15
                      [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
C       10.254.0.0/21 is directly connected, _OCVPN2-0a
C       10.254.0.1/32 is directly connected, _OCVPN2-0a
C       10.254.8.0/21 is directly connected, _OCVPN2-0b
C       10.254.8.1/32 is directly connected, _OCVPN2-0b
C       10.254.64.0/21 is directly connected, _OCVPN2-1a
C       10.254.64.1/32 is directly connected, _OCVPN2-1b_0 <==shortcut tunnel
C       10.254.64.2/32 is directly connected, _OCVPN2-1a
C       10.254.72.0/21 is directly connected, _OCVPN2-1b
C       10.254.72.2/32 is directly connected, _OCVPN2-1b
                       is directly connected, _OCVPN2-1b_0
C       172.16.17.0/24 is directly connected, internal1
C       172.16.18.0/24 is directly connected, internal2
C       172.16.200.0/24 is directly connected, wan1
C       192.168.1.0/24 is directly connected, internal
C       192.168.4.0/24 is directly connected, wan2
B       192.168.5.0/24 [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                       [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                       [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                       [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                       [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                       [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
                       [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                       [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
C       192.168.44.0/24 is directly connected, loop1
B       192.168.55.0/24 [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                        [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                        [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                        [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                        [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                        [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
                        [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                        [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
```

2. Check the VPN tunnel state:

```
FGT_C # diagnose vpn tunnel list

list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-1b_0 ver=2 serial=1c 172.16.18.3:0->172.16.15.4:0 dst_mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1 overlay_id=4

 parent=_OCVPN2-1b index=0
proxyid_num=1 child_num=0 refcnt=15 ilast=0 olast=0 ad=r/2
stat: rxp=641 txp=1025 rxb=16436 txb=16446
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1b proto=0 sa=1 ref=3 serial=1 auto-negotiate adr
```

```
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=42650/0B replaywin=1024
       seqno=407 esn=0 replaywin_lastseq=00000280 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=43186/43200
  dec: spi=90f03d9d esp=aes key=16 6cb33685bbc67d5c85488e0176ecf7b0
       ah=sha1 key=20 7d11b3babe62c840bf444b7b1f637b4324722a71
  enc: spi=7bc94bda esp=aes key=16 b4d8fc731d411eb24448b4077a5872ca
       ah=sha1 key=20 b724064d827304a6d80385ed4914461108b7312f
  dec:pkts/bytes=641/16368, enc:pkts/bytes=2053/123426
  npu_flag=03 npu_rgwy=172.16.15.4 npu_lgwy=172.16.18.3 npu_selid=1f dec_npuid=1 enc_
npuid=1
-----------------------------------------------------
name=_OCVPN2-0a ver=2 serial=18 172.16.17.3:0->172.16.13.1:0 dst_mtu=1500
bound_if=8 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1 overlay_id=1

proxyid_num=1 child_num=0 refcnt=20 ilast=0 olast=0 ad=r/2
stat: rxp=1665 txp=2922 rxb=278598 txb=70241
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=7
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0a proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41599/0B replaywin=1024
       seqno=890 esn=0 replaywin_lastseq=00000680 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=90f03d95 esp=aes key=16 a6ffcc197bb1b46ec745d0b595cdd69a
       ah=sha1 key=20 8007c134e41edf282f95daf9c9033d688ef05ccc
  enc: spi=a1bf21bf esp=aes key=16 ead05be389b0dec222f969e2f9c46b1d
       ah=sha1 key=20 b04105d34d4b0e61b018f2e60591f9b1510783bb
  dec:pkts/bytes=1665/278538, enc:pkts/bytes=4237/265074
  npu_flag=03 npu_rgwy=172.16.13.1 npu_lgwy=172.16.17.3 npu_selid=1b dec_npuid=1 enc_
npuid=1
-----------------------------------------------------
name=_OCVPN2-1a ver=2 serial=1a 172.16.17.3:0->172.16.11.1:0 dst_mtu=1500
bound_if=8 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1 overlay_id=3

proxyid_num=1 child_num=0 refcnt=17 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=2913 rxb=16376 txb=69642
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=5
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1a proto=0 sa=1 ref=28 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41653/0B replaywin=1024
       seqno=887 esn=0 replaywin_lastseq=00000002 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42900/43200
  dec: spi=90f03d9b esp=aes key=16 ee03f5b0f617a26c6177e91d60abf90b
       ah=sha1 key=20 f60cbbc4ebbd6d0327d23137da707b7ab2dc49e6
  enc: spi=a543a7d3 esp=aes key=16 1d37efab13a5c0347b582b2198b15cb8
       ah=sha1 key=20 427ee4c82bac6f26f0bcabfe04328c7f57ce682e
  dec:pkts/bytes=1/16316, enc:pkts/bytes=4229/264036
  npu_flag=03 npu_rgwy=172.16.11.1 npu_lgwy=172.16.17.3 npu_selid=1d dec_npuid=1 enc_
npuid=1
```

```
----------------------------------------------------
name=_OCVPN2-0b ver=2 serial=19 172.16.18.3:0->172.16.14.1:0 dst_mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1 overlay_id=2

proxyid_num=1 child_num=0 refcnt=20 ilast=0 olast=0 ad=r/2
stat: rxp=1665 txp=2917 rxb=278576 txb=69755
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=7
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0b proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41599/0B replaywin=1024
       seqno=88b esn=0 replaywin_lastseq=00000680 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=90f03d96 esp=aes key=16 9d7eb233c1d095b30796c3711d53f2fd
       ah=sha1 key=20 d8feacd42b5e0ba8b5e38647b2f2734c94644bd1
  enc: spi=a1bf21c0 esp=aes key=16 d2c0984bf86dc504c5475230b24034f0
       ah=sha1 key=20 3946e4033e1f42b0d9a843b94448f56fd5b57bee
  dec:pkts/bytes=1665/278516, enc:pkts/bytes=4233/264411
  npu_flag=03 npu_rgwy=172.16.14.1 npu_lgwy=172.16.18.3 npu_selid=1c dec_npuid=1 enc_
npuid=1
----------------------------------------------------
name=_OCVPN2-1b ver=2 serial=1b 172.16.18.3:0->172.16.12.1:0 dst_mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1 overlay_id=4

proxyid_num=1 child_num=1 refcnt=19 ilast=1 olast=0 ad=r/2
stat: rxp=1 txp=2922 rxb=16430 txb=70173
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=4
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1b proto=0 sa=1 ref=28 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41656/0B replaywin=1024
       seqno=890 esn=0 replaywin_lastseq=00000002 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42903/43200
  dec: spi=90f03d9c esp=aes key=16 a655767c1ed6cff4575857eb3981ad81
       ah=sha1 key=20 bfc2bccd7103a201be2641d4c6147d437d2c3f70
  enc: spi=a543a7d4 esp=aes key=16 7221b814e483165b01edfdc8260d261a
       ah=sha1 key=20 d54819643c2f1b20da2aea4282d50a1f1bc1d72a
  dec:pkts/bytes=1/16370, enc:pkts/bytes=4238/265164
  npu_flag=03 npu_rgwy=172.16.12.1 npu_lgwy=172.16.18.3 npu_selid=1e dec_npuid=1 enc_
npuid=1
```

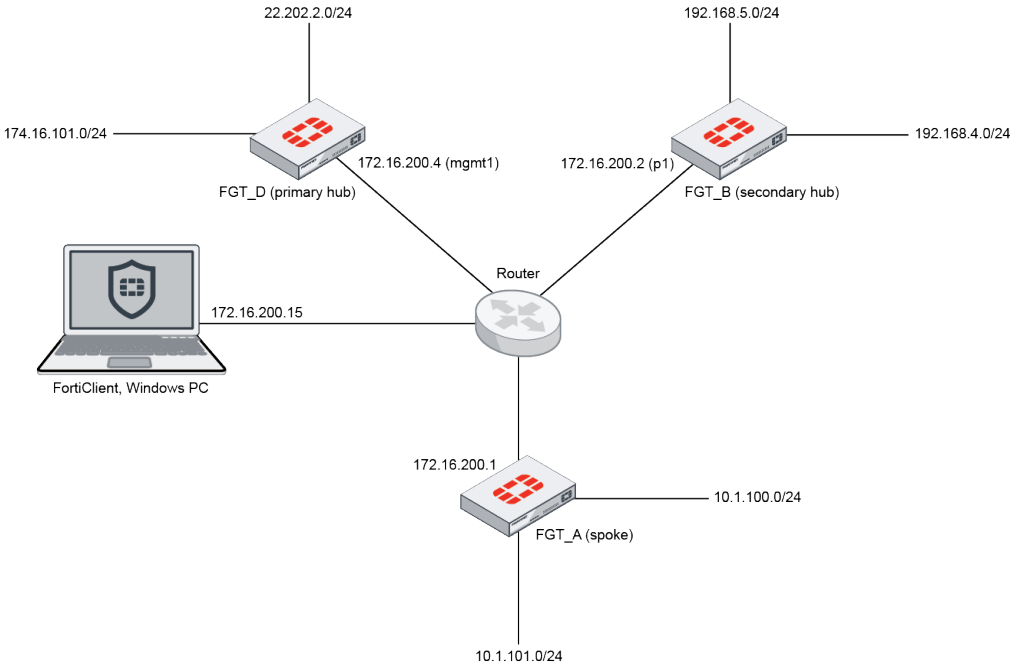
3. Check the SD-WAN state:

```
FGT_C # diagnose system virtual-wan-link health-check
Health Check(Default_DNS):
Health Check(Default_Office_365):
Health Check(Default_Gmail):
Health Check(Default_AWS):
Health Check(Default_Google Search):
Health Check(Default_FortiGuard):
Health Check(ocvpn):
Seq(1 _OCVPN2-0a): state(alive), packet-loss(0.000%) latency(0.364), jitter(0.028) sla_
map=0x0
```

```
Seq(2 _OCVPN2-0b): state(alive), packet-loss(0.000%) latency(0.287), jitter(0.026) sla_
map=0x0
Seq(3 _OCVPN2-1a): state(dead), packet-loss(100.000%) sla_map=0x0
Seq(4 _OCVPN2-1b): state(dead), packet-loss(100.000%) sla_map=0x0
Seq(4 _OCVPN2-1b_0): state(alive), packet-loss(0.000%) latency(0.289), jitter(0.029) sla_
map=0x0
```

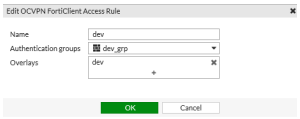
Allow FortiClient to join OCVPN

Administrators can configure remote access for FortiClient within an OCVPN hub. This provides simple configurations to allow a user group access to an overlay network.



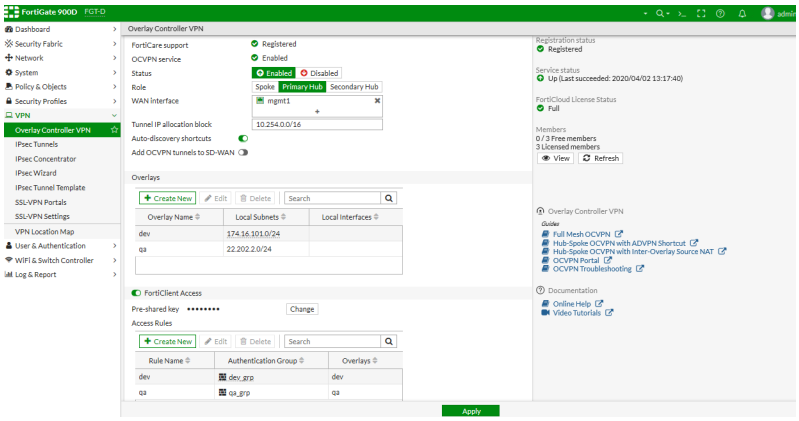
To configure remote FortiClient access to an OCVPN hub in the GUI:

1. On the primary hub, configure the users and user groups required for the FortiClient dialup user authentication and authorization. In this example, there are two user groups (*dev_grp* and *qa_grp*).
2. Go to *VPN* > *Overlay Controller VPN* and in the *Overlays* section, click *Create New*.
3. Enter a name and the local subnet (174.16.101.0/24 for *dev* and 22.202.2.0/24 for *qa*).
4. Enable *FortiClient Access*.
5. In the *Access Rules* section, click *Create New*.
6. Enter a name, and select the authentication groups and overlays.



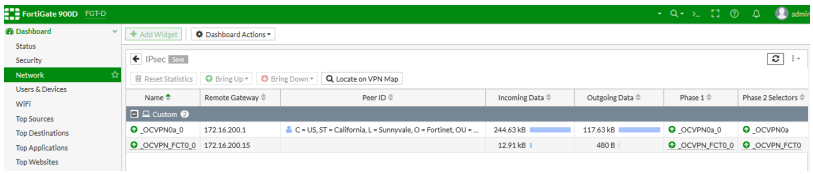
   The authentication groups will be used by the IPsec phase 1 interface for authentication, and by firewall policies for authorization. The overlay allows access to the resource.
7. Click *OK*.
8. Create more rules if needed.

9. Click *Apply*.



To view the tunnel status and activity in the GUI:

1. Go to *Dashboard* > *Network*.
2. Click the *IPsec* widget to expand to full screen view.



To configure remote FortiClient access to an OCVPN hub in the CLI:

```
config vpn ocvpn
    set status enable
    set role primary-hub
    set wan-interface "mgmt1"
    set ip-allocation-block 10.254.0.0 255.255.0.0
    config overlays
        edit "dev"
            config subnets
                edit 1
                    set subnet 174.16.101.0 255.255.255.0
                next
            end
        next
        edit "qa"
            config subnets
                edit 1
                    set subnet 22.202.2.0 255.255.255.0
                next
            end
        next
    end
    config forticlient-access
        set status enable
        set psksecret xxxxxxxxxxxx
        config auth-groups
            edit "dev"
```

```
                set auth-group "dev_grp"
                set overlays "dev"
            next
            edit "qa"
                set auth-group "qa_grp"
                set overlays "qa"
            next
        end
    end
end
```

To view the tunnel status and activity in the CLI:

```
# diagnose vpn ike gateway list

vd: root/0
name: _OCVPN_FCT0_0
version: 1
interface: mgmt1 4
addr: 172.16.200.4:4500 -> 172.16.200.15:64916
created: 110s ago
xauth-user: usera
groups:
  dev_grp 1
assigned IPv4 address: 10.254.128.1/255.255.255.255
nat: peer
IKE SA: created 1/1  established 1/1  time 20/20/20 ms
IPsec SA: created 1/1  established 1/1  time 0/0/0 ms

  id/spi: 72 1ccd2abf2d981123/fd8da107f9e4d312
  direction: responder
  status: established 110-110s ago = 20ms
  proposal: aes256-sha256
  key: 105a0291b0c05219-3decdf78938a7bea-78943651e1720536-625114d66e46f668
  lifetime/rekey: 86400/86019
  DPD sent/recv: 00000000/00000af3
```

To view data on the PC running FortiClient:

```
C:\ route print
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0       10.1.100.5      10.1.100.13    281
       10.1.100.0    255.255.255.0     10.254.128.2     10.254.128.1      1
      10.1.100.13  255.255.255.255         On-link       10.1.100.13    281
       10.1.101.0    255.255.255.0     10.254.128.2     10.254.128.1      1
        10.6.30.0    255.255.255.0         On-link        10.6.30.13    281
       10.6.30.13  255.255.255.255         On-link        10.6.30.13    281
      10.6.30.255  255.255.255.255         On-link        10.6.30.13    281
       10.254.0.0      255.255.0.0     10.254.128.2     10.254.128.1      1
     10.254.128.1  255.255.255.255         On-link      10.254.128.1    257
       22.202.2.0    255.255.255.0     10.254.128.2     10.254.128.1      1
```

```
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
     172.16.200.4  255.255.255.255       10.1.100.5      10.1.100.13     25
     174.16.101.0    255.255.255.0     10.254.128.2     10.254.128.1      1
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link      10.254.128.1    257
        224.0.0.0        240.0.0.0         On-link        10.6.30.13    281
        224.0.0.0        240.0.0.0         On-link       10.1.100.13    281
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link      10.254.128.1    257
  255.255.255.255  255.255.255.255         On-link        10.6.30.13    281
  255.255.255.255  255.255.255.255         On-link       10.1.100.13    281
===========================================================================
Persistent Routes:
  Network Address          Netmask  Gateway Address  Metric
          0.0.0.0          0.0.0.0       10.1.100.5  Default
```

The PC can access the *dev* resource overlay, but not *qa*:

```
C:\Users\tester>ping 174.16.101.44

Pinging 174.16.101.44 with 32 bytes of data:
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63

Ping statistics for 174.16.101.44:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\tester>ping 22.202.2.2

Pinging 22.202.2.2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 22.202.2.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Secure access

This section includes secure access features added to FortiOS:

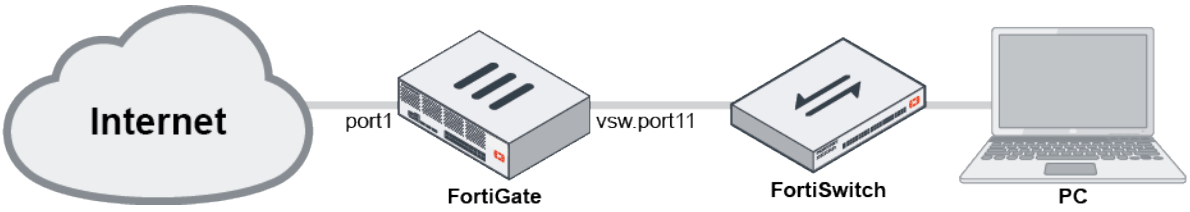
- Switch controller - quarantine by redirect on page 85
- Wireless IPv6 support on page 87
- Support for spectrum analysis of FortiAP E models on page 93
- Increase in maximum number of managed FortiAPs on page 99
- VLAN interface templates for FortiSwitch devices on page 100

- Improved FortiSwitch support on page 104
- Even distribution of FortiAP reports on page 104
- View detailed information for individual WiFi connections on page 107
- VLAN probe report on page 116
- FortiAP client load balancing per AP on page 120
- Layer three ACL configurations for Wireless APs on page 121

Switch controller - quarantine by redirect

Quarantine by redirect makes the FortiSwitch redirect traffic from the quarantined host to the FortiGate, keeping the device on its original network. This is the default quarantine mode.

Quarantine by VLAN, which moves the device from the normal switch VLAN to the quarantine VLAN, can be complicated for administrators that use DHCP or static IP address assignments. When a device is sent to quarantine, its IP address is no longer valid for the quarantined VLAN segment, making it difficult to perform remediation on the device.



In this example, the PC can access the internet when there is an allowed policy from interface vsw.port11 to port1 (called *PC to Internet*). When the PC is quarantined, a firewall address is automatically created for the PC, which is added to an automatically created address group called *QuarantinedDevices*. A policy (called *quarantine*) is created that applies to this address group and blocks traffic from the PC to the internet.

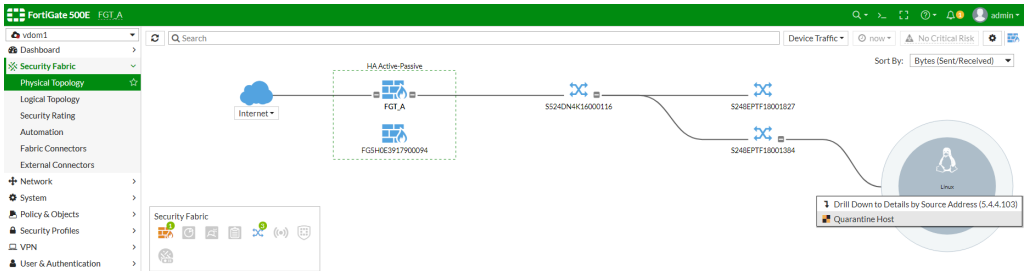
The FortiSwitch configuration is done automatically after the FortiGate configured.

To configure the quarantine mode:

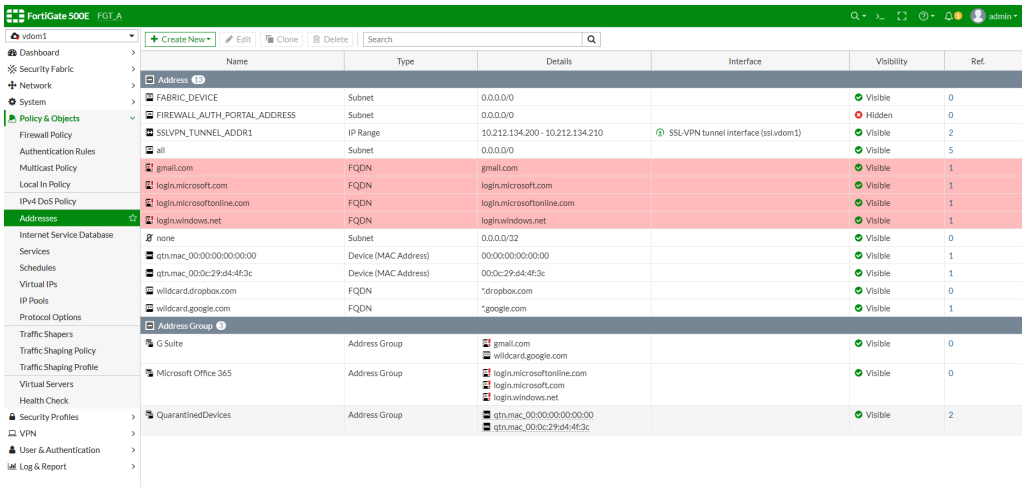
```
config switch-controller global
    set quarantine-mode {by-vlan | by-redirect (default)}
end
```

To quarantine an active device, based on the device's MAC address, in the GUI:

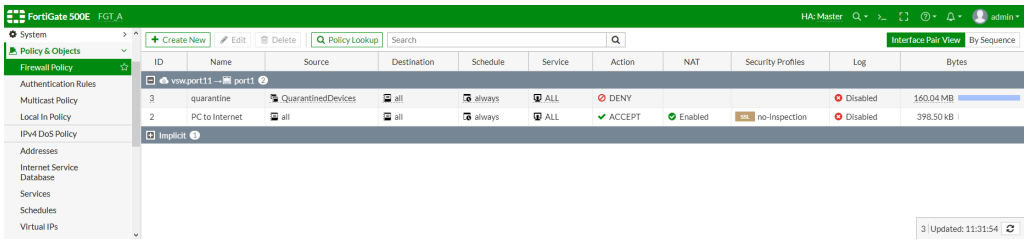
1. Go to *Security Fabric* > *Physical Topology* or *Security Fabric* > *Logical Topology*.
2. Mouse over the bubble of an active device, and select *Quarantine Host* from the right-click menu.



3. Click *OK* in the *Quarantine Host* page to quarantine the device.
   Firewall addresses and an address group (*QuarantinedDeivces*) are automatically added for the quarantined devices.



4. Go to *Policy & Objects* > *Firewall Policy* and create a policy to block traffic from quarantined devices to the internet.



To quarantine an active device, based on the device's MAC address, in the CLI:

```
config user quarantine
    set traffic-policy quarantine
    set firewall-groups "QuarantinedDevices"
    config targets
        edit "manual-qtn-1"
            set description "Manually quarantined"
            config macs
                edit 00:0c:29:d4:4f:3c
                    set description "manual-qtn"
                    set drop disable
                next
            end
        next
    end
end
```

Firewall addresses are automatically created for the quarantined MAC address, and the addresses are added to the *QuarantinedDevices* address group:

```
# show firewall address | grep -f qtn
config firewall address
    edit "qtn.mac_00:00:00:00:00:00" <---
        set uuid 9069e73c-3c6e-51ea-28d4-b807167fdcb7
```

```
        set type mac
        set comment "Quarantine dummy MAC to keep the addrgrp"
    next
    edit "qtn.mac_00:0c:29:d4:4f:3c" <---
        set uuid 869847ce-3c84-51ea-59c2-964152415e22
        set type mac
        set start-mac 00:0c:29:d4:4f:3c
        set end-mac 00:0c:29:d4:4f:3c
        set comment "Quarantine MAC"
    next
end
```

```
# show firewall addrgrp | grep -f Quarantined
config firewall addrgrp
    edit "QuarantinedDevices" <---
        set uuid 9069d332-3c6e-51ea-17e1-cab3dd4dde6c
        set member "qtn.mac_00:00:00:00:00:00" "qtn.mac_00:0c:29:d4:4f:3c"
    next
end
```

To view the automatic configuration changes on the FortiSwitch:

```
config switch quarantine
    edit 00:0c:29:d4:4f:3c
        set acl-id 2
        set cos-queue 0
        set description "manual-qtn "
        set policer 1
    next
end
config switch acl ingress
    edit 2
        config action
            set cos-queue 0
            set count enable
            set policer 1
        end
        config classifier
            set src-mac 00:0c:29:d4:4f:3c
        end
        set ingress-interface-all enable
    next
end
```

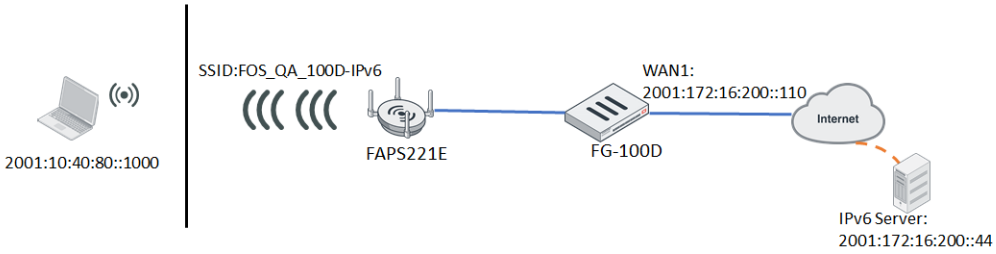
Wireless IPv6 support

Wireless client IPv6 traffic is supported from both tunnel and local bridge mode SSID:

- Tunnel mode SSID IPv6 traffic on page 88
- Local bridge mode SSID IPv6 traffic on page 90
- CLI commands for IPv6 rules on page 92

Tunnel mode SSID IPv6 traffic

In the following example, FortiAP S221E is managed by FortiGate 100D and broadcasts tunnel mode SSID:FOS_QA_100D-IPv6.



To configure a WiFi client accessing IPv6 tunnel mode traffic:

1. Create a tunnel mode VAP:

```
config wireless-controller vap
    edit "wifi4"
        set ssid "FOS_QA_100D-IPv6"
        set passphrase ********
        set schedule "always"
    next
end
```

2. Create an IPv6 address for the VAP with DHCP enabled:

```
config system interface
    edit "wifi4"
        set vdom "vdom1"
        set ip 10.40.80.1 255.255.255.0
        set allowaccess ping https http
        set type vap-switch
        set alias "vdom1:"
        set device-identification enable
        set role lan
        set snmp-index 36
        config ipv6
            set ip6-address 2001:10:40:80::1/64
            set ip6-allowaccess ping https http
            set ip6-send-adv enable
            set ip6-manage-flag enable
            set ip6-other-flag enable
        end
    next
end

config system dhcp6 server
    edit 1
        set subnet 2001:10:40:80::/64
        set interface "wifi4"
        config ip-range
            edit 1
                set start-ip 2001:10:40:80::1000
```

```
                set end-ip 2001:10:40:80::1100
            next
        end
    next
end
```

3. Create an IPv6 policy from the VAP to WAN1:

```
config firewall policy
    edit 1
        set name "ipv6"
        set srcintf "wifi4"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
end
```

4. Verify the IPv6 address in the station list:
   a. In the FortiGate CLI:

```
# diagnose wireless-controller wlac -d sta online
   vf=4 wtp=3 rId=1 wlan=wifi4 vlan_id=0 ip=10.40.80.2 ip6=2001:10:40:80::1000
mac=b4:ae:2b:cb:d1:72 vci=MSFT 5.0 host=DESKTOP-DO33HQP user= group= signal=-29 noise=-
93 idle=1 bw=48 use=5 chan=6 radio_type=11N security=wpa2_only_personal mpsk=default
encrypt=aes cp_authed=no online=yes mimo=2
                ip6=fe80::c5c5:6c09:8021:d2d0,88, *2001:10:40:80::1000,8,
```

   b. In the FortiAP CLI:

```
FortiAP-S221E # sta
wlan00 (FOS_QA_100D-IPv6) client count 1
    MAC:b4:ae:2b:cb:d1:72 ip:10.40.80.2 ip_proto:dhcp ip_age:84 host:DESKTOP-DO33HQP
vci:MSFT 5.0
                          ip6:fe80::c5c5:6c09:8021:d2d0 ip6_proto:arp ip6_age:2 ip6_
rx:101
                          ip6:2001:10:40:80::1000 ip6_proto:dhcp ip6_age:82 ip6_rx:20
        vlanid:0 Auth:Yes channel:6 rate:130Mbps rssi:65dB idle:0s
        Rx bytes:256951 Tx bytes:53947 Rx rate:130Mbps Tx rate:130Mbps Rx last:0s Tx
last:0s
        AssocID:1 Mode:  Normal Flags:f PauseCnt:0
        KEY type=aes_ccm pad=0 keyix=65535 keylen=16 flags=3(xmit recv) RSC=0 TSC=0
            e7 6f 05 ce   06 e1 4a 9b   3a d4 4f 43   1f 57 bb 49
            00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
            00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
        KEY type=aes_ccm pad=0 keyix=1 keylen=16 flags=83(xmit recv dflt) RSC=0 TSC=0
            01 47 6f 21   9b ac 73 4b   7c ae 07 66   7e 5a c6 7e
            00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
            00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
FortiAP-S221E #

FortiAP-S221E # usta
```

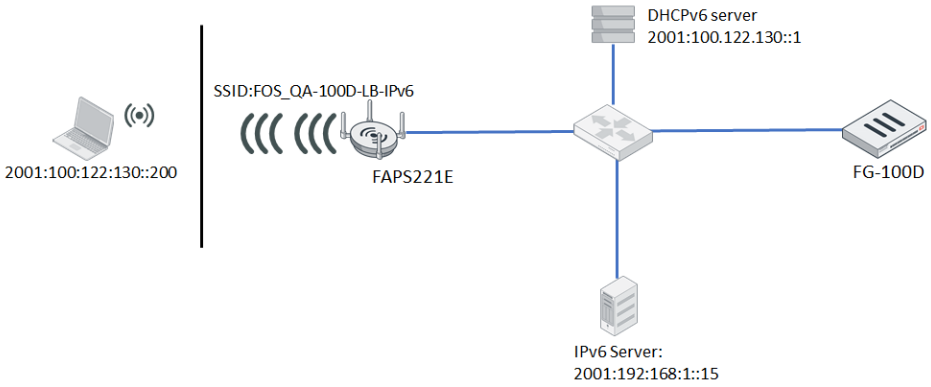
```
WTP daemon STA info:

  1/1   b4:ae:2b:cb:d1:72 00:00:00:00:00:00 vId=0    type=wl----sta,  vap=wlan00,FOS_
QA_100D-IPv6(0) mpsk=default  ip=10.40.80.2/1  host=DESKTOP-DO33HQP vci=MSFT 5.0
os=Windows
                          ip6=fe80::c5c5:6c09:8021:d2d0/2 rx=101
                          ip6=2001:10:40:80::1000/1 rx=21
                          replycount=0000000000000002

Total STAs: 1
```

Local bridge mode SSID IPv6 traffic

In the following example, FortiAP S221E is managed by FortiGate 100D through a local NATed switch and broadcasts local bridge mode SSID:FOS_QA_100D-LB-IPv6.



To configure a WiFi client accessing IPv6 local bridge mode traffic:

1. Create a local bridge mode VAP:

```
config wireless-controller vap
    edit "test1"
        set ssid "FOS_QA-100D-LB-IPv6"
        set passphrase ********
        set local-bridging enable
        set schedule "always"
    next
end
```

2. Create an IPv6 DHCP server for the local NATed switch (FortiWiFi 60E is used in this example):

```
config system interface
    edit "internal6"
        set vdom "vdom1"
        set ip 2.2.3.1 255.255.255.0
        set allowaccess ping https http fabric
        set type physical
        set snmp-index 18
        config ipv6
            set ip6-address 2001:100:122:130::1/64
            set ip6-allowaccess ping https http fabric
```

```
            set ip6-send-adv enable
            set ip6-manage-flag enable
            set ip6-other-flag enable
        end
    next
end

config system dhcp6 server
    edit 1
        set subnet 2001:100:122:130::/64
        set interface "internal6"
        config ip-range
            edit 1
                set start-ip 2001:100:122:130::200
                set end-ip 2001:100:122:130::300
            next
        end
    next
end
```

3. Create an IPv6 policy for the local NATed switch:

```
config firewall policy
    edit 2
        set name "ipv6"
        set srcintf "internal6"
        set dstintf "internal7"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
end
```

4. Verify the IPv6 address in the station list:
   a. In the FortiGate CLI:

```
# diagnose wireless-controller wlac -d sta online
    vf=4 wtp=3 rId=2 wlan=test1 vlan_id=0 ip=2.2.3.3 ip6=2001:100:122:130::200
mac=f0:98:9d:76:64:c4 vci= host=iPhoneX user= group= signal=-41 noise=-105 idle=18 bw=0
use=5 chan=36 radio_type=11AC security=wpa2_only_personal mpsk=default encrypt=aes cp_
authed=no online=yes mimo=2
                ip6=fe80::82a:9eba:69c5:5454,13, *2001:100:122:130::200,2,
```

   b. In the FortiAP CLI:

```
FortiAP-S221E # sta
wlan10 (FOS_QA-100D-LB-IPv6) client count 1
    MAC:f0:98:9d:76:64:c4 ip:2.2.3.3 ip_proto:dhcp ip_age:8 host:iPhoneX vci:
                          ip6:fe80::82a:9eba:69c5:5454 ip6_proto:arp ip6_age:1 ip6_
rx:12
                          ip6:2001:100:122:130::200 ip6_proto:dhcp ip6_age:8 ip6_rx:2
        vlanid:0 Auth:Yes channel:36 rate:173Mbps rssi:64dB idle:0s
        Rx bytes:26654 Tx bytes:27949 Rx rate:78Mbps Tx rate:173Mbps Rx last:0s Tx
last:0s
```

```
            AssocID:1 Mode:  Normal Flags:1000000b PauseCnt:0
            KEY type=aes_ccm pad=0 keyix=65535 keylen=16 flags=3(xmit recv) RSC=0 TSC=0
                83 25 7e 72   d2 b1 d2 ef   30 9f 6e 9f   50 e5 6f 5a
                00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
                00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
            KEY type=aes_ccm pad=0 keyix=1 keylen=16 flags=83(xmit recv dflt) RSC=0 TSC=0
                1f 25 64 3e   02 4d e2 f1   2c b0 5e 03   ed 99 a4 47
                00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
                00 00 00 00   00 00 00 00   00 00 00 00   00 00 00 00
    FortiAP-S221E #

    FortiAP-S221E # usta

    WTP daemon STA info:

     1/1   f0:98:9d:76:64:c4 00:00:00:00:00:00 vId=0    type=wl----sta,  vap=wlan10,FOS_QA-
    100D-LB-IPv6(0) mpsk=default  ip=2.2.3.3/1  host=iPhoneX vci= os=iOS
                              ip6=fe80::82a:9eba:69c5:5454/2 rx=12
                              ip6=2001:100:122:130::200/1 rx=2
                              replycount=0000000000000002

    Total STAs: 1
```

CLI commands for IPv6 rules

The following IPv6 rules can be used in VAP configurations:

Command	Description
drop-icmp6ra	Drop ICMPv6 router advertisement (RA) packets that originate from wireless clients.
drop-icmp6rs	Drop ICMPv6 router solicitation (RS) packets to be sent to wireless clients.
drop-llmnr6	Drop Link-Local Multicast Name Resolution (LLMNR) packets.
drop-icmp6mld2	Drop ICMPv6 Multicast Listener report V2 (MLD2) packets.
drop-dhcp6s	Drop DHCPv6 server generated packets that originate from wireless clients.
drop-dhcp6c	Drop DHCPv6 client generated packets to be sent to wireless clients.
ndp-proxy	Enable IPv6 NDP proxy; send back NA on behalf of the client and drop the NS.
drop-ns-dad	Drop ICMPv6 NS DAD when target address is not found in the NDP proxy cache.
drop-ns-nondad	Drop ICMPv6 NS non-DAD when target address is not found in the NDP proxy cache.

To configure IPv6 rules on a VAP:

```
config wireless-controller vap
    edit "wifi4"
        set ssid "FOS_QA_100D-IPv6"
        set passphrase ********
        set schedule "always"
```

```
        set ipv6-rules drop-icmp6ra drop-icmp6rs drop-llmnr6 drop-icmp6mld2 drop-dhcp6s drop-
dhcp6c ndp-proxy drop-ns-dad drop-ns-nondad
    next
end
```

The IPv6 rules settings can be pushed to a FortiAP when the VAP is broadcast.

To view the pushed settings on the FortiAP:

```
FortiAP-S221E # iwpriv wlan00 get_bmcs6
wlan00    get_bmcs6:991  (0x3df)
00000001 icmp6-ra            : yes
00000002 icmp6-rs            : yes
00000004 dhcp6-server        : yes
00000008 dhcp6-client        : yes
00000010 llmnr               : yes
00000040 icmp6-mld2          : yes
00000080 ndp-proxy           : yes
00000100 ns-dad              : yes
00000200 ns-nondad           : yes
```

Support for spectrum analysis of FortiAP E models

Spectrum analysis is available for FortiAP E models running 6.4.0 and later firmware. The analysis is visible in the GUI through the *Managed FortiAPs* page. Spectrum analysis can also be performed in the CLI.

To start or stop the spectrum analysis:

```
execute wireless-controller spectral-scan <wtp-id> <radio-id > <on | off>
<duration> <channel> <report-interval>
```

To verify the results:

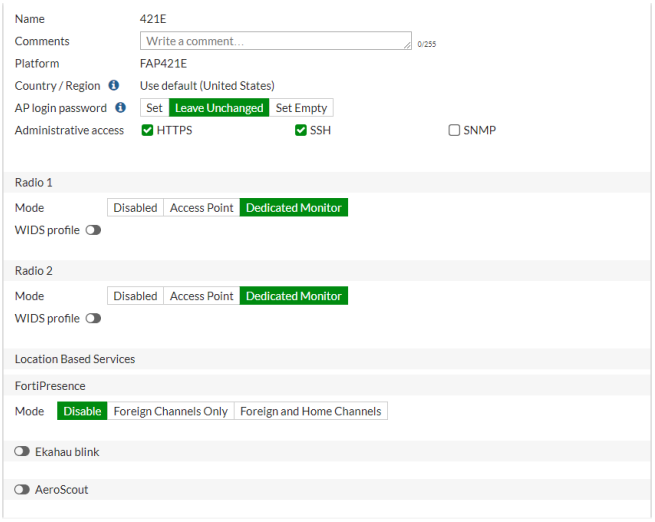
```
diagnose wireless-controller wlac -c rf-sa <wtp-id> <radio-id> <channel>
```

```
get wireless-controller spectral-info <wtp-id> <radio-id>
```

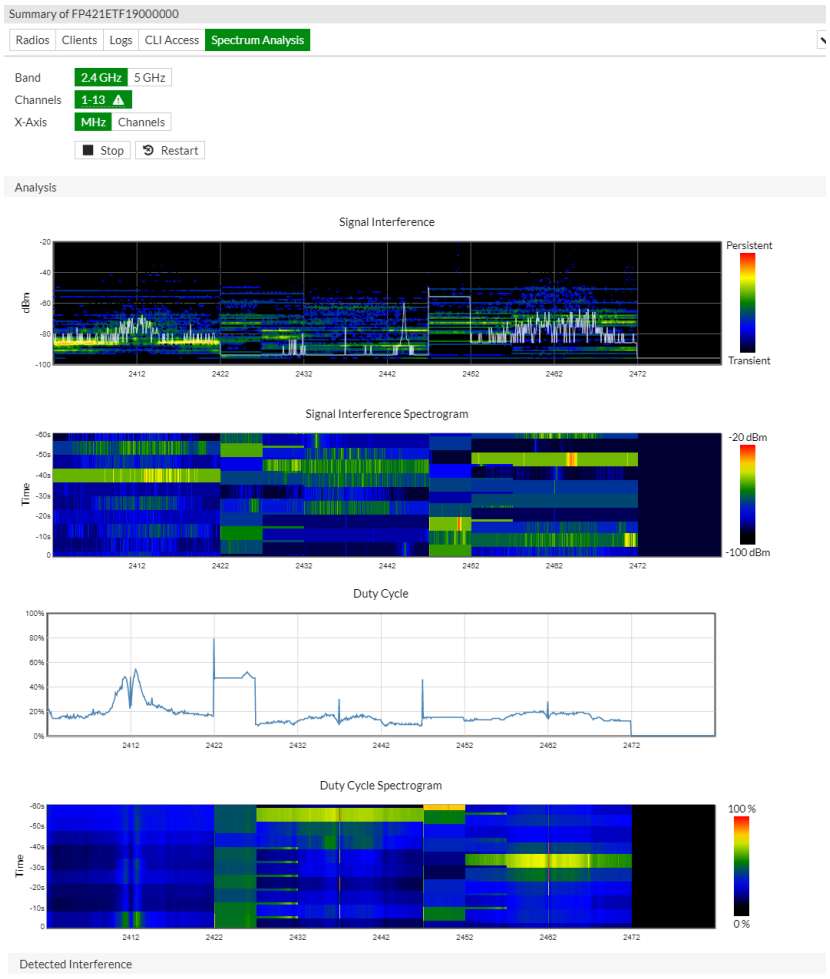
The following examples use a FortiAP 421E (radio 1 at 2.4 GHz and radio 2 at 5 GHz) that is managed by a FortiGate 80E-POE.

To view spectrum analysis in the GUI:

1. Change the radio mode:
   a. Go to *WiFi & Switch Controller* > *FortiAP Profiles* and double-click the FortiAP to edit the profile.
   b. In the *Radio 1* and *Radio 2* sections for *Mode*, select *Dedicated Monitor*.

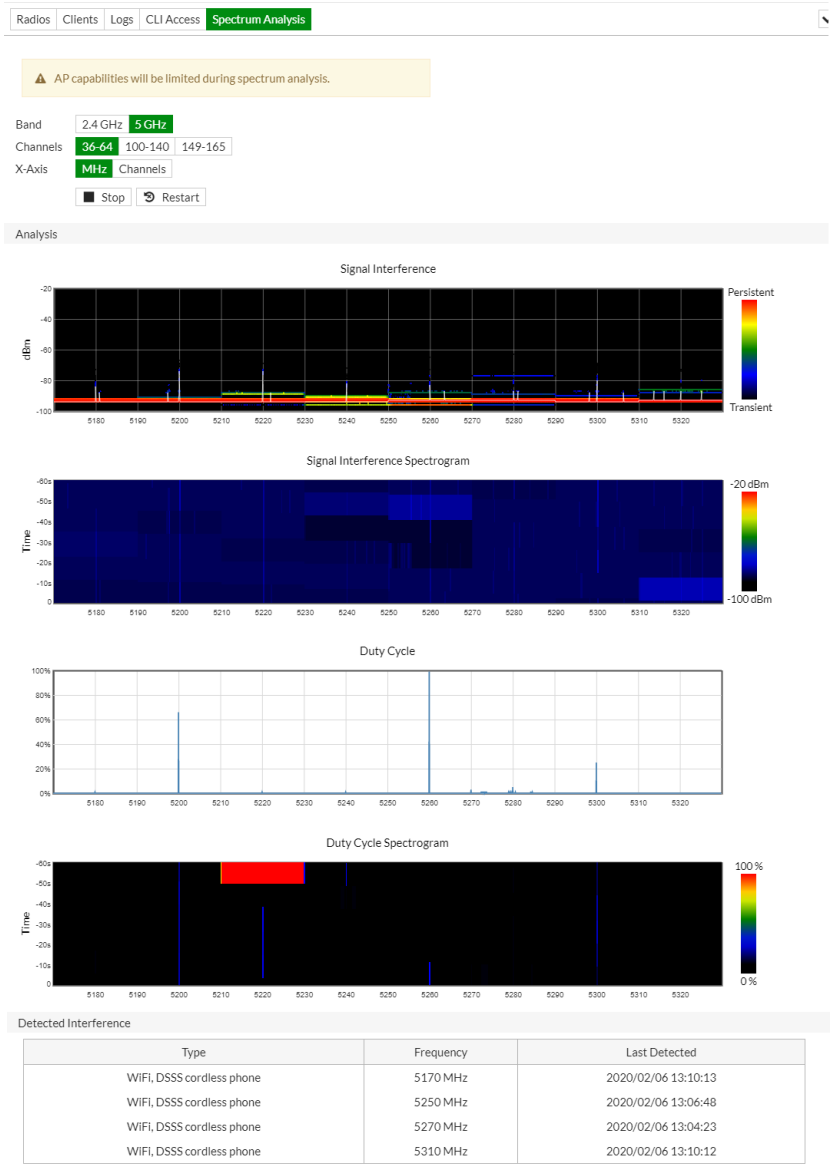


   c. Click *OK*.
2. Go to *WiFi & Switch Controller* > *Managed FortiAPs*.
3. In the table, hover over the AP so the context menu appears and click *Details*. The summary pane appears.
4. Click *Spectrum Analysis*.
5. Click a band frequency to view the analysis for: *Signal Interference*, *Signal Interference Spectrogram*, *Duty Cycle*, *Duty Cycle Spectrogram*, and *Detected Interference* (list).
   Analysis for 2.4 GHz:



Type	Frequency	Last Detected
WiFi, DSSS cordless phone	2402 MHz	2020/02/06 13:00:29
Bluetooth, FHSS cordless phone	2402 MHz	2020/02/06 12:53:43
Bluetooth, FHSS cordless phone	2412 MHz	2020/02/06 13:00:30
WiFi, DSSS cordless phone	2412 MHz	2020/02/06 13:00:30
WiFi, DSSS cordless phone	2427 MHz	2020/02/06 13:00:25
Bluetooth, FHSS cordless phone	2427 MHz	2020/02/06 12:57:26
Microwave	2437 MHz	2020/02/06 13:00:09
WiFi, DSSS cordless phone	2437 MHz	2020/02/06 13:00:27
Bluetooth, FHSS cordless phone	2437 MHz	2020/02/06 12:57:45
WiFi, DSSS cordless phone	2452 MHz	2020/02/06 13:00:28
Bluetooth, FHSS cordless phone	2452 MHz	2020/02/06 12:58:52

Analysis for 5 GHz:



6. Click *Close*.

To change the radio mode in the CLI:

```
config wireless-controller wtp-profile
    edit "421E"
        config platform
            set type 421E
        end
        config radio-1
            set mode monitor
        end
        config radio-2
            set mode monitor
        end
    next
end
```

To view spectrum analysis for radio 1 in the CLI:

1. Start the spectrum analysis on channel 1:

```
# execute wireless-controller spectral-scan FP421ETF19000000 1 on 30 1 1000
```

2. View the analysis results:

```
# diagnose wireless-controller wlac -c rf-sa FP421ETF19000000 1 1
------------------------------RF Spectrum Data    1--------------------------
rId: 1 Age: 24   gen 27     rssi: 11  nf: -96 bw: 1 Freq: 2412 Chan: 1   Cnt bin 256
Interf: 0   (idx,duty_max,duty,pwr_max,pwr)
  0  45  14   -67  -89        1  45  14   -60  -89        2  44  14   -63  -89        3  44
 13   -57  -83  -
  4  44  13   -61  -89        5  43  12   -67  -89        6  43  11   -67  -89        7  42
 11   -67  -89
  8  42  10   -67  -89        9  42  10   -67  -89       10  41  10   -67  -83  -    11  41
 10   -67  -89
 12  41  10   -67  -89       13  42  10   -67  -89       14  41  10   -67  -83  -    15  41
 10   -67  -89
 16  41  10   -61  -89       17  41  10   -67  -89       18  41  10   -67  -89       19  41
 9    -67  -89
 20  41  10   -67  -89       21  41  10   -67  -89       22  41  10   -67  -89       23  42
 10   -67  -79  -

# get wireless-controller spectral-info FP421ETF19000000 1
================================================================================
Spectrum info for band freq [2402, 2482] chan [1,13]: (idx,age,gen,duty_max,duty,pwr_
max,pwr)
2402      0      1      7   19   19    -21   -83    -              1      1      7   18
18    -33   -83    -
          2      1      7   18   18    -35   -83    -              3      1      7   17
17    -39   -83    -
          4      1      7   17   17    -43   -83    -              5      1      7   16
16    -47   -83    -
          6      1      7   15   15    -33   -83    -              7      1      7   15
15    -45   -83    -
          8      1      7   14   14    -59   -83    -              9      1      7   14
14    -53   -83    -
         10      1      7   14   14    -59   -83    -             11      1      7   14
14    -59   -83    -
```

3. Stop the spectrum analysis on radio 1:

```
# execute wireless-controller spectral-scan FP421ETF19000000 1 off
```

4. Verify the analysis has stopped:

```
# get wireless-controller spectral-info FP421ETF19000000 1
================================================================================
No spectrum info is found for band freq [2402, 2482] chan [1,13]
================================================================================
No spectrum info is found for band freq [5170, 5330] chan [36,64]
================================================================================
No spectrum info is found for band freq [5490, 5710] chan [100,140]
================================================================================
No spectrum info is found for band freq [5735, 5835] chan [149,165]
FortiGate-80E-POE # diagnose wireless-controller wlac -c rf-sa FP421ETF19000000 1 1
------------------------------Total    0 RF Spectrum Datas--------------------------
```

To view spectrum analysis for radio 2 in the CLI:

1. Start the spectrum analysis on all channels:

```
# execute wireless-controller spectral-scan FP421ETF19000000 2 on
```

2. View the analysis results:

```
# get wireless-controller spectral-info FP421ETF19000000 2
 ================================================================================
 No spectrum info is found for band freq [2402, 2482] chan [1,13]
 ================================================================================
 Spectrum info for band freq [5170, 5330] chan [36,64]: (idx,age,gen,duty_max,duty,pwr_
max,pwr)
 5170      0     24      9   0    0     -92   -94                    1     24      9   0
0     -92   -94
           2     24      9   0    0     -92   -94                    3     24      9   0
0     -92   -94
           4     24      9   0    0     -92   -94                    5     24      9   0
0     -92   -94
           6     24      9   0    0     -92   -94                    7     24      9   0
0     -92   -94
           8     24      9   0    0     -92   -94                    9     24      9   0
0     -92   -94
          10     24      9   0    0     -92   -94                   11     24      9   0
0     -92   -94
          12     24      9   0    0     -92   -94                   13     24      9   0
0     -92   -94
         14     24      9   0    0     -92   -94                   15     24      9   0
0     -92   -94
```

3. Check the spectrum analysis results on specific channels:

```
# diagnose wireless-controller wlac -c rf-sa FP421ETF19000000 2 36
------------------------------RF Spectrum Data    1----------------------------
rId: 2 Age: 6    gen 7      rssi: 2   nf: -96 bw: 1 Freq: 5180 Chan: 36  Cnt bin 256
Interf: 0   (idx,duty_max,duty,pwr_max,pwr)
  0  0   0    -92  -94        1  0   0    -92  -94        2  0   0    -92  -94        3  0
 0    -92  -94
  4  0   0    -92  -94        5  0   0    -92  -94        6  0   0    -92  -94        7  0
 0    -92  -94
  8  0   0    -92  -94        9  0   0    -92  -94       10  0   0    -92  -94       11  0
 0    -92  -94
 12  0   0    -92  -94       13  0   0    -92  -94       14  0   0    -92  -94       15  0
 0    -92  -94
 16  0   0    -92  -94       17  0   0    -92  -94       18  0   0    -92  -94       19  0
 0    -92  -94
 20  0   0    -92  -94       21  0   0    -92  -94       22  0   0    -92  -94       23  0
 0    -92  -94
 24  0   0    -92  -94       25  0   0    -92  -94       26  0   0    -92  -94       27  0
 0    -92  -94
 28  0   0    -92  -94       29  0   0    -92  -94       30  0   0    -92  -94       31  0
 0    -92  -94

# diagnose wireless-controller wlac -c rf-sa FP421ETF19000000 2 165
------------------------------RF Spectrum Data    1----------------------------
rId: 2 Age: 22   gen 6      rssi: 11  nf: -96 bw: 1 Freq: 5825 Chan: 165 Cnt bin 256
Interf: 0   (idx,duty_max,duty,pwr_max,pwr)
```

```
  0   0    0     -90   -90         1   0    0     -90   -90         2   0    0     -90   -90         3   0
0     -90   -90
  4   0    0     -90   -90         5   0    0     -90   -90         6   0    0     -90   -90         7   0
0     -90   -90
  8   0    0     -90   -90         9   0    0     -90   -90        10   0    0     -90   -90        11   0
0     -90   -90
 12   0    0     -90   -90        13   0    0     -90   -90        14   0    0     -90   -90        15   0
0     -90   -90
 16   0    0     -90   -90        17   0    0     -90   -90        18   0    0     -90   -90        19   0
0     -90   -90
 20   0    0     -90   -90        21   0    0     -90   -90        22   0    0     -90   -90        23   0
0     -90   -90
 24   0    0     -90   -90        25   0    0     -90   -90        26   0    0     -90   -90        27   0
0     -90   -90
 28   0    0     -90   -90        29   0    0     -90   -90        30   0    0     -90   -90        31   0
0     -90   -90
```

4. Stop the spectrum analysis on radio 2:

```
# execute wireless-controller spectral-scan FP421ETF19000000 2 off
```

5. Verify the analysis has stopped:

```
# get wireless-controller spectral-info FP421ETF19000000 2
================================================================================
No spectrum info is found for band freq [2402, 2482] chan [1,13]
================================================================================
No spectrum info is found for band freq [5170, 5330] chan [36,64]
================================================================================
No spectrum info is found for band freq [5490, 5710] chan [100,140]
================================================================================
No spectrum info is found for band freq [5735, 5835] chan [149,165]
```

Increase in maximum number of managed FortiAPs

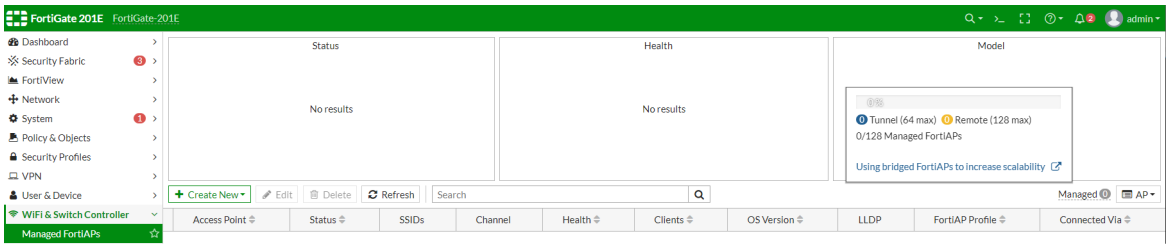
The maximum number of managed FortiAPs has increased in some FortiGate E models for added wireless capability and scalability.

The following comparison table shows the maximum number of FortiAPs supported in FortiGate E models:

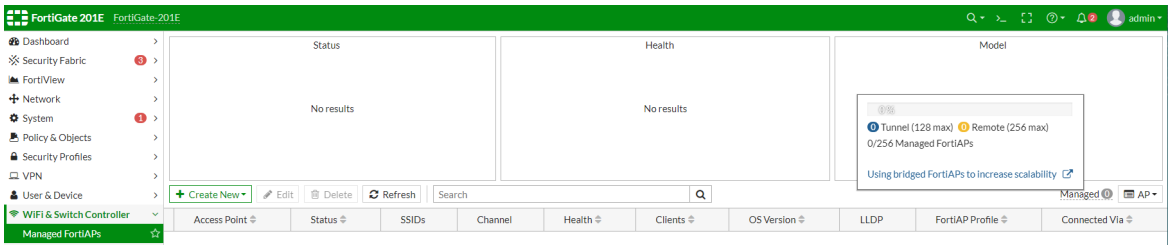
FGT Model	FortiOS 6.2	FortiOS 6.4
FGT200E, FGT201E	128	256
FGT3960E, FGT3980E	4.096	8,192

To view the maximum in the GUI:

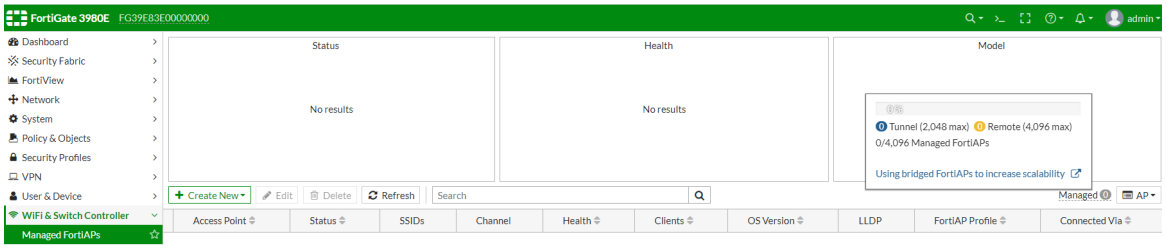
1. Go to *Wifi &Switch controller* > *Managed FortiAPs*.
2. At the right-side of the page, hover over *Managed*. The new maximum appears in the information window. FGT201E can support a maximum of *128 Managed FortiAPs* with FortiOS 6.2.



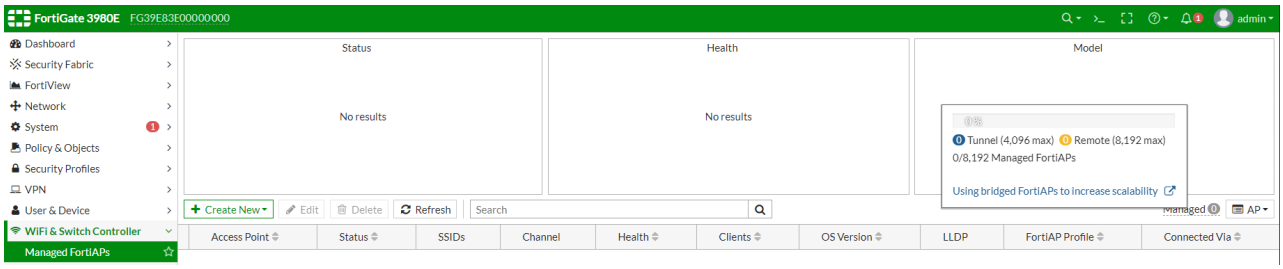
FGT201E can support a maximum of 256 FortiAPs with FortiOS 6.4.



FGT3980E can support a maximum of 4,096 FortiAPs with FortiOS 6.2.



FGT3980E can support a maximum of 8,192 FortiAPs with FortiOS 6.4.



VLAN interface templates for FortiSwitch devices

You can create configuration templates that define the VLAN interfaces and are applied to new FortiSwitch devices when they are discovered and managed by the FortiGate.

For each VDOM, you can create templates, and then assign those templates to the automatically created switch VLAN interfaces for six types of traffic. The network subnet that is reserved for the switch controller can also be customized.

To ensure that switch VLAN interface names are unique for each system, the following naming rules are used:

- root VDOM: the interface names are the same as the template names.
- other VDOMs: the interface name is created from the template name and the SNMP index of the interface. For example, if the template name is *quarantined* and the SNMP index is *29*, then the interface name is *quarantined.29*.

You can also customize the FortiLink management VLAN per FortiLink interface:

```
config system interface
    edit <fortilink interface>
        set fortilink enable
        set switch-controller-mgmt-vlan <integer>
    next
end
```

The management VLAN can be a number from 1 to 4094. the default value is 4094.

Create VLAN interface templates

To configure the VLAN interface templates:

```
config switch-controller initial-config template
    edit <template_name>
        set vlanid <integer>
        set ip <ip/netmask>
        set allowaccess {options}
        set auto-ip {enable | disable}
        set dhcp-server {enable | disable}
    next
end
```

<template_name>	The name, or part of the name, of the template.
vlanid <integer>	The unique VLAN ID for the type of traffic the template is assigned to (1 - 4094, default = 4094)
ip <ip/netmask>	The IP address and subnet mask of the switch VLAN interface. This can only be configured when `auto-ip` is disabled.
allowaccess {options}	The permitted types of management access to this interface.
auto-ip {enable | disable}	When enabled, the switch-controller will pick an unused 24 bit subnet from the `switch-controller-reserved-network` (configured in `config system global`).
dhcp-server {enable | disable}	When enabled, the switch-controller will create a DHCP server for the switch VLAN interface

To assign the templates to the specific traffic types:

```
config switch-controller initial-config vlans
    set default-vlan <template>
    set quarantine <template>
    set rspan <template>
    set voice <template>
    set video <template>
    set nac <template>
end
```

default-vlan <template>	Default VLAN assigned to all switch ports upon discovery.
quarantine <template>	VLAN for quarantined traffic.

rspan <template>	VLAN for RSPAN/ERSPAN mirrored traffic.
voice <template>	VLAN dedicated for voice devices.
video <template>	VLAN dedicated for video devices.
nac <template>	VLAN for NAC onboarding devices.

To configure the network subnet that is reserved for the switch controller:

```
config system global
    set switch-controller-reserved-network <ip/netmask>
end
```

The default value is `169.254.0.0 255.255.0.0`.

Example

In this example, six templates are configured with different VLAN IDs. Except for the default template, all of them have DHCP server enabled. When a FortiSwitch is discovered, VLANs and the corresponding DHCP servers are automatically created.

To configure six templates and apply them to VLAN traffic types:

```
config switch-controller initial-config template
    edit "default"
        set vlanid 1
        set auto-ip disable
    next
    edit "quarantine"
        set vlanid 4093
        set dhcp-server enable
    next
    edit "rspan"
        set vlanid 4092
        set dhcp-server enable
    next
    edit "voice"
        set vlanid 4091
        set dhcp-server enable
    next
    edit "video"
        set vlanid 4090
        set dhcp-server enable
    next
    edit "onboarding"
        set vlanid 4089
        set dhcp-server enable
    next
end
```

```
config switch-controller initial-config vlans
    set default-vlan "default"
    set quarantine "quarantine"
    set rspan "rspan"
```

```
    set voice "voice"
    set video "video"
    set nac "onboarding"
end
```

To see the automatically created VLANs and DHCP servers:

```
show system interface
    edit "default"
        set vdom "root"
        set snmp-index 24
        set switch-controller-feature default-vlan
        set interface "fortilink"
        set vlanid 1
    next
    edit "quarantine"
        set vdom "root"
        set ip 169.254.11.1 255.255.255.0
        set description "Quarantine VLAN"
        set security-mode captive-portal
        set replacemsg-override-group "auth-intf-quarantine"
        set device-identification enable
        set snmp-index 25
        set switch-controller-access-vlan enable
        set switch-controller-feature quarantine
        set color 6
        set interface "fortilink"
        set vlanid 4093
    next
    ...
end
```

```
show system dhcp server
    edit 2
        set dns-service local
        set ntp-service local
        set default-gateway 169.254.1.1
        set netmask 255.255.255.0
        set interface "fortilink"
        config ip-range
            edit 1
                set start-ip 169.254.1.2
                set end-ip 169.254.1.254
            next
        end
        set vci-match enable
        set vci-string "FortiSwitch" "FortiExtender"
    next
    edit 3
        set dns-service default
        set default-gateway 169.254.11.1
        set netmask 255.255.255.0
        set interface "quarantine"
        config ip-range
            edit 1
                set start-ip 169.254.11.2
                set end-ip 169.254.11.254
```

```
            next
        end
        set timezone-option default
    next
    ...
end
```

Improved FortiSwitch support

The number of managed FortiSwitch devices has increased in some FortiGate E models.

FortiGate model	Number of managed FortiSwitches
200E, 201E	64 (from 32)
300E, 301E, 400E, 401E	72 (from 48)
500E, 501E	72 (from 48)
600E, 601E	96 (from 64)
2000E, 2500E	196 (from 128)

Even distribution of FortiAP reports

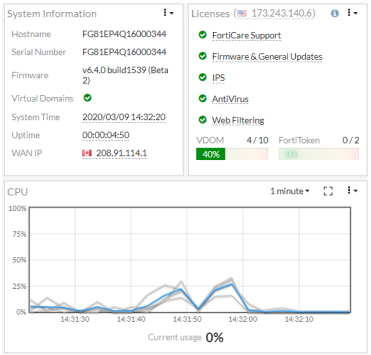
Reporting intervals for FortiAP are now evenly distributed to prevent spikes in CPU usage in FortiGates that manage a large number of AP devices.

FortiAP sends periodic reports to FortiGate when WIDS profiles, DARRP, or auto-power-level are enabled in WTP profiles. Before this improvement was implemented, these periodic reports would frequently reach the wireless controller at the same time, causing spikes in CPU usage.

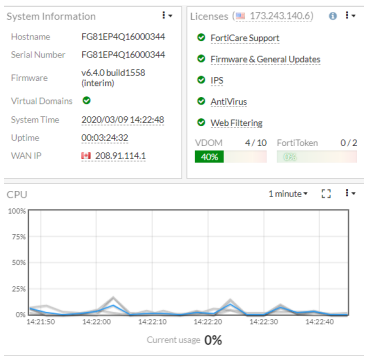
GUI

The following images compare the CPU usage in a FortiGate that manages 16 FortiAPs before and after the improvement was implemented.

Before the improvement, CPU usage is above 25%. The spike in usage can go as high as 90% if the FortiGate manages more than 16 devices.



After the improvement is implemented, CPU usage is approximately 10% in the same FortiGate.



CLI

The following examples show the improvements in the CLI for the same FortiGate device.

In this example, you can see 16 wireless sessions in the CLI.

```
FG81EP4Q16000344 (root) # diag wire wlac -c ws | grep "WTP session"
WTP session  : 0-10.43.1.1:62332  CWAS_RUN
WTP session  : 0-10.43.1.1:62350  CWAS_RUN
WTP session  : 0-10.43.1.1:62356  CWAS_RUN
WTP session  : 0-10.43.1.1:62357  CWAS_RUN
WTP session  : 0-10.43.1.1:62325  CWAS_RUN
WTP session  : 0-10.43.1.1:15246  CWAS_RUN
WTP session  : 0-10.43.1.1:62362  CWAS_RUN
WTP session  : 0-10.43.1.1:62364  CWAS_RUN
WTP session  : 0-10.43.1.1:62366  CWAS_RUN
WTP session  : 0-10.43.1.1:62367  CWAS_RUN
WTP session  : 0-10.43.1.1:62319  CWAS_RUN
WTP session  : 0-10.43.1.1:62321  CWAS_RUN
WTP session  : 0-10.43.1.1:62320  CWAS_RUN
WTP session  : 0-10.43.1.1:62370  CWAS_RUN
WTP session  : 0-10.43.1.1:62323  CWAS_RUN
WTP session  : 0-10.43.1.1:62329  CWAS_RUN
```

Before the improvement is implemented, the FortiAP WTP reports are not indexed, which can cause spikes in CPU usage.

```
FG81EP4Q16000344 (root) # diag wireless-controller  wlac -c ws | grep report
FG81EP4Q16000344 (root) #
```

After the improvement is implemented, the AC assigns a wtp-report-index to each managed FortiAP, preventing spikes in CPU usage.

```
FG81EP4Q16000344 (root) # diag wireless-controller  wlac -c ws | grep report
wtp-report-index  : 1
wtp-report-index  : 2
wtp-report-index  : 3
wtp-report-index  : 4
wtp-report-index  : 5
wtp-report-index  : 6
wtp-report-index  : 7
wtp-report-index  : 8
wtp-report-index  : 9
wtp-report-index  : 10
wtp-report-index  : 11
```

```
wtp-report-index  : 12
wtp-report-index  : 13
wtp-report-index  : 14
wtp-report-index  : 15
wtp-report-index  : 16
```

You can see the value for the wtp-report-index when you filter the data by device. In this example, the report index is 16.

```
FG81EP4Q16000344 (root) # diag wireless-controller  wlac -c ws 10.231.40.15
------------------------------WTP SESSION  1----------------------------
WTP session  : 0-10.43.1.1:62433  CWAS_RUN
Ctrl in_ifIdx  : 5/wan1
indev  : 5/wan1
Data in_ifIdx  : 5/wan1
indev  : 0/
mesh uplink  : ethernet
id  : FP423E3X16000304
mgmt_vlanid  : 0
wtp_wanlan_mode  : wan-only
refcnt  : 10
deleted  : no
plain_ctl  : disabled
wtp-mode  : normal
wtp-report-index  : 16
data-chan-sec  : clear-text
ctl-msg-offload  : ac=01ff/wtp_loc=01ff/wtp_rem=01ff/oper=01ff
session_id  : 70386ec03c8bdcd630efda365b3f9ce0
ehapd cfg  : done
message queue  : 0/128 max 65
tId_10_sec  : 3537
Ekahau  : disabled
Aeroscout  : disabled
FortiPresence  : disabled
Radio 1  : AP
wlan cfg  : 81ep_ssid1  81ep_ssid2  81ep_ssid4  81ep_wpa3_sae
vap-01(1)  : 81ep_ssid1  90:6c:ac:dc:60:b0  lsw  FOS-QA-Bruce_81ep1 Config success State RUN
vap-02(2)  : 81ep_ssid2  90:6c:ac:dc:60:b1  lsw  FOS-QA-Bruce_81ep2 Config success State RUN
vap-03(3)  : 81ep_ssid4  90:6c:ac:dc:60:b2  lsw  FOS-QA-BRUCE_roaming Config success State
     RUN
vap-04(4)  : 81ep_wpa3_sae  90:6c:ac:dc:60:b3  lsw  81ep_wpa3_sae  Config success State INIT
Radio 2  : AP
wlan cfg  : 81ep_ssid1  81ep_ssid2  81ep_ssid4  81ep_wpa3_sae
vap-01(1)  : 81ep_ssid1  90:6c:ac:dc:60:b8  lsw  FOS-QA-Bruce_81ep1 Config success State RUN
vap-02(2)  : 81ep_ssid2  90:6c:ac:dc:60:b9  lsw  FOS-QA-Bruce_81ep2 Config success State RUN
vap-03(3)  : 81ep_ssid4  90:6c:ac:dc:60:ba  lsw  FOS-QA-BRUCE_roaming Config success State
     RUN
vap-04(4)  : 81ep_wpa3_sae  90:6c:ac:dc:60:bb  lsw  81ep_wpa3_sae  Config success State N/A
Radio 3  : Not Exist
Radio 4  : Not Exist
Radio 5  : Not Exist
```

You can also see the device's wtp-report-index value when you view the WTP configuration in FortiAP.

```
FortiAP-423E # cw_diag -c wtp-cfg
WTP Configuration
   name  : FortiAP-423E
   loc  : N/A
   ap mode  : thin AP
   fmvap  : FG81EP4Q16000344,(12ac979c,5e693999,1),1800,0
```

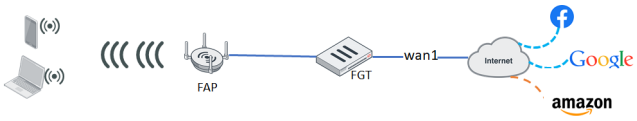
```
atf mode  : disabled
dual-5g mode  : disabled
poe mode  : auto
poe mode oper  : 802.3at
led mode  : normal
led schedules  : SMTWTFS 00:00->00:00,
WAN port cnt  : 2
lan1  : carrier=1, speed=1000, duplex=full
lan2  : carrier=0, speed=0, duplex=
energy-efficient-eth : disable
extension info enable: enable
allowaccess  : https ssh
lldp enable  : enable
wtp-report-index  : 16
ctl-msg-offload  : ac=01ff/wtp=01ff/oper=01ff
radio cnt  : 2
sta info  : 0/0
echo-interval  : 30
keep-alive-interval  : 30
max-retransmit  : 3
dc-dead-interval  : 120
discovery-interval  : 5
report-interval  : 30
sta-stats-interval  : 1
vap-stats-interval  : 15
radio-stats-interval : 15
sta-cap-interval  : 30
idle-timeout  : 300
fpresence-interval  : 3600, 30
statistics-interval  : 120
fsm-state  : RUN 439
wtp-ip-addr  : 10.231.40.15:25246 - 10.231.40.15:36529
ac-ip-addr  : 172.18.56.46:5246 - 172.18.56.46:5247  DHCP
base-mac  : 90:6c:ac:dc:60:a8
bulk data seq num  : -1
ap-mgmt-vlanid  : 0
ac-cert-version  : 1
cert-version-oper  : 1
data-chan-sec-cfg  : clear-text dtls ipsec
data-chan-sec-oper  : clear-text
ip-frag-prevent  : TCP_MSS  (ul_mtu=1500 dl_mtu=1500)
ekahau  : disabled
aeroscout  : disabled
data-ethernet-II  : disabled
fortipresence  : disabled, ble enabled, rogue disabled, unassoc_sta enabled, freq 30
server 0.0.0.0:3000 secret csum [0xc6a7] project [fortipresence]
LAN mode  : disabled
LAN port cnt  : 0
encrypt_key[0-15]  : 14-aa-7f-3e-34-a1-83-e7-ca-51-49-2c-e3-64-b3-03
encrypt_key[16-31]  : 70-1a-42-5b-a5-5d-79-f0-c4-6e-e0-2f-a8-81-58-13
```

View detailed information for individual WiFi connections

Administrators can use the GUI to view detailed information about the health of individual WiFi connections from the *Dashboard* or the *WiFi Clients* console. You can also *Quarantine* or *Disassociate* a wireless client. The information in

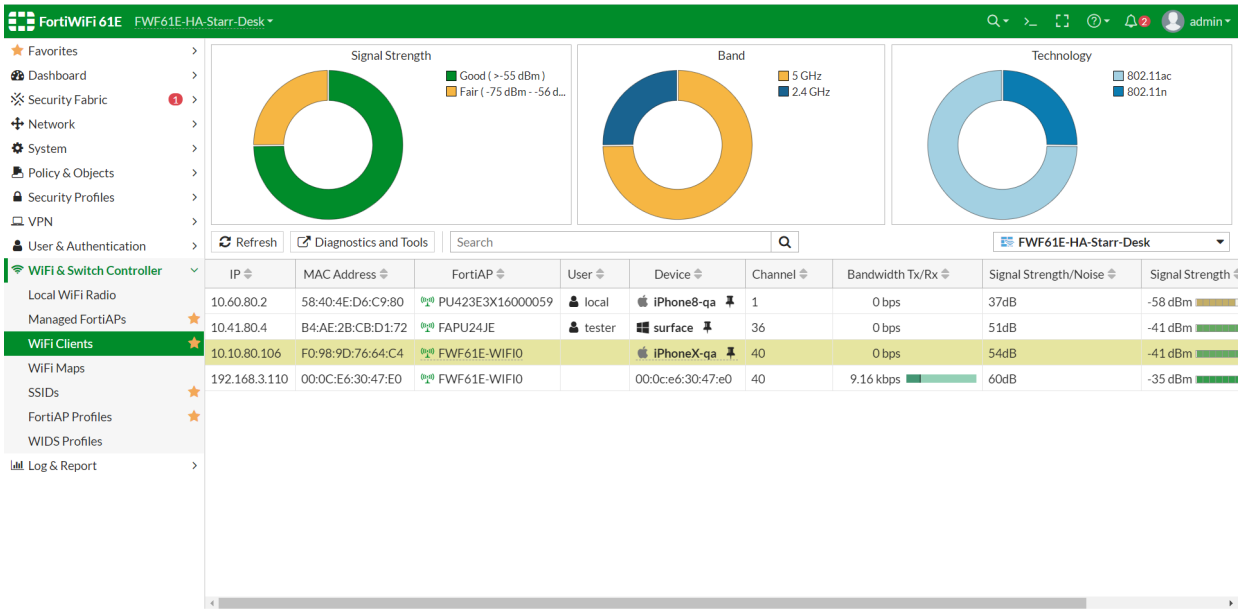
the *FortiView* page is now displayed as tabs in the summary window for each wireless client.

Sample topology

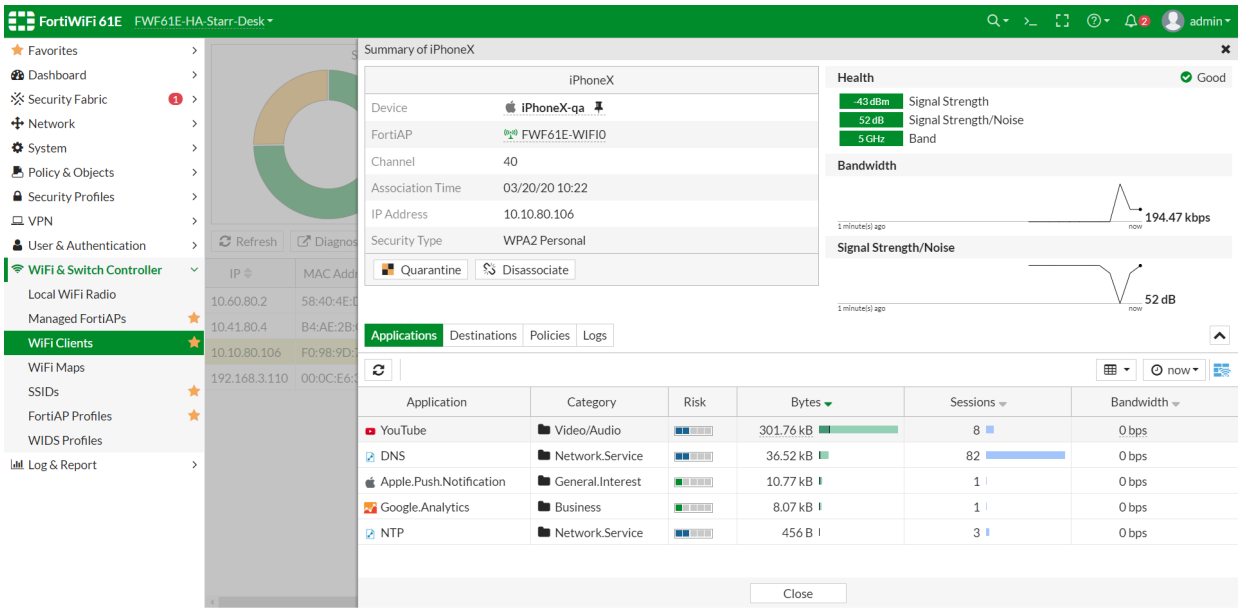


To view the summary page for a wireless client in the GUI:

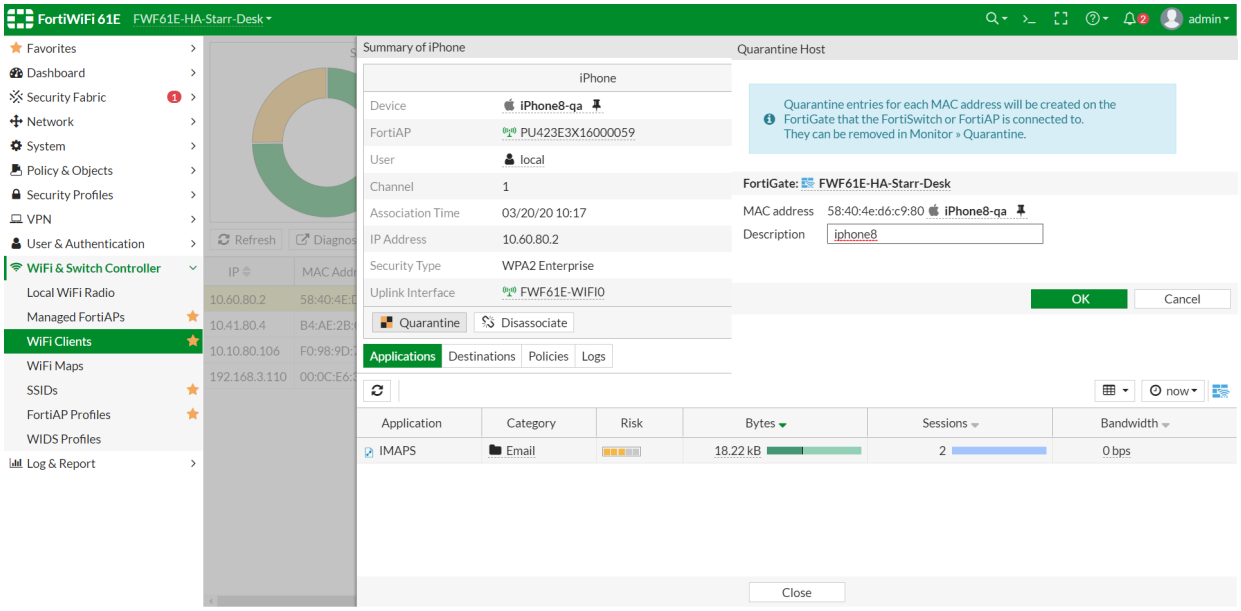
1. Go to *WiFi & Switch Controller* > *WiFi Clients*, and select a wireless client. Click *Diagnostics and Tools* to the right side of the *Refresh* icon.



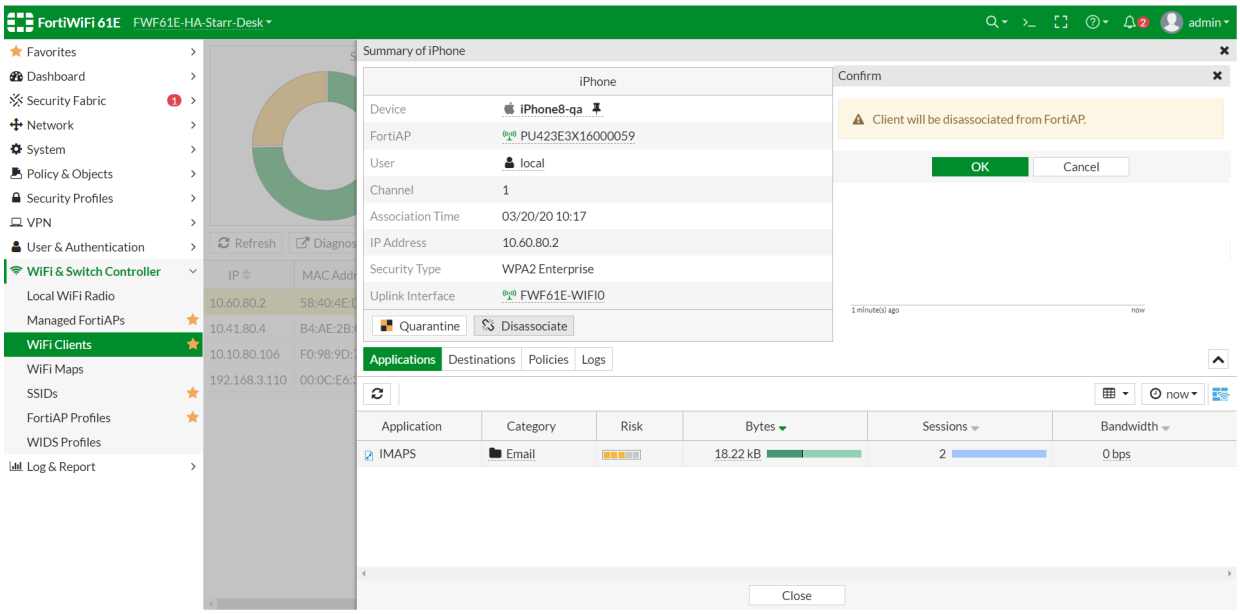
2. The summary page for the selected client opens.



3. On the summary page, click *Quarantine*.The *Quarantine Host* dialog opens. Click *OK* to quarantine the selected wireless client, and close the dialog.



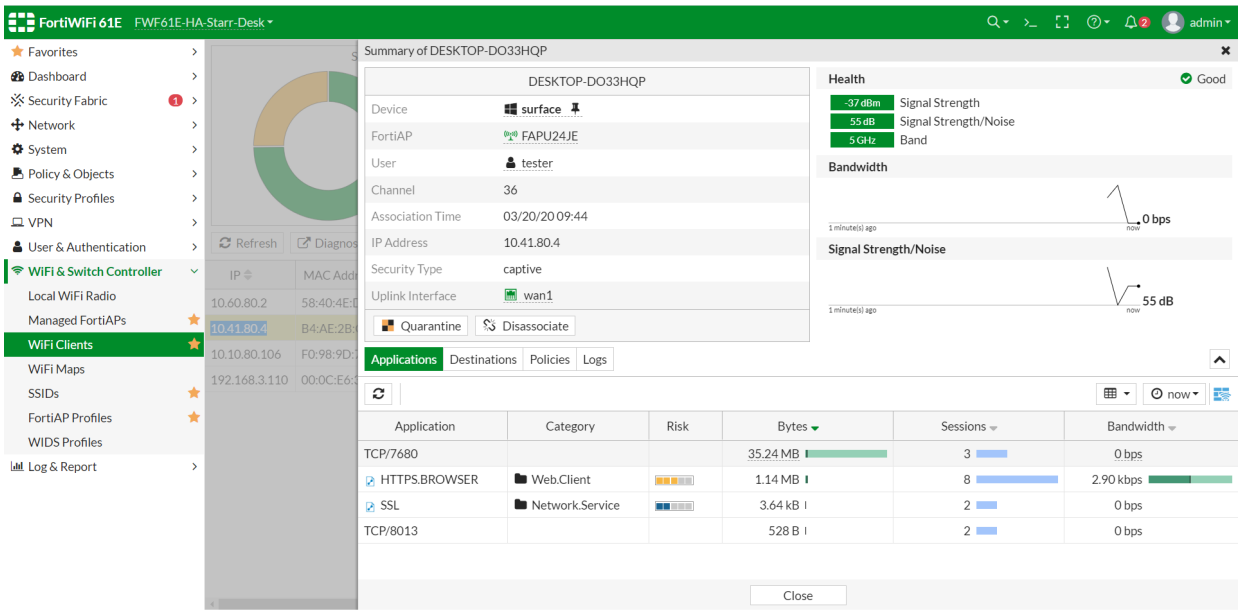
4. On the summary page, click the *Disassociate* icon. The *Confirm* dialog opens. Click *OK* to dissociate the selected wireless client, and close the dialog.



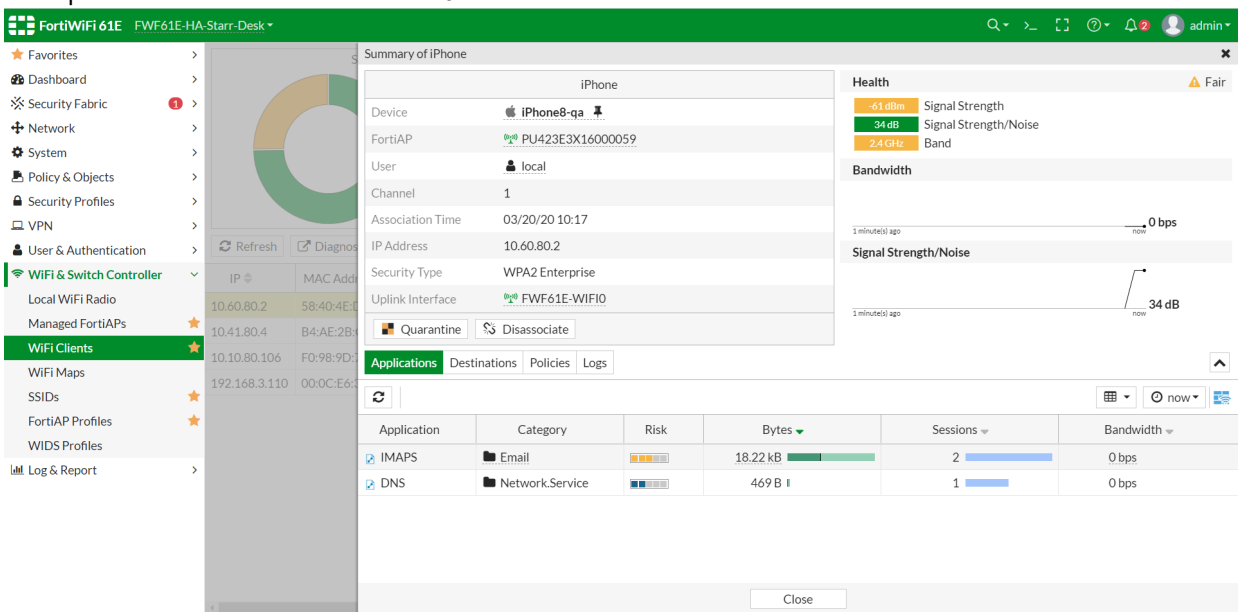
5. From the summary page, the *Health* section displays the overall health for the wireless connection. The overall health of the connection is:
   - *Good* if the value range for all three conditions are Good.
   - *Fair* or *Poor* if one of the three conditions is *Fair* or *Poor*.

Condition	Value range
Signal Strength	• *Good* > -56dBm • -56dBm > *Fair* > -75dBm • *Poor* < -75dBm
Signal Strengthen	• *Good* > 39dBm • 20dBm < *Fair* < 39dBm • *Poor* < 20dBm
Band	• *Good* = 5G band • *Fair* = 2.4G band

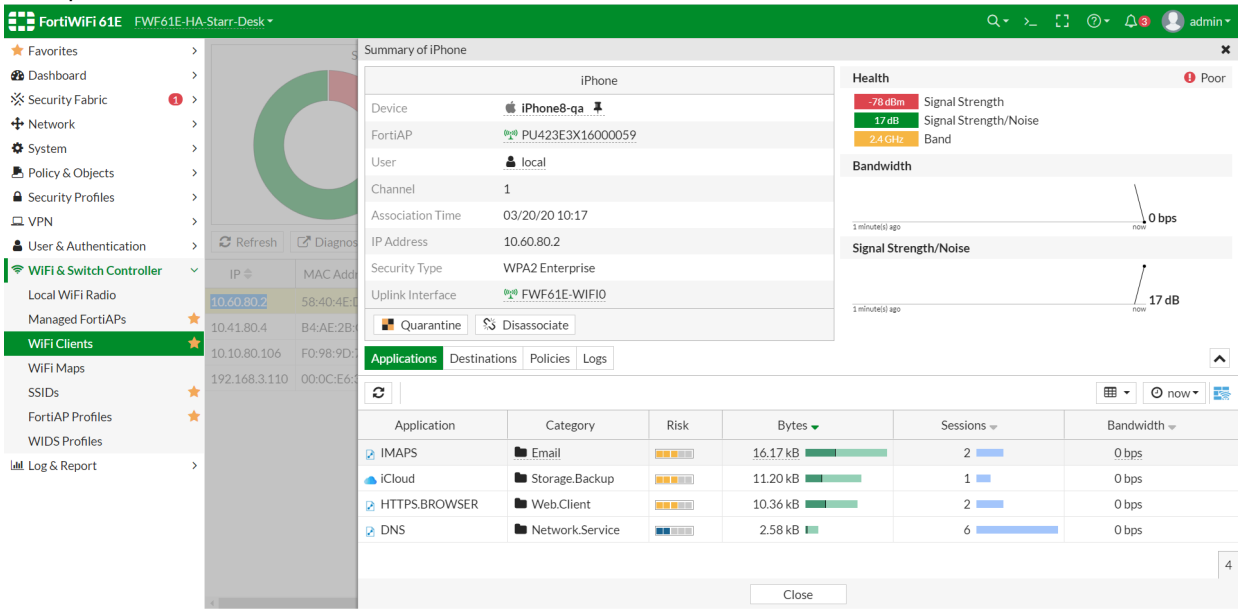
Example of an overall health status of *Good*.



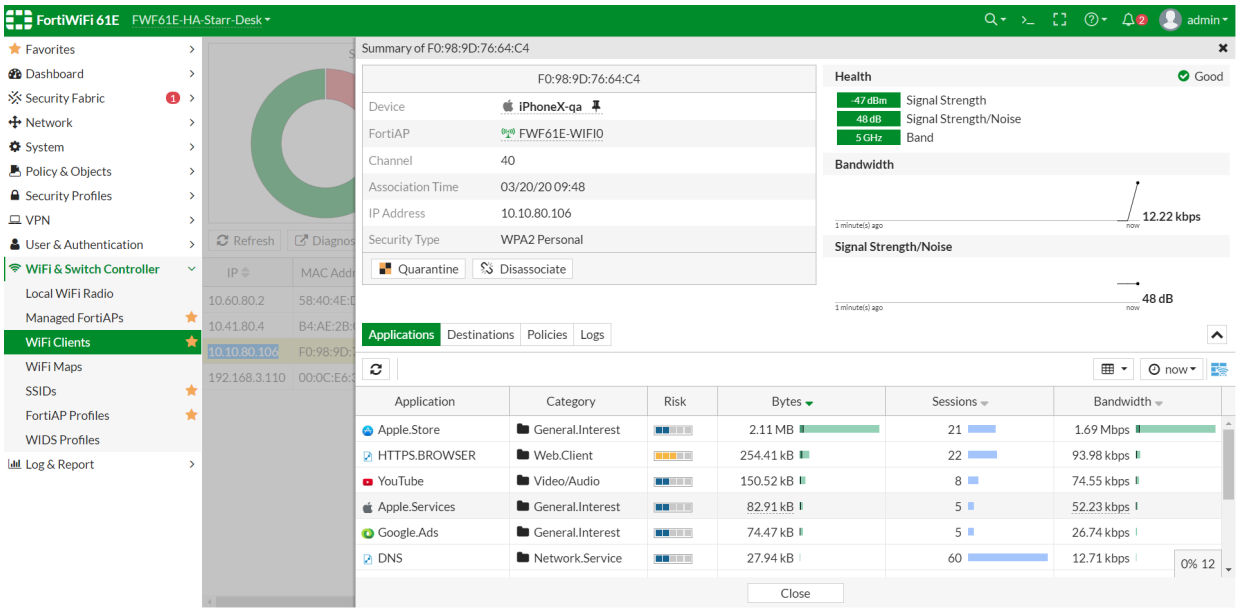
Example of an overall health status of *Fair*.

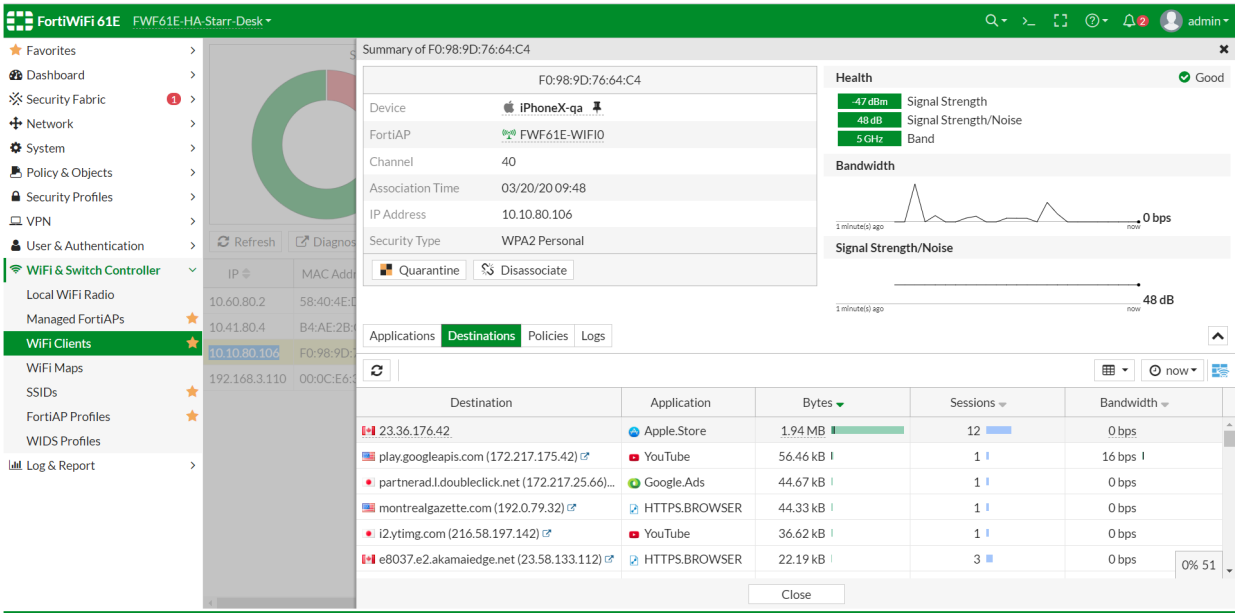


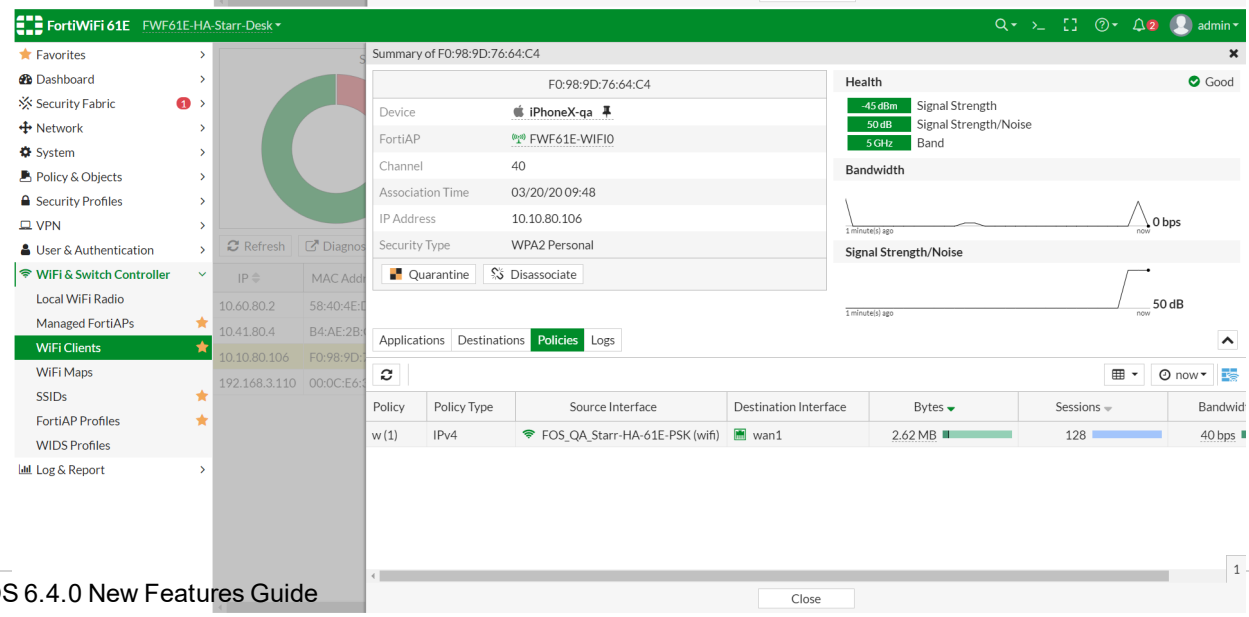
Example of an overall health status of *Poor*.

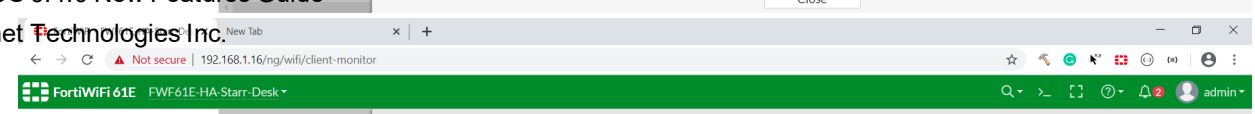


6. The summary page contains four FortiView tabs:
   - *Applications*
   - *Destinations*
   - *Policies*
   - *Logs*

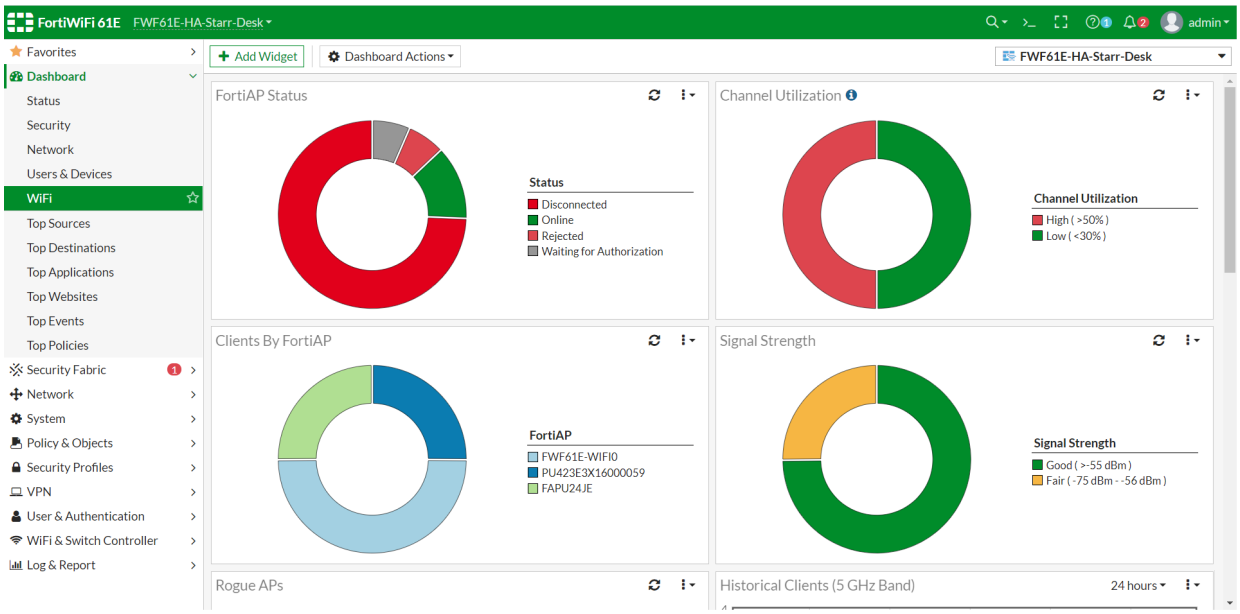

FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Summary of F0:98:9D:76:64:C4
F0:98:9D:76:64:C4
Device
iPhoneX-qa
FortiAP
FWF61E-WIFI0
Channel
40
Association Time
03/20/20 09:48
IP Address
10.10.80.106
Security Type
WPA2 Personal
Quarantine
Disassociate
Health
Good
-47 dBm
Signal Strength
48 dB
Signal Strength/Noise
5 GHz
Band
Bandwidth
12.22 kbps
Signal Strength/Noise
48 dB
Applications
Destinations
Policies
Logs
Application
Category
Risk
Bytes
Sessions
Bandwidth
Apple.Store
General.Interest
2.11 MB
21
1.69 Mbps
HTTPS.BROWSER
Web.Client
254.41 kB
22
93.98 kbps
YouTube
Video/Audio
150.52 kB
8
74.55 kbps
Apple.Services
General.Interest
82.91 kB
5
52.23 kbps
Google.Ads
General.Interest
74.47 kB
5
26.74 kbps
DNS
Network.Service
27.94 kB
60
12.71 kbps
Close

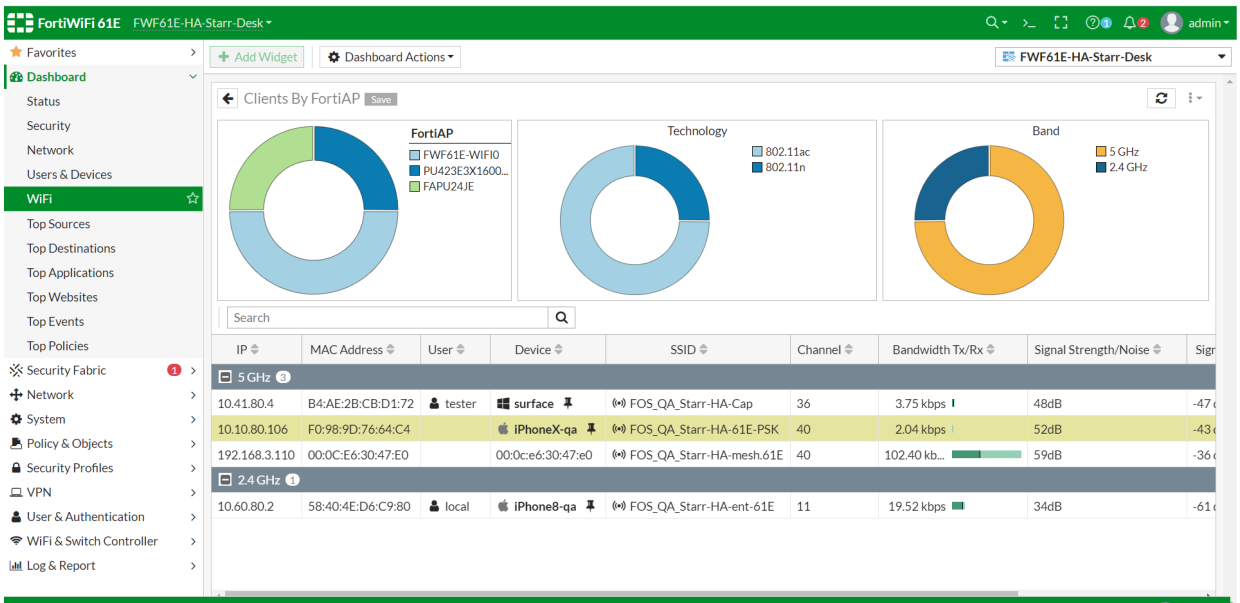

FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Summary of F0:98:9D:76:64:C4
F0:98:9D:76:64:C4
Device
iPhoneX-qa
FortiAP
FWF61E-WIFI0
Channel
40
Association Time
03/20/20 09:48
IP Address
10.10.80.106
Security Type
WPA2 Personal
Quarantine
Disassociate
Health
Good
-47 dBm
Signal Strength
48 dB
Signal Strength/Noise
5 GHz
Band
Bandwidth
0 bps
Signal Strength/Noise
48 dB
Applications
Destinations
Policies
Logs
Destination
Application
Bytes
Sessions
Bandwidth
23.36.176.42
Apple.Store
1.94 MB
12
0 bps
play.googleapis.com (172.217.175.42)
YouTube
56.46 kB
1
16 bps
partnerad.l.doubleclick.net (172.217.25.66)...
Google.Ads
44.67 kB
1
0 bps
montrealgazette.com (192.0.79.32)
HTTPS.BROWSER
44.33 kB
1
0 bps
i2.ytimg.com (216.58.197.142)
YouTube
36.62 kB
1
0 bps
e8037.e2.akamaiedge.net (23.58.133.112)
HTTPS.BROWSER
22.19 kB
3
0 bps
Close

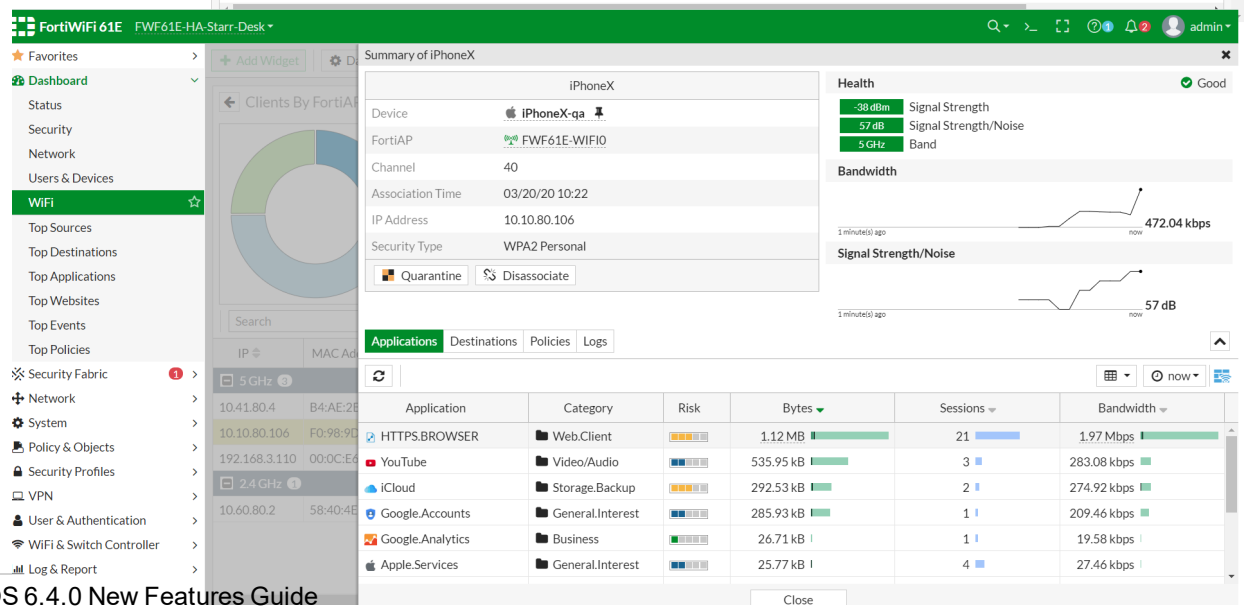

FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Summary of F0:98:9D:76:64:C4
F0:98:9D:76:64:C4
Device
iPhoneX-qa
FortiAP
FWF61E-WIFI0
Channel
40
Association Time
03/20/20 09:48
IP Address
10.10.80.106
Security Type
WPA2 Personal
Quarantine
Disassociate
Health
Good
-45 dBm
Signal Strength
50 dB
Signal Strength/Noise
5 GHz
Band
Bandwidth
0 bps
Signal Strength/Noise
50 dB
Applications
Destinations
Policies
Logs
Policy
Policy Type
Source Interface
Destination Interface
Bytes
Sessions
w (1)
IPv4
FOS_QA_Starr-HA-61E-PSK (wifi)
wan1
2.62 MB
128
40 bps
Close


New Tab
Not secure
192.168.1.16/ng/wifi/client-monitor
FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin

7. Go to *Dashboard* > *WiFi*. Click the *Clients By FortiAP* widget to view the drill-down information for the wireless client.


FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Favorites
Dashboard
Status
Security
Network
Users & Devices
WiFi
Top Sources
Top Destinations
Top Applications
Top Websites
Top Events
Top Policies
Security Fabric
Network
System
Policy & Objects
Security Profiles
VPN
User & Authentication
WiFi & Switch Controller
Log & Report
Add Widget
Dashboard Actions
FWF61E-HA-Starr-Desk
FortiAP Status
Status
Disconnected
Online
Rejected
Waiting for Authorization
Channel Utilization
Channel Utilization
High (>50%)
Low (<30%)
Clients By FortiAP
FortiAP
FWF61E-WIFI0
PU423E3X16000059
FAPU24JE
Signal Strength
Signal Strength
Good (>-55 dBm)
Fair (-75 dBm - -56 dBm)
Rogue APs
Historical Clients (5 GHz Band)
24 hours


FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Add Widget
Dashboard Actions
FWF61E-HA-Starr-Desk
Clients By FortiAP
Save
FortiAP
FWF61E-WIFI0
PU423E3X1600...
FAPU24JE
Technology
802.11ac
802.11n
Band
5 GHz
2.4 GHz
Search
IP
MAC Address
User
Device
SSID
Channel
Bandwidth Tx/Rx
Signal Strength/Noise
5 GHz
10.41.80.4
B4:AE:2B:CB:D1:72
tester
surface
FOS_QA_Starr-HA-Cap
36
3.75 kbps
48dB
10.10.80.106
F0:98:9D:76:64:C4
iPhoneX-qa
FOS_QA_Starr-HA-61E-PSK
40
2.04 kbps
52dB
192.168.3.110
00:0C:E6:30:47:E0
00:0c:e6:30:47:e0
FOS_QA_Starr-HA-mesh.61E
40
102.40 kb...
59dB
2.4 GHz
10.60.80.2
58:40:4E:D6:C9:80
local
iPhone8-qa
FOS_QA_Starr-HA-ent-61E
11
19.52 kbps
34dB


FortiWiFi 61E
FWF61E-HA-Starr-Desk
admin
Summary of iPhoneX
iPhoneX
Device
iPhoneX-qa
FortiAP
FWF61E-WIFI0
Channel
40
Association Time
03/20/20 10:22
IP Address
10.10.80.106
Security Type
WPA2 Personal
Quarantine
Disassociate
Health
Good
-38 dBm
Signal Strength
57 dB
Signal Strength/Noise
5 GHz
Band
Bandwidth
472.04 kbps
Signal Strength/Noise
57 dB
Applications
Destinations
Policies
Logs
now
Application
Category
Risk
Bytes
Sessions
Bandwidth
HTTPS.BROWSER
Web.Client
1.12 MB
21
1.97 Mbps
YouTube
Video/Audio
535.95 kB
3
283.08 kbps
iCloud
Storage.Backup
292.53 kB
2
274.92 kbps
Google.Accounts
General.Interest
285.93 kB
1
209.46 kbps
Google.Analytics
Business
26.71 kB
1
19.58 kbps
Apple.Services
General.Interest
25.77 kB
4
27.46 kbps
Close

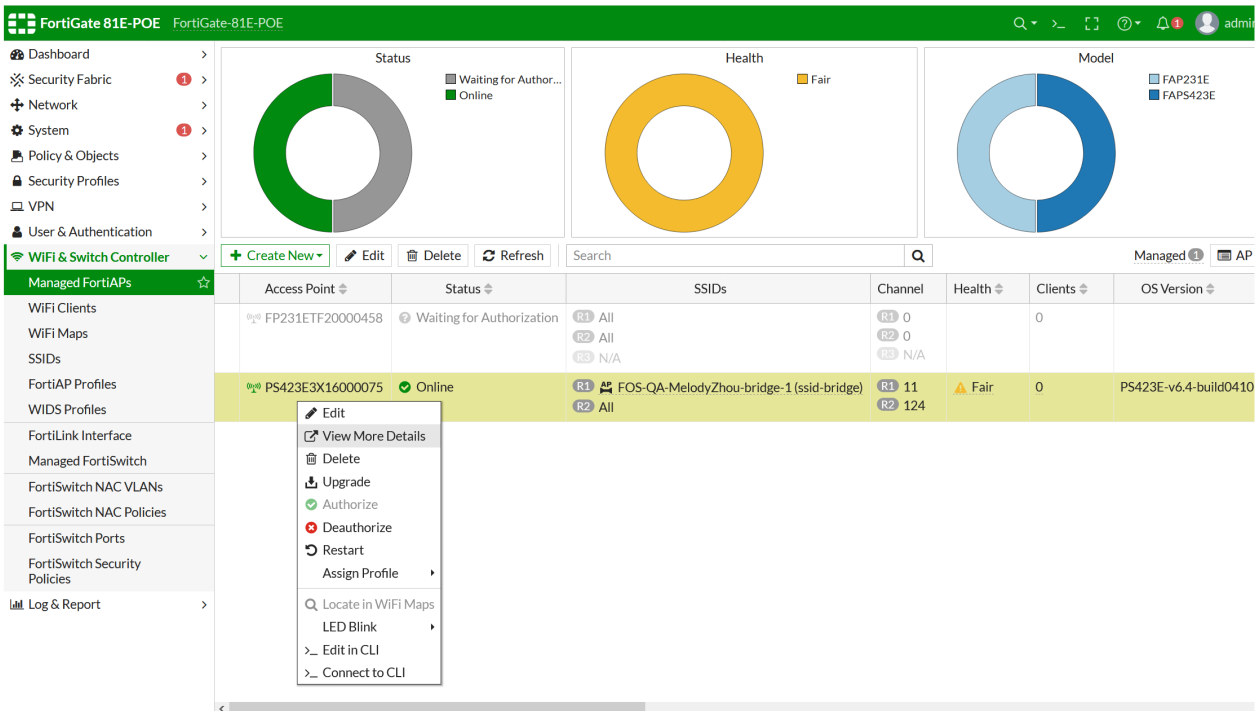
VLAN probe report

FortiGates that manage FortiAPs have the ability to probe VLANs and subnets connected to an access point. Use the VLAN probe wireless tool to help troubleshoot why users cannot connect to the Internet.

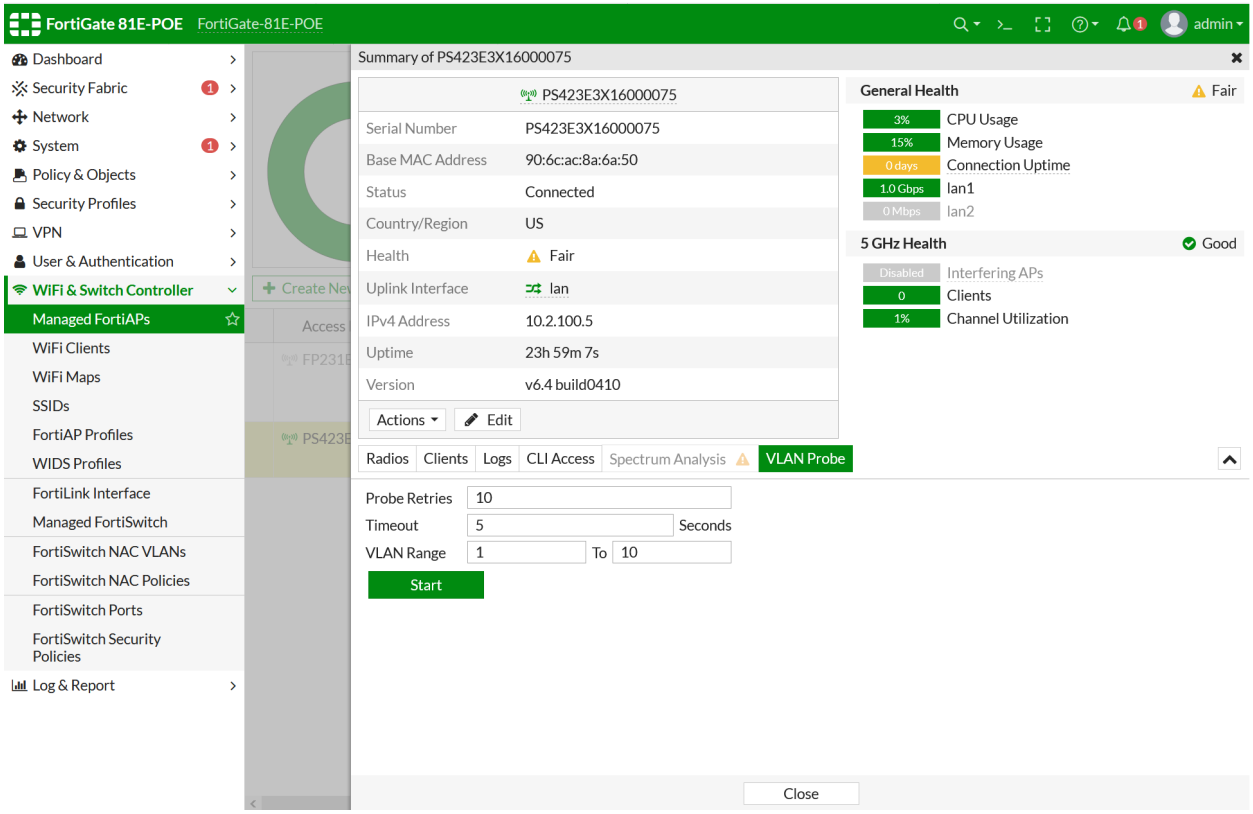
GUI

To perform a VLAN probe in the GUI:

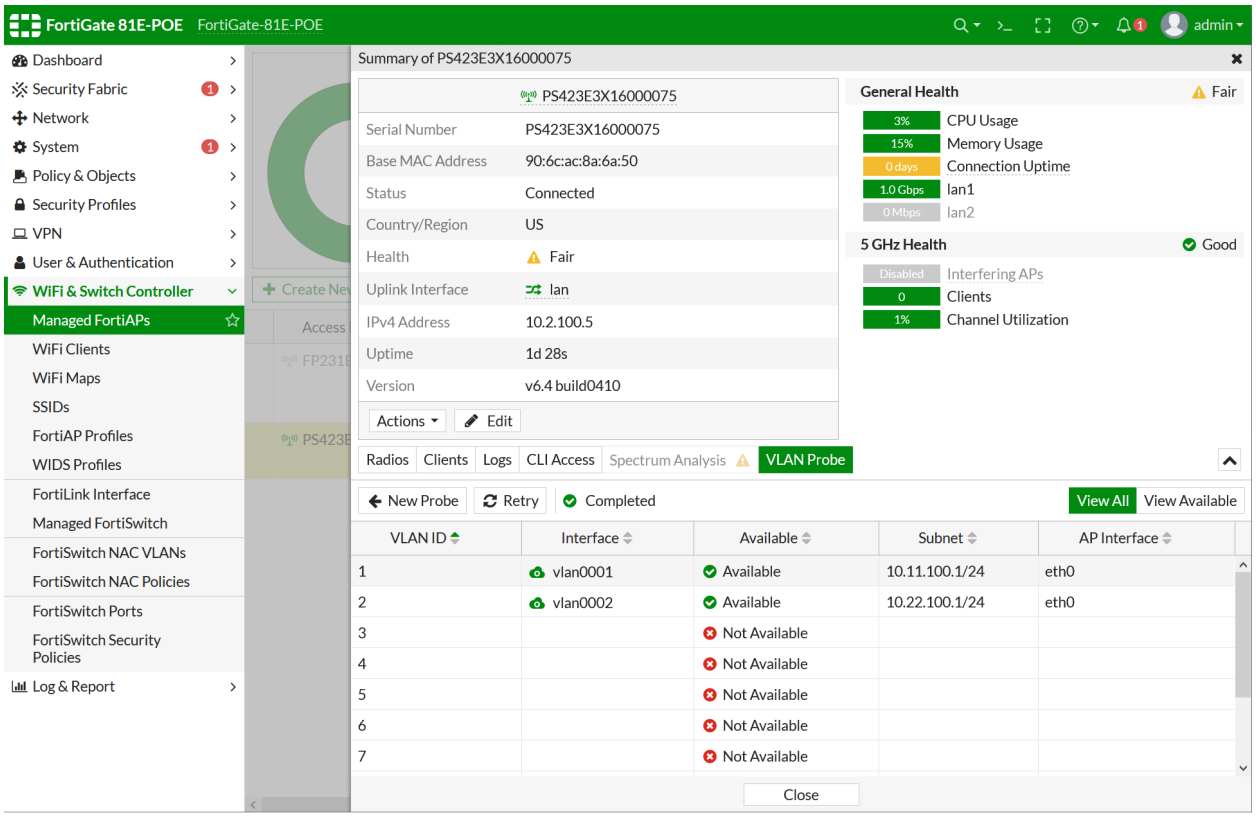
1. Go to *WIFI & Switch Controller* > *Managed FortiAPs*.
2. Right click a FortiAP entry, and select *View More Details*.



3. Click the *VLAN Probe* tab.
   a. Configure the settings in the *VLAN Range* field.
   b. Click *Start*.



4. After the VLAN probing is complete, the VLAN probing report appears in the summary page.



CLI

You can use the CLI console in FortiGate and FortiAP to perform a VLAN probe and view the report.

FortiGate

Command syntax:

```
diagnose wireless-controller wlac -c vlan-probe-cmd  <FAP Serial Number> <action> <interface
     ID> <start Vlan ID>  <end Vlan ID>  <retry> <timeout>
diagnose wireless-controller wlac -c vlan-probe-rpt  <FAP Serial Number> <interface ID>
```

Where the value for `action` is:

- `0` — Start
- `1` — Stop

And where the value for `interface ID` is:

- `0` — All Ethernet port(s) of FortiAP
- `1` — The 1st Ethernet port of FortiAP, eth0
- `2` — The 2nd Ethernet port of FortiAP, eth1, if the hardware exists

To perform a VLAN probe in the CLI:

```
FortiGate-81E-POE (vdom1) # diagnose wireless-controller wlac -c vlan-probe-cmd
     PS423E3X16000075 0 0 1  4094  2 10

Sending VLAN probe command to PS423E3X16000075: action=start wan-port=1 vlan=[1,4094]
     retries=2 timeout=10s
Sending VLAN probe command to PS423E3X16000075: action=start wan-port=2 vlan=[1,4094]
     retries=2 timeout=10s
   Stop VLAN probing. You don't need to run stop command to get the probing report, only use it
       when you want to stop probing.
FortiGate-81E-POE (vdom1) # diag wi wlac -c vlan-probe-cmd PS423E3X16000075 1 0
Sending VLAN probe command to PS423E3X16000075: action=stop wan-port=1
Sending VLAN probe command to PS423E3X16000075: action=stop wan-port=2
```

To view the VLAN probe report in the CLI:

```
FortiGate-81E-POE (vdom1) # diagnose wireless-controller wlac -c vlan-probe-rpt
     PS423E3X16000075 0

VLAN probing status on eth0: Done

intf eth0  VLAN_ID=0001 gateway=10.11.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0002 gateway=10.22.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0003 gateway=10.33.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0004 gateway=10.44.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0005 gateway=10.55.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0006 gateway=10.66.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0007 gateway=10.77.100.1/24     probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0100 gateway=10.10.20.2/24      probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=0200 gateway=10.4.100.1/24      probed_at=Wed Jan 16 17:09:48 2019
intf eth0  VLAN_ID=0300 gateway=10.5.100.1/24      probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=0400 gateway=10.6.100.1/24      probed_at=Wed Jan 16 17:09:49 2019
```

```
intf eth0  VLAN_ID=0500 gateway=10.7.100.1/24      probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=0600 gateway=10.9.100.1/24      probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=1000 gateway=10.10.100.1/24     probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=2000 gateway=10.20.100.1/24     probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=3000 gateway=10.30.100.1/24     probed_at=Wed Jan 16 17:09:49 2019
intf eth0  VLAN_ID=4000 gateway=10.40.100.1/24     probed_at=Wed Jan 16 17:09:49 2019

VLAN probing status on eth1: Done
```

FortiAP

Command syntax

```
cw_diag -c vlan-probe-cmd <action> <interface ID> <start Vlan ID> <end Vlan
      ID> <retry> <timeout>
cw_diag -c vlan-probe-rpt
```

Where the value for `action` is:

- `0`— Start
- `1` — Stop

To perform a VLAN probe in the CLI:

```
PS423E3X16000075 # cw_diag -c vlan-probe-cmd 0 eth0 1 4094 2 10
VLAN probing: start intf [eth0] vlan range[1,4094] retries[2] timeout[10s] ...

   Stop VLAN probing.You don't need to run stop command to get the probing report, only use it
        when you want to stop probing.

PS423E3X16000075 # cw_diag -c vlan-probe-cmd 1 eth0
VLAN probing: stop intf [eth0] vlan range[0,0] retries[0] timeout[0s] ...
```

To view the VLAN probe report in the CLI:

```
PS423E3X16000075 # cw_diag -c vlan-probe-rpt

WTP VLAN probing status: Idle

VLAN probing report on intf[eth0] vlan range[1,4094] retries[2] timeout[10]:

   VLAN_ID=0001 gateway=10.11.100.1/24  age=289
   VLAN_ID=0002 gateway=10.22.100.1/24  age=289
   VLAN_ID=0003 gateway=10.33.100.1/24  age=289
   VLAN_ID=0004 gateway=10.44.100.1/24  age=289
   VLAN_ID=0005 gateway=10.55.100.1/24  age=289
   VLAN_ID=0006 gateway=10.66.100.1/24  age=289
   VLAN_ID=0007 gateway=10.77.100.1/24  age=289
   VLAN_ID=0100 gateway=10.3.100.1/24  age=289
   VLAN_ID=0200 gateway=10.4.100.1/24  age=289
   VLAN_ID=0300 gateway=10.5.100.1/24  age=289
   VLAN_ID=0400 gateway=10.6.100.1/24  age=289
   VLAN_ID=0500 gateway=10.7.100.1/24  age=289
   VLAN_ID=0600 gateway=10.9.100.1/24  age=289
   VLAN_ID=1000 gateway=10.10.100.1/24  age=289
   VLAN_ID=2000 gateway=10.20.100.1/24  age=289
```

```
   VLAN_ID=3000 gateway=10.30.100.1/24  age=289
   VLAN_ID=4000 gateway=10.40.100.1/24  age=289

VLAN probing report on intf[eth1] vlan range[1,4094] retries[2] timeout[10]:
```

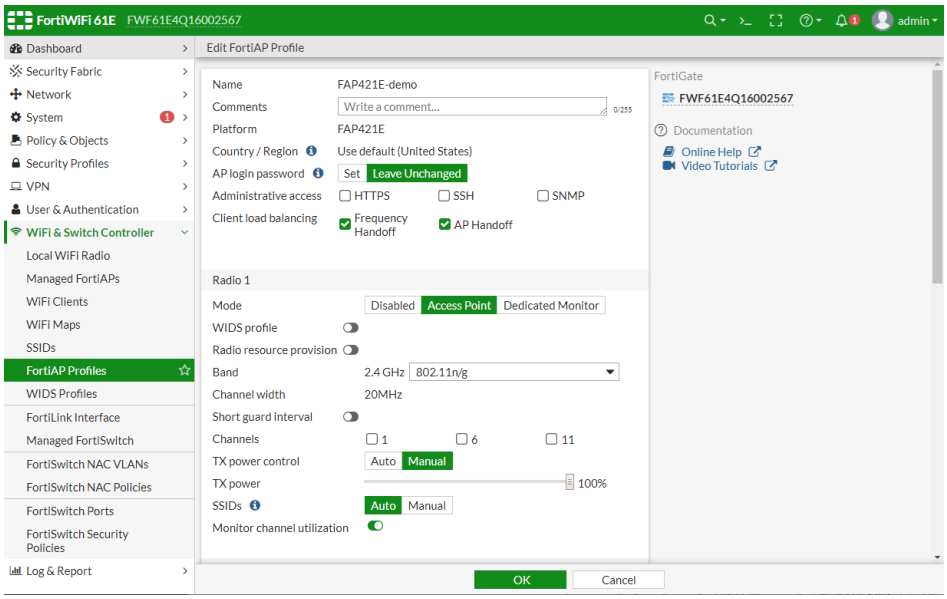
FortiAP client load balancing per AP

The frequency and AP handoff options are moved from the radio level to the global section of FortiAP profiles. If either load balancing options are enabled on any radio prior to upgrading, the setting will be enabled after upgrading.

In this example, a new custom profile is created with both client load balancing options are enabled.

To configure a custom AP profile in the GUI:

1. Go to *WiFi & Switch Controller* > *FortiAP Profiles*.
2. Click *Create New* or edit an existing custom profile.
3. In the *Client load balancing* field, select *Frequency Handoff* and *AP Handoff*.



4. Configure the remaining settings as required.
5. Click *OK*.

To configure a custom AP profile in the CLI:

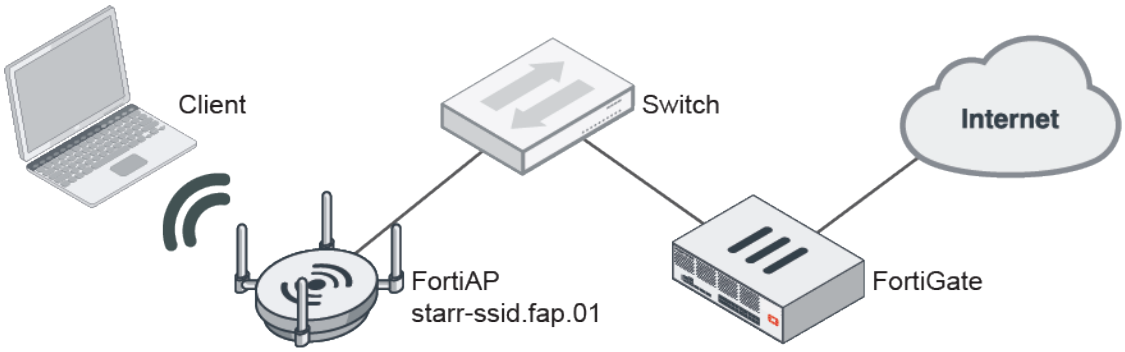
```
config wireless-controller wtp-profile
    edit "FAP421E-demo"
        config platform
            set type 421E
        end
        set handoff-sta-thresh 55
        set frequency-handoff enable
        set ap-handoff enable
        config radio-1
            set band 802.11n,g-only
        end
```

```
        config radio-2
            set band 802.11ac
        end
    next
end
```

Layer three ACL configurations for Wireless APs

For FortiAP devices (6.4.0 and later) that are managed by FortiGate, a layer three (L3) access control list (ACL) can be applied to a bridge or tunnel mode SSID.

Example



In this example:

- Rule 10 is to block all traffic to 172.16.200.44
- Rule 20 is to block all ICMP traffic
- Rule 30 is to block traffic to destination port 21 (FTP)

To configure L3 ACL:

1. Create L3 firewall rules:

```
config wireless-controller access-control-list
    edit "ACL-1"
        config layer3-ipv4-rules
            edit 10
                set dstaddr 172.16.200.44/255.255.255.255
                set action deny
            next
            edit 20
                set protocol 1
                set action deny
            next
            edit 30
                set dstport 21
                set action deny
            next
        end
    next
end
```

2. Apply the rules to VAP:

```
config wireless-controller vap
    edit "wifi.fap.01"
        set ssid "starr-ssid.fap.01"
        set passphrase **********
        set local-bridging enable
        set access-control-list "ACL-1"
    next
end
```

3. Check the rules on the FortiGate:

```
# diagnose wireless-controller wlac -c afwprof

AFWPROF (001/001) vdom,name: vdom1, ACL-1
    refcnt           : 2 own(1) wlan(1)
    deleted          : no
    Layer3 ipv4 rule : 3
        ------------------------------------------------------------------------
         ##### Policy Prot Source ==> Destination
        ------------------------------------------------------------------------
            10 deny   any  any:any ==> 172.16.200.44/32:any
            20 deny   1    any:any ==> any:any
            30 deny   any  any:any ==> any:21
        ------------------------------------------------------------------------
    wlan cnt         : 1
        vap 001 : 1    wifi.fap.01
```

4. Confirm that the L3 rules are pushed to the FortiAP:

```
# cw_diag -c afw-rules

Interface wlan00 firewall rules:
================================================================================
 RuleID HitCounter Policy Prot (IPv4)Source ==> Destination
------- ---------- ------ ---- ------------------------------------------------
     10          0 deny   any  any:any ==> 172.16.200.44/32:any
     20          0 deny   1    any:any ==> any:any
     30          0 deny   any  any:any ==> any:21
================================================================================
```

5. On the client, confirm that the rules are applied:
   a. Rule 10: Traffic to 172.16.200.44 is blocked, and traffic to other destinations are allowed:

```
root@pc_wifi:~# curl 172.16.200.44 -v
* Rebuilt URL to: 172.16.200.44/
*   Trying 172.16.200.44...
* connect to 172.16.200.44 port 80 failed: Connection timed out
* Failed to connect to 172.16.200.44 port 80: Connection timed out
* Closing connection 0
curl: (7) Failed to connect to 172.16.200.44 port 80: Connection timed out
root@pc_wifi:~#

root@pc_wifi:~# curl -k https://172.18.56.163
<html><body><h1>It works!</h1>
<p>This is the default web page for this server-44.</p>
<p>The web server software is running but no content has been added, yet. Managed by
Starr Q</p>
```

b. Rule 20: ICMP traffic is blocked and HTTPS traffic is allowed:

```
root@pc_wifi:~# ping 172.16.200.44
PING 172.16.200.44 (172.16.200.44) 56(84) bytes of data.
^C
--- 172.16.200.44 ping statistics ---
86 packets transmitted, 0 received, 100% packet loss, time 85680ms

root@pc_wifi:~# curl -k https://172.18.56.163
<html><body><h1>It works!</h1>
<p>This is the default web page for this server-44.</p>
<p>The web server software is running but no content has been added, yet. Managed by
Starr Q</p>
```

c. Rule 30: FTP traffic is blocked:

```
oot@pc_wifi:~# ftp 172.18.56.163
ftp: connect: Connection timed out
ftp> ^C
ftp> bye
```

Zero-trust Network Access

This section lists the new features added to FortiOS for zero-trust network access:

- NAC on page 124

NAC

This section includes NAC features added to FortiOS:

- IoT detection service on page 124
- Support NAC policies on switch ports on page 126
- Added ability in FortiSwitch to query FortiGuard IoT service for device details on page 130
- FortiSwitch voice device detection on page 132

IoT detection service

Internet of Things (IoT) detection is a subscription service that allows FortiGate to detect unknown devices in FortiGuard that are not detected by the local Device Database (CIDB). When the service is activated, FortiGate can send device information to the FortiGuard collection server. When a new device is detected, FortiGate queries the results from the FortiGuard query for more information about the device.

The IoT detection service requires an IOTH contract, which is part of the Enterprise and FortiCare 360 bundle, or can be purchased on its own.

FortiGate device requirements:

The FortiGate device must be:

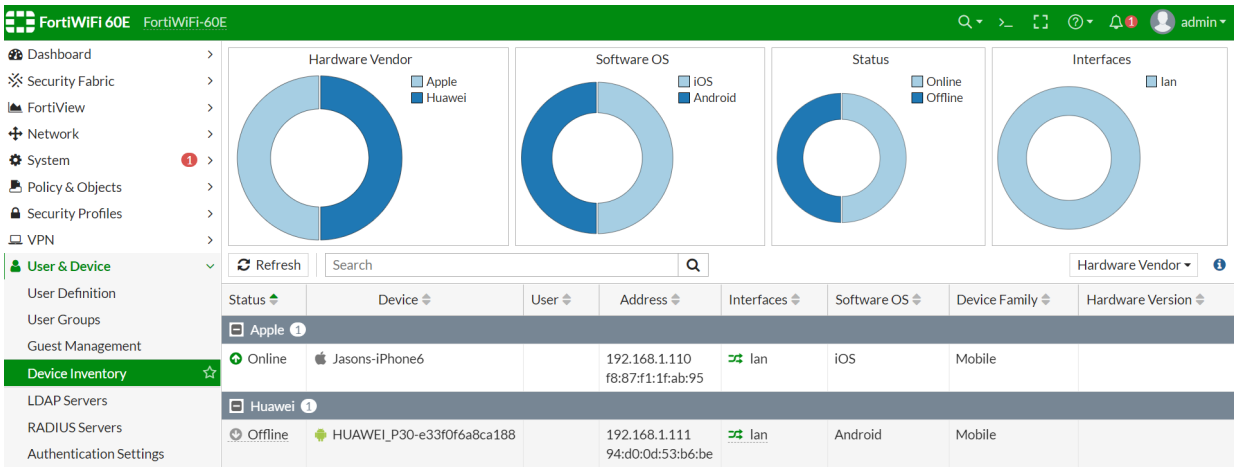
- Registered with FortiCare
- Connected to an anycast FortiGuard server

How the service works:

1. Enable Device Detection on an interface..
2. FortiGate uses the interface to detect device traffic flow.
3. Upon detecting traffic from an unknown device, FortiGate sends the device data to the FortiGuard collection server.
4. The collection server returns data about the new device to the FortiGuard query server.
5. If the device signature does not appear in the local Device Database (CIDB) or some fields are not complete, FortiGate queries FortiGuard for more information about the device.

GUI

To view the latest device information in the GUI, go to *User & Authentication* > *Device Inventory*.



To debug the daemon in the CLI:

1. Disable the local device database in order to force all queries to go to FortiGuard.
   ```
   diagnose src-vic local-sig disable
   ```
2. Enable iotd debugs.
   ```
   diagnose debug application iotd -1
   diagnose debug enable
   ```
   FortiGate sends the device data to the FortiGuard collection server.
   ```
   FortiWiFi-60E # [iotd] recv request from caller size:61
   [iotd] service:collect hostname: ip: fd:-1 request tlv_len:41
   [iotd] txt(.....y...w.....Jasons-iPhone6....579=23..)
   [iotd] hex
        (02010007017903060f77fc0203000e4a61736f6e732d6950686f6e65360204000835373 93d32330cff)
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip: fd:-1 got server hostname
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:-1 got
        server ip
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13 socket
        created
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13
        connecting
   [iotd] fd:13 monitor event:pollout
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13 build
        req packet
   [iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13 collect
        resp:1(pending)
   ```
   The FortiGuard collection server returns new device data to the FortiGuard query server.
   ```
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got query
        resp
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 id:0 total_
        len:48 header_len:16 tlv_len:32 confidence:100 mac:f8:87:f1:1f:ab:95
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
        len:32 type:1 len:6
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
        category:'Mobile'
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
        len:24 type:2 len:6
   [iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv sub_
        category:'Mobile'
   ```

```
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
     len:16 type:3 len:5
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
     vendor:'Apple'
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
     len:9 type:4 len:0
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
     len:7 type:5 len:3
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
     os:'iOS'
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 remaining_
     len:2 type:6 len:0
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 send query
     response to caller size:48
[iotd] txt(............d0 ...Mobile..Mobile..Apple....iOS..)
[iotd] hex
     (f887f11fab950000000000006430200001064d6f62696c6502064d6f62696c6503054170706c65040005
     03694f530600)
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 read resp:0
     (good)
```

3. The query returns the device information including the information source (`src fortiguard`).

```
diagnose user device list
   vd root/0  f8:87:f1:1f:ab:95  gen 26  req OUA/34
      created 503s  gen 23  seen 102s  lan  gen 7
      ip 192.168.1.110  src arp
      hardware vendor 'Apple'  src fortiguard  id 0  weight 100
      type 'Mobile'  src fortiguard  id 0  weight 100
      family 'Mobile'  src fortiguard  id 0  weight 100
      os 'iOS'  src fortiguard  id 0  weight 100
      host 'Jasons-iPhone6'  src dhcp
```

Support NAC policies on switch ports

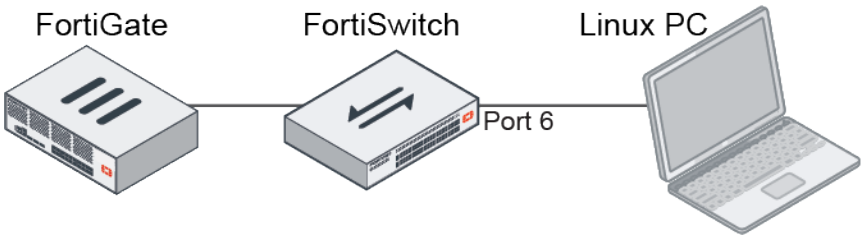
Network access control (NAC) helps administrators implement policies to control the devices and users that have access to their networks. A NAC policy can use user or detected device information, such as device type or OS, to put traffic into a specific VLAN or apply specific port settings.

The NAC function can be enabled on all FortiSwitches, or on specific FortiSwitch ports.

Initially, devices connected to ports with the NAC function enabled are put into an onboarding VLAN. The onboarding VLAN usually has a restrictive security policy, device identification enabled, a DHCP server, and captive portal enabled. The device identification feature collects device information. When the device matches the patterns that are defined in a NAC policy, an action is applied to the device, such as moving it to a specific VLAN or having a security policy applied.

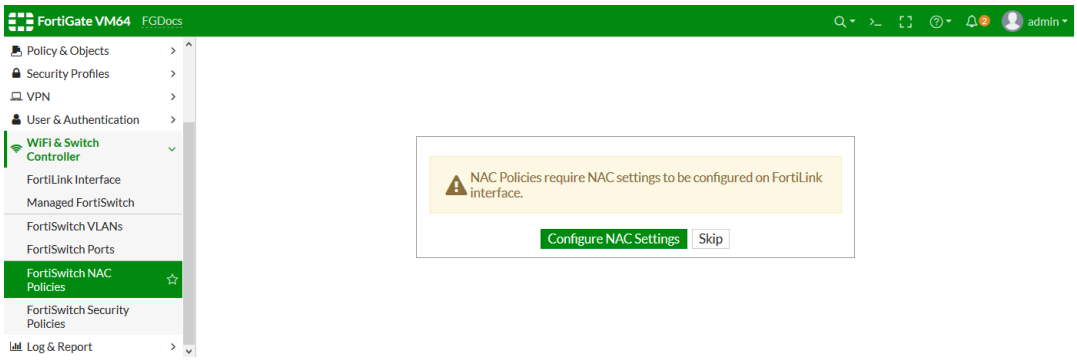
Example

In this example, NAC settings are enabled and configured so that a Linux PC is automatically moved into a VLAN dedicated to Linux PCs after it comes online and gets identified.

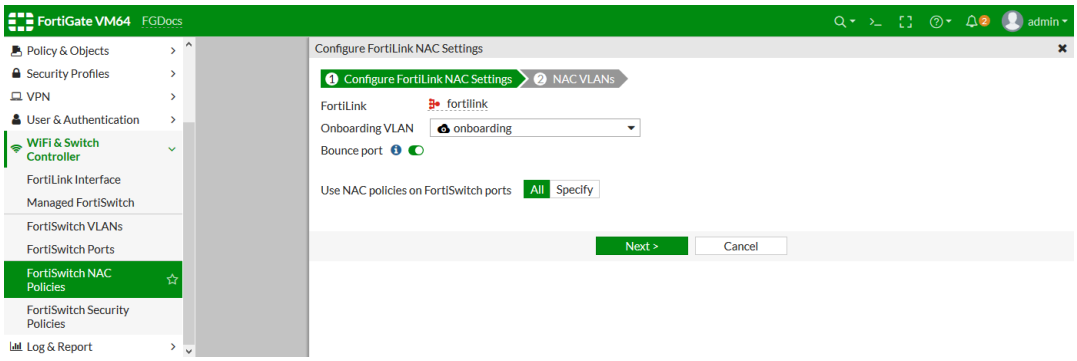


To configure a NAC policy on a switch in the GUI:

1. Use the wizard to enable the NAC feature and configure basic settings:
   a. Go to *WiFi & Switch Controller* > *FortiSwitch NAC Policies*. If FortiSwitch options are not visible, see Feature visibility for instructions on making them visible.
   b. Click *Configure NAC Settings* in the message box.

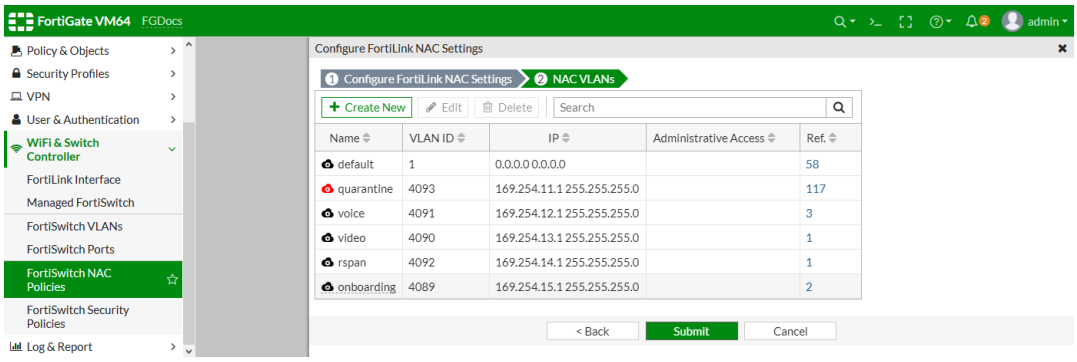


   c. Specify the switch ports that NAC access mode will be enabled on, or enable it on all of them.
   d. Select the onboarding VLAN. If no VLAN exists, click *Create* in the drop down menu to create a new NAC VLAN interface.



   e. Click *Next*.

f. Create or modify NAC VLANs (also known as FortiSwitch VLANs) that can be used in NAC policies.



g. Create or edit NAC VLANs as needed. See FortiLink Setup for details.

h. Click *Submit*.

The NAC settings can be edited in *WiFi & Switch Controller* > *FortiLink Interface*.

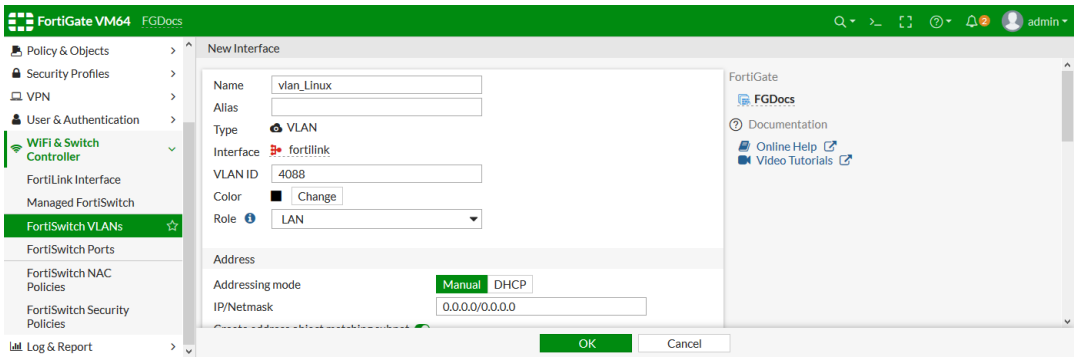
The NAC VLANs can be edited in *WiFi & Switch Controller* > *FortiSwitch VLANs*.

The access mode of the switch ports is changed to NAC and the native VLAN is set to the onboarding VLAN.

2. Create a NAC VLAN for all Linux PCs:

a. Go to *WiFi & Switch Controller* > *FortiSwitch VLANs* and click *Create New*.

b. Set *Name* to *vlan_Linux*.



c. Configure the remaining settings as required.

d. Click *OK*.

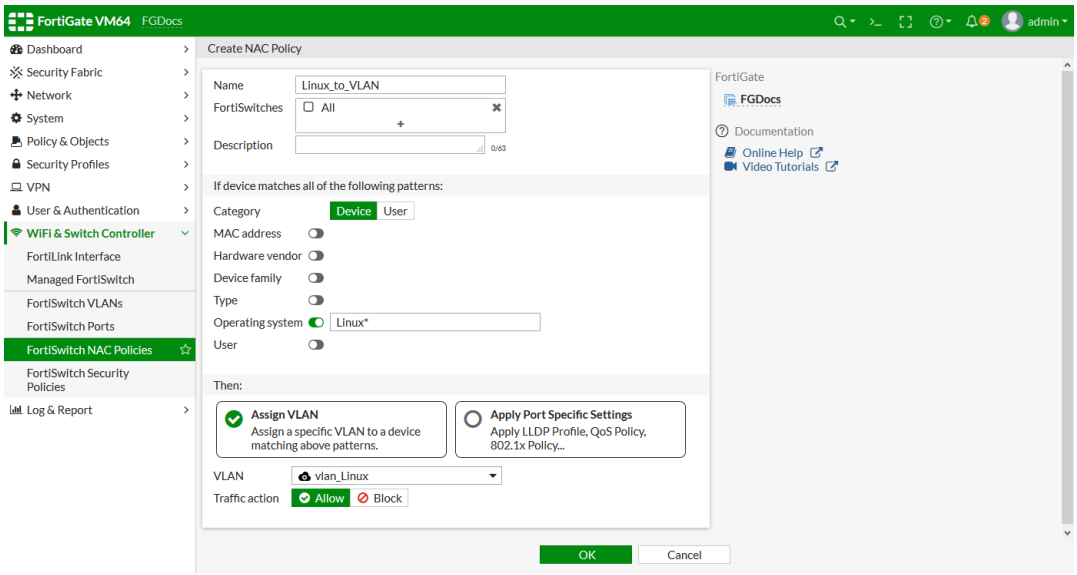
3. Create a NAC policy to match all Linux PCs and assign them to the specific VLAN:

a. Go to *WiFi & Switch Controller* > *FortiSwitch NAC Policies* and click *Create New*.

b. Enter a name for the policy, such as *Linux_to_VLAN*.

c. Enable *Operating system* and enter *Linux** in the field.

d. Select the *Assign VLAN* card and set *VLAN* to *vlan_Linux*.

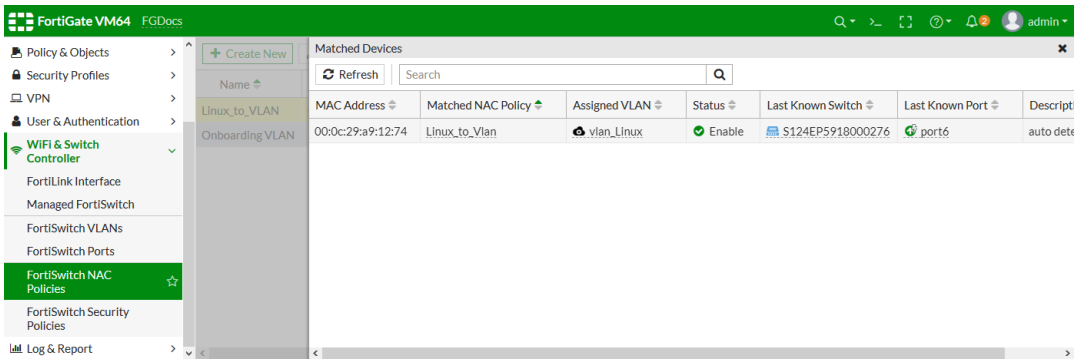


e. Click *OK*.

4. After the Linux PC connects, check that it is matched to the policy:

a. Go to *WiFi & Switch Controller* > *FortiSwitch NAC Policies*.

b. Select the *Linux_to_VLAN* policy and click *View Matched Devices*.
The *Matched Devices* pane opens, showing the devices that matched the policy.



c. Go to *WiFi & Switch Controller* > *FortiSwitch Ports*.
The port that the Linux PC is connected to will include *vlan_Linux* in the *Allowed VLANs* column.

To configure a NAC policy on a switch in the CLI:

1. Configure the FortiLink interface:

```
config system interface
    edit "fortilink"
        set vdom "root"
        set fortilink enable
        set ip 169.254.1.1 255.255.255.0
        set allowaccess ping fabric
        set type aggregate
        set member "internal11"
        set lldp-reception enable
```

```
        set lldp-transmission enable
        set snmp-index 8
        set auto-auth-extension-device enable
        set switch-controller-nac "fortilink"
    next
end
```

2. Configure the integrated NAC settings:

```
config switch-controller nac-settings
    edit "fortilink"
        set mode global
        set onboarding-vlan "onboarding"
    next
end
```

3. Configure the NAC policy matching pattern to identify matching NAC devices:

```
config user nac-policy
    edit "Linux_to_VLAN"
        set os "Linux*"
        set switch-fortilink "fortilink"
        set switch-mac-policy "Linux_to_VLAN"
    next
end
```

4. Configure the MAC policy to be applied on the managed FortiSwitch devices through the NAC device:

```
config switch-controller mac-policy
    edit "Linux_to_VLAN"
        set fortilink "fortilink"
        set vlan "vlan_Linux"
    next
end
```

5. View the NAC devices learned on the managed FortiSwitch ports that match the NAC policy:

```
show switch-controller nac-device
    config switch-controller nac-device
        edit 1
            set description "auto detected @ 2020-04-01 15:36:24"
            set mac 00:0c:29:a9:12:74
            set last-known-switch "S124EP5918000276"
            set last-known-port "port6"
            set matched-nac-policy "Linux_to_VLAN"
            set mac-policy "Linux_to_VLAN"
        next
    end
```

Added ability in FortiSwitch to query FortiGuard IoT service for device details

Capability was added to FortiSwith to work with FortiGate and the new FortiGuard IoT detection service for the purpose of device identification.

FortiSwitch devices are now able to assist FortiGates with capturing the most accurate device information, allowing FortiGate to identify devices for the user device list. When the new FortiGuard IoT detection service is activated, FortiGate will leverage the IoT detection service to help reduce the workload for device identification.

To use this feature, the following are required:
An IoT detection service subscription. See IoT detection service on page 124.
FortiSwitch 2.0.3 and higher.

The following CLI command and parameters were added under `switch-controller` to control when FortiSwitch should start and stop collecting device packets for FortiGate:

```
config switch-controller system
   set iot-weight-threshold
   set iot-scan-interval
   set iot-holdoff
   set iot-mac-idle
```

Parameter	Description	Type	Defaults
iot-weight-threshold	The confidence value for the MAC entry. The Value is re-queried when it is below this value.	Integer	• Default = 1 • Disable = 0
iot-scan-interval	The IoT scan interval.	Integer	• Minimum minutes = 2 • Maximum minutes = 4294967295 • Default = 60 minutes • Disable = 0
iot-holdoff	The creation time for the MAC entry. The time must be greater than this value for an entry to be created.	Integer	Default = 5 minutes
iot-mac-idle	The idle time for the MAC entry. The MAC entry is removed after this value.		Default = 1440 minutes

Example

Example topology

FGT500E-----FSW248EP(port1)-----FortiAP

In this example, FortiSwitch will help FortiGate collect packets from FortiAP every 30 minutes and stop for 30 minutes. FortiSwitch will stop collecting packets from FortiAP when the weight of the device information reaches a threshold of 80.

To collect IoT device information for identification in the CLI:

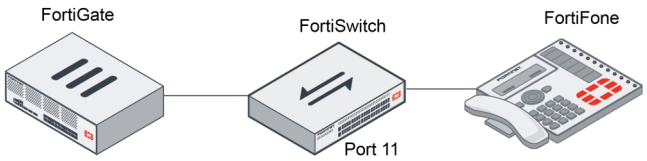
1. This CLI command is configured with the IoT parameters.
```
FGT_A (global) # config switch-controller system
FGT_A (system) # get
iot-weight-threshold: 80
iot-scan-interval   : 30
iot-holdoff         : 5
iot-mac-idle        : 1440
FGT_A (system) # end
```
2. When the scheduled time to capture the packets is reached, the diagnose command initiates the scan.
```
FGT_A (vdom1) # dia switch-controller traffic-capture show
MAC                   session-in-use  switch                 fortilink-interface-name
     port             status
=============================================================================================
     ================================
08:5b:0e:06:6a:d4     1               S248EPTF18001384       port11
     port1  running
Global stats:
================
node add = 16
node delete = 15
node add failed = 0
node delete failed = 0
```
3. A corresponding sniffer profile is created on FortiSwitch to help collect the data.
```
S524DN4K16000116 # config system sniffer-profile
S524DN4K16000116 (sniffer-profile) # show
config system sniffer-profile
   edit "08:5b:0e:06:6a:d4"
      set filter "ether host 08:5b:0e:06:6a:d4"
      set max-pkt-count 1000
      set max-pkt-len 256
      set switch-interface "port1"
   next
end
```
4. The data is collected and sent to the FortiGuard service for identification. The device information is updated in the device list with *src fortiguard*.
```
FGT_A (vdom1) # dia user device list
hosts
   vd vdom1/1  08:5b:0e:06:6a:d4  gen 17  req OUA/34
      created 42s  gen 13  seen 1s  onboarding.13  gen 4
      hardware vendor 'FORTINET'  src fortiguard  id 0  weight 100
      type 'Network'  src fortiguard  id 0  weight 100
      family 'Router'  src fortiguard  id 0  weight 100
      os 'NULL'  src fortiguard  id 0  weight 100
      hardware version 'FortiAP-320B'   id 0  weight 100
      host 'FP320B3X13000599'  src capwap
```

FortiSwitch voice device detection

FortiSwitch is able to parse LLDP messages from voice devices such as FortiFone, and pass this information to FortiGate for device detection. You can use FortiSwitch NAC policies to assign a device to an LLDP profile, QoS policy, and VLAN policy. When a detected device is matched to a NAC policy, the corresponding policy actions will be applied on the switch port.

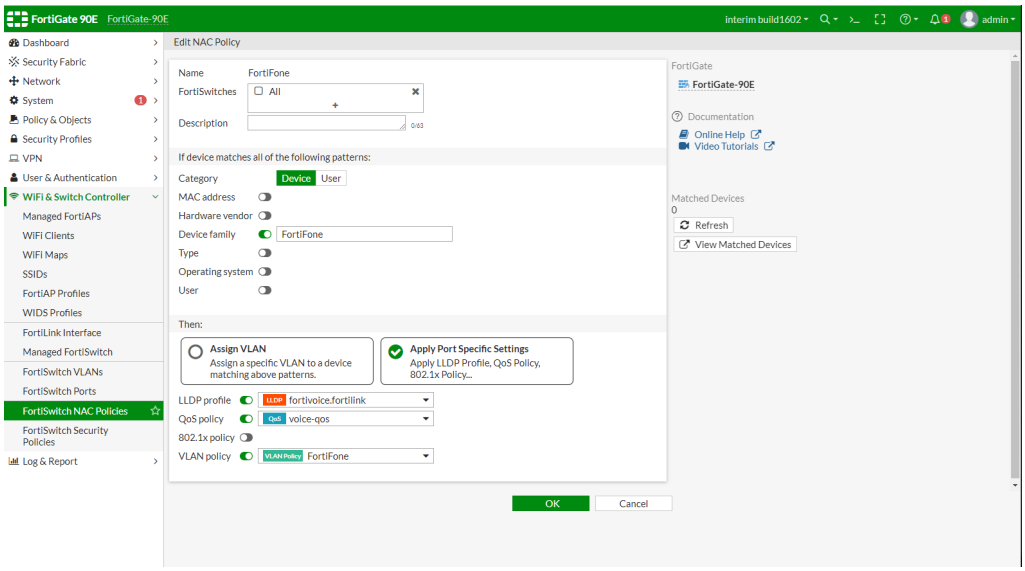
Example

In the following example, FortiFone is connected to port11 of FortiSwitch. A NAC policy is created to apply a VLAN policy, LLDP policy, and QoS policy to Device Family FortiFone.

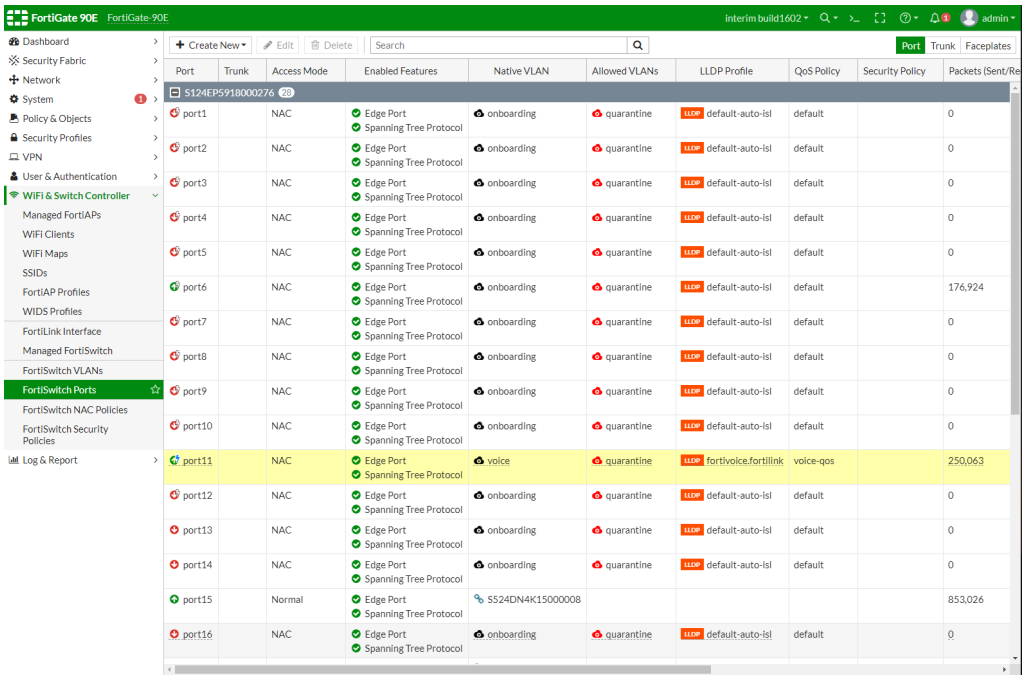


To create a FortiSwitch NAC policy in the GUI:

1. Configure a NAC policy on a switch port. See Support NAC policies on switch ports on page 126.
2. Go to *WiFi & Switch Controller* > *FortiSwitch NAC Policies*.
3. Create or edit an NAC policy.
4. Configure the NAC policy settings.
   a. Set the *Category* to *Device*.
   b. Enable *Device family*, and enter name such as *FortiFone*.
   c. Select *Apply Port Specific Settings*.
   d. Enable *LLDP profile*, and select a voice profile from the dropdown.
   e. Enable *QoS policy*, and select a voice policy from the dropdown.
   f. Enable *VLAN policy*, and select a voice policy from the dropdown.



The NAC policy is applied after a FortiFone is plugged into port11 of the FortiSwitch:



To create a FortiSwitch NAC policy in the CLI:

1. Assign the FortiFone to a VLAN policy, LLDP policy, and QoS Policy.

```
config user nac-policy
   edit "FortiFone"
      set family "FortiFone"
      set switch-fortilink "fortilink"
      set switch-port-policy "FortiFone"
   next
end

config switch-controller port-policy
   edit "FortiFone"
      set fortilink "fortilink"
      set lldp-profile "fortivoice.fortilink"
      set qos-policy "voice-qos"
      set vlan-policy "fortiFone"
   next
end

config switch-controller vlan-policy
   edit "fortiFone"
      set fortilink "fortilink"
      set vlan "voice"
   next
end

config switch-controller lldp-profile
   edit "fortivoice.fortilink"
      set med-tlvs inventory-management network-policy location-identification
      set auto-isl disable
      config med-network-policy
         edit "voice"
```

```
            set status enable
            set vlan-intf "voice"
            set assign-vlan enable
            set dscp 46
         next
         edit "voice-signaling"
            set status enable
            set vlan-intf "voice"
            set assign-vlan enable
            set dscp 46
         next
         edit "guest-voice"
         next
         edit "guest-voice-signaling"
         next
         edit "softphone-voice"
         next
         edit "video-conferencing"
         next
         edit "streaming-video"
         next
         edit "video-signaling"
         next
      end
   next
end

config switch-controller qos qos-policy
   edit "voice-qos"
      set trust-dot1p-map "voice-dot1p"
      set trust-ip-dscp-map "voice-dscp"
      set queue-policy "voice-egress"
   next
end
```

2. FortiSwitch receives an LLDP message from FortiFone after it is plugged into port11.
3. Configure `dia switch-controller switch-info` to check the device information on FortiGate. The FortiFone is identified.

```
FortiGate-90E # dia switch-controller switch-info lldp neighbors-detail S124EP5918000276
     port11
Vdom: root
Managed Switch : S124EP5918000276 0

Capability codes:
   R:Router, B:Bridge, T:Telephone, C:DOCSIS Cable Device
   W:WLAN Access Point, P:Repeater, S:Station, O:Other

MED TLV Capability codes:
   C:Capabilities, P:Network Policies, L:Location, S:MDI PSE
   D:MDI PD, I:Inventory


_____________________________________________________________
   Neighbor learned on port port11 by LLDP protocol
   Last change 20 seconds ago
   Last packet received 20 seconds ago

   Chassis ID: 169.254.15.3 (ip)
```

```
   System Name: FON-675i
   System Description:
   :14.0.0.1.r4

   Time To Live: 60 seconds
   System Capabilities: BT
   Enabled Capabilities: BT
   MED type: Communication Device Endpoint (Class III)
   MED Capabilities: CP
   Management IP Address: 169.254.15.3

   Port ID: 70:4c:a5:e2:6b:b2 (mac)
   Port description: WAN Port 10M/100M/1000M
   IEEE802.3, Power via MDI:
      Power devicetype: PD
      PSE MDI Power: Not Supported
      PSE MDI Power Enabled: No
      PSE Pair Selection: Can not be controlled
      PSE power pairs: Signal
      Power class: 1 (class-0)
      Power type: 802.3at off
      Power source: Unknown
      Power priority: Unknown
      Power requested: 0.0W
      Power allocated: 0.0W
LLDP-MED, Network Policies:
   voice: VLAN: 256 (untagged), Priority: 0 DSCP: 46
   voice-signaling: VLAN: 256 (untagged), Priority: 0 DSCP: 46
   streaming-video: VLAN: 256 (untagged), Priority: 0 DSCP: 46

FortiGate-90E # dia user device list
hosts
   vd root/0  70:4c:a5:e2:6b:b2  gen 5  req OUA/34
      created 3522s  gen 3  seen 24s  onboarding  gen 2
      hardware vendor 'Fortinet'  src lldp weight 128
      type 'IP Phone'  src lldp  id 1523  weight 128
      family 'FortiFone'  src lldp  id 1523  weight 128
      host 'FON-675i'  src lldp
```

AI-driven Security Operations

This section lists the new features added to FortiOS for AI-driven security operations:

- ATP on page 137

ATP

This section includes ATP (advanced threat protection) features added to FortiOS:

- Credential phishing prevention on page 137
- Extend ISDB to include well-known MAC address list on page 139
- GeoIP matching by registered and physical location on page 141

Credential phishing prevention

When credential phishing prevention is enabled, the FortiGate scans user names and passwords in submission traffic to external URLs against the sensitive corporate network credentials stored in the corporate domain controller. Based on the configured antiphishing rules in proxy mode web filter profiles, the FortiGate will block the URL or alert the user.

Credential phishing prevention includes the following components:

- The antiphishing profile that defines the corporate network credentials stored in the domain controller, antiphishing check option, default action if no rules match, antiphishing status, and so on.
- The antiphishing table that defines specific URLs and what antiphishing action (block, log, exempt) should be taken when the URL submission request matches.
- The FortiGuard category-based antiphishing rule that defines what action will occur if the submission request matches a FortiGuard category.



The antiphishing table and FortiGuard category-based antiphishing rule behavior is different than the `urlfilter` and FortiGuard filter:

- If a request is blocked by the `urlfilter` or FortiGuard filter, there will be no further antiphishing scanning.
- Antiphishing scanning occurs only after the local `urlfilter` and FortiGuard filter allow traffic. If a submission matches the antiphishing table, the defined action will be taken. No further FortiGuard category-based rules will be applied.
- Similar to firewall rules, the antiphishing table and FortiGuard category-based antiphishing rule use a top-down priority. Whichever rules matches first is the one that is used.

To configure credential phishing prevention:

1. Configure the antiphishing profile, which includes the FortiGuard category rule:

```
config webfilter profile
    edit "<profile-name>"
        ...
        config web
            ...
        end
        config antiphish
            set status enable
            set domain-controller "win2016"
            set default-action block
            set check-uri enable
            set check-basic-auth enable
            set max-body-len 65536
            config inspection-entries
                edit "inspect-37"
                    set fortiguard-category 37
                    set action block
                next
                edit "inspect-others"
                    set fortiguard-category all
                    set action log
                next
            end
            config custom-patterns
                edit "customer-name"
                    set category username
                next
                edit "customer-passwd"
                   set category password
                next
            end
        end
        ...
        set web-antiphishing-log enable
    next
end
```

   In this example, URLs that match the FortiGuard category 37 (social networking) will be blocked and other categories will be logged.

   Besides built-in user name and password keyword patterns, you can configure additional keyword patterns with `config custom-patterns`.

2. Add the antiphishing attribute to the `urlfilter` table:

```
config webfilter urlfilter
    edit 1
        set name "antiphish-table"
        set antiphish enable
        config entries
            edit 1
                set url "www.example.com"
                set type simple
                set antiphish-action block
```

```
                set status enable
                set referrer-host ''
            next
        end
    next
end
```

3. Add the antiphishing table to the `webfilter profile`.
   The `antiphish-table` references a `urlfilter` table with the `antiphish` attribute enabled. The `urlfilter-table` and `antiphish-table` cannot refer to the same list number:

```
config webfilter profile
    edit "<profile-name>"
        config web
            set urlfilter-table 4
            set antiphish-table 1
        end
        ...
     next
end
```

Extend ISDB to include well-known MAC address list

ISDB now includes well-known vendor MAC address range lists. The lists can only be used for source MAC addresses in IPv4 policies, and include the vendor name and the MAC address ranges that the vendor belongs to.

To view the vendor list:

```
# diagnose vendor-mac id
Please input Vendor MAC ID.
ID: 1 name: "Asus"
ID: 2 name: "Acer"
ID: 3 name: "Amazon"
ID: 4 name: "Apple"
ID: 5 name: "Xiaomi"
ID: 6 name: "BlackBerry"
ID: 7 name: "Canon"
ID: 8 name: "Cisco"
ID: 9 name: "Linksys"
ID: 10 name: "D-Link"
ID: 11 name: "Dell"
ID: 12 name: "Ericsson"
ID: 13 name: "LG"
ID: 14 name: "Fujitsu"
ID: 15 name: "Fitbit"
ID: 16 name: "Fortinet"
ID: 17 name: "OPPO"
ID: 18 name: "Hitachi"
ID: 19 name: "HTC"
ID: 20 name: "Huawei"
ID: 21 name: "HP"
ID: 22 name: "IBM"
ID: 23 name: "Juniper"
ID: 24 name: "Lenovo"
```

```
ID: 25 name: "Microsoft"
ID: 26 name: "Motorola"
ID: 27 name: "Netgear"
ID: 28 name: "Nokia"
ID: 29 name: "Nintendo"
ID: 30 name: "PaloAltoNetworks"
ID: 31 name: "Polycom"
ID: 32 name: "Samsung"
ID: 33 name: "Sharp"
ID: 34 name: "Sony"
ID: 35 name: "Toshiba"
ID: 36 name: "VMware"
ID: 37 name: "Vivo"
ID: 38 name: "Zyxel"
ID: 39 name: "ZTE"
```

To view the MAC address ranges for a vendor:

```
# diagnose vendor-mac id 16
Vendor MAC: 16(Fortinet)
Version: 0000700021
Timestamp: 201908081432
Number of MAC ranges: 6
00:09:0f:00:00:00 - 00:09:0f:ff:ff:ff
04:d5:90:00:00:00 - 04:d5:90:ff:ff:ff
08:5b:0e:00:00:00 - 08:5b:0e:ff:ff:ff
70:4c:a5:00:00:00 - 70:4c:a5:ff:ff:ff
90:6c:ac:00:00:00 - 90:6c:ac:ff:ff:ff
e8:1c:ba:00:00:00 - e8:1c:ba:ff:ff:ff
```

To query the vendor of a specific MAC address or range:

```
# diagnose vendor-mac match 00:09:0f:ff:ff:ff 48
Vendor MAC: 16(Fortinet), matched num: 1
```

To use the vendor ID in a firewall policy:

```
config firewall policy
    edit 9
        set name "policy_id_9"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set src-vendor-mac 36 16
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Only packets whose source MAC address belong to Fortinet or VMware are passed by the policy.

GeoIP matching by registered and physical location

IP addresses have both a physical and registered location in the geography IP database. Sometimes these two locations are different. The new `geoip-match` command allows users to match an IP address in an IPv4 policy to its physical or registered location when a GeoIP is used as a source or destination address.

In the following example, the physical location of 220.243.219.10 is CA (Canada), the registered location is CN (China), and it is not an anycast IP.

To configure GeoIP matching based on registered location:

1. Create a firewall policy to match the IP:

```
config firewall policy
    edit 1
        set name "policy_id_1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "test-geoip-CA"
        set action accept
        set schedule "always"
        set service "ALL"
        set geoip-match registered-location
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

   Since CA is applied as a destination address and registered location IP matching is enabled, if the destination IP of the traffic is 220.243.219.10, then the traffic will be blocked because the registered location is CN.

2. Verify that the policy is blocking traffic from the IP address:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
5.383798 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
6.381982 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
7.382608 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
^C
3 packets received by filter
0 packets dropped by kernel
```

To configure GeoIP matching based on physical location:

1. Create a firewall policy to match the IP:

```
config firewall policy
    edit 1
        set name "policy_id_1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "test-geoip-CA"
        set action accept
```

```
        set schedule "always"
        set service "ALL"
        set geoip-match physical-location
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Since CA is applied as a destination address and physical location IP matching is enabled, if the destination IP of the traffic is 220.243.219.10, then the traffic will pass through.

2. Verify that the policy is allowing traffic from the IP address:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
5.273985 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
5.274176 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
6.274426 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
6.274438 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
7.273978 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
7.273987 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
^C
6 packets received by filter
0 packets dropped by kernel
```

Fabric Management Platform

This section lists the new features added to FortiOS for the Security Fabric management platform:

- Single pane on page 143
- Security Fabric automation on page 153

Single pane

This section includes single pane features added to FortiOS:

- SAML SP for VPN authentication on page 143
- Display cloud service communications statistics on page 146
- Confirmation prompt when creating new VDOMs on page 147
- Admin profile option for diagnostic access on page 148
- Override FortiAnalyzer and syslog server settings on page 148
- FortinetOne renamed FortiCloud on page 152

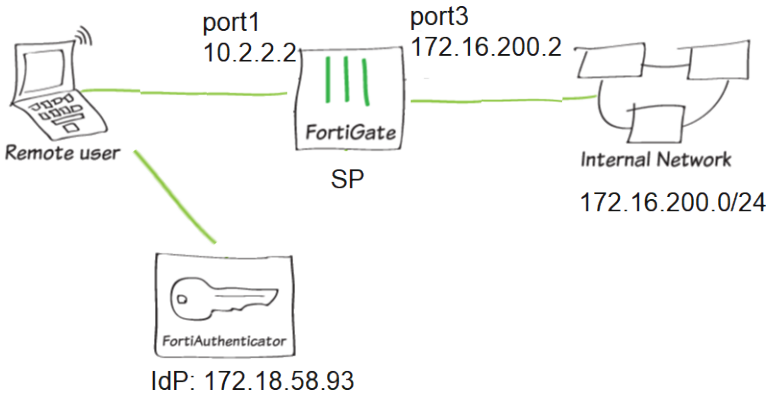
SAML SP for VPN authentication

When you configure a FortiGate as a service provider (SP), you can create an authentication profile that uses SAML for both firewall and SSL VPN web portal authentication. Once the firewall is authenticated, entering SAML credentials is not required for SSL VPN web portal authentication.



You must use the identity provider's (IdP) remote certificate on the SPs.

The following example uses a FortiGate as an SP and FortiAuthenticator as the IdP server:



To configure firewall authentication:

1. Configure the FortiGate SP to be a SAML user:

```
config user saml
    edit "fac-firewall"
        set entity-id "http://10.2.2.2:1000/saml/metadata/"
        set single-sign-on-url "https://10.2.2.2:1003/saml/login/"
        set single-logout-url "https://10.2.2.2:1003/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/bbbbbb/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-idp/bbbbbb/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-idp/bbbbbb/logout/"
        set idp-cert "REMOTE_Cert_3"
        set user-name "username"
        set group-name "group"
    next
end
```

2. Add the SAML user to the user group (optionally, you can configure group matching):

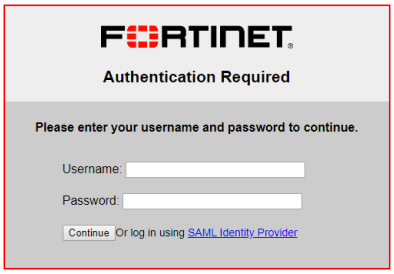
```
config user group
    edit "saml_firewall"
        set member "fac-firewall"
        config match
            edit 1
                set server-name "fac-firewall"
                set group-name "user_group1"
            next
        end
    next
end
```

3. Add the SAML user group to a firewall policy:

```
config firewall policy
    edit 2
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "pc4"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set fsso disable
        set groups "saml_firewall" "group_local"
        set users "first"
        set nat enable
    next
end
```

4. Configure the FortiAuthenticator IdP as needed.

5. Run HTTP/HTTPS authentication for a remote user. The SAML login page appears:



To configure SSL VPN web portal authentication:

1. Configure the FortiGate SP to be a SAML user:

```
config user saml
    edit "fac-sslvpn"
        set entity-id "https://10.2.2.2:10443/remote/saml/metadata/"
        set single-sign-on-url "https://10.2.2.2:10443/remote/saml/login/"
        set single-logout-url "https://10.2.2.2:10443/remote/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/ssssss/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-idp/ssssss/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-idp/ssssss/logout/"
        set idp-cert "REMOTE_Cert_3"
        set user-name "username"
    next
end
```

2. Add the SAML user to the user group (group matching may also be configured):

```
config user group
    edit "saml_sslvpn"
        set member "fac-sslvpn"
    next
end
```

3. Configure SSL VPN:

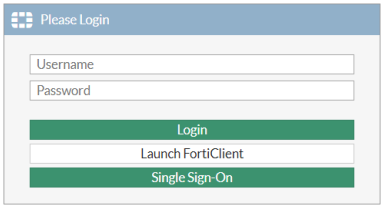
```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "port3"
    set source-address "all"
    set source-address6 "all"
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "saml_sslvpn"
            set portal "web-access"
        next
    end
end
```

4. Add the SAML user group to a firewall policy:

```
config firewall policy
    edit 8
```

```
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set groups "local" "saml_sslvpn"
        set nat enable
    next
end
```

5. Configure the FortiAuthenticator IdP as needed.
6. Run SSL VPN web mode authentication for a remote user. The SAML login page appears:

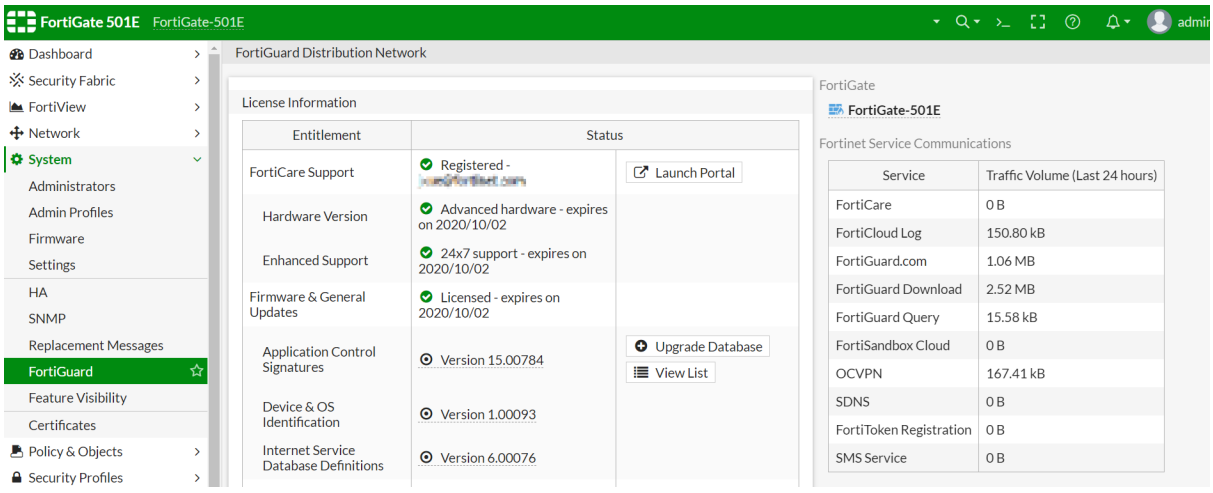


Display cloud service communications statistics

Fortinet service communications statistics are displayed on the *FortiGuard* page. The statistics correspond with the output from `diagnose system service-communication`. The values for traffic volume in the GUI are sums of data from the last 24 hours.

To view Fortinet service communications statistics:

1. Go to *System* > *FortiGuard*.
   The *Fortinet Service Communications* statistics are displayed on the right-side of the screen:



   These statistics correspond to the following:

```
# diagnose system service-communication
FortiCare:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
```

```
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiGuard Download:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 195752 0 21051904 36342800 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0
The last 7 days(in bytes):  57590456 0 0 0 0 0 0
FortiGuard Query:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 2805 0 1298 1709 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  5812 0 0 0 0 0 0
FortiCloud Log:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiSandbox Cloud:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiGuard.com:
The last 1 hour(in bytes):  2014 0 1329 0 1329 0 1329 0 1329 0 2020 103930
The last 24 hours(in bytes):  3343 112595 9032861 18584815 17757745 16054191 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  61545550 0 0 0 0 0 0
FortiToken Registration:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
SMS Service:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
```

Confirmation prompt when creating new VDOMs

A VDOM confirmation prompt has been added so users do not create new VDOMs accidentally in the CLI. This setting is disabled by default. Once enabled, when an administrator creates a new VDOM, the FortiGate displays a prompt to confirm before the VDOM is created.

To use the VDOM confirmation prompt:

1. Enable the prompt:

```
config system global
    set edit-vdom-prompt enable
end
```

2. Create a new VDOM:

```
(global) # config vdom
    edit vdomtest1
    The input VDOM name doesn't exist.
    Do you want to create a new VDOM?
    Please press 'y' to continue, or press 'n' to cancel. (y/n)y
```

```
        current vf=vdomtest1:4

        next
        edit vdomtest2
        The input VDOM name doesn't exist.
        Do you want to create a new VDOM?
        Please press 'y' to continue, or press 'n' to cancel. (y/n)n

    end
```

Admin profile option for diagnostic access

The `system-diagnostics` command in an administrator profile can be used to control access to diagnose commands for global and VDOM level administrators.

To block an administrator's access to diagnose commands:

1. Create an admin profile that cannot access diagnose commands:

```
config system accprofile
    edit "nodiagnose"
        ...
        set system-diagnostics disable
    next
end
```

2. Apply the profile to an administrator:

```
config system admin
    edit "nodiag"
        set accprofile "nodiagnose"
        set vdom "root"
        set password ********
    next
end
```

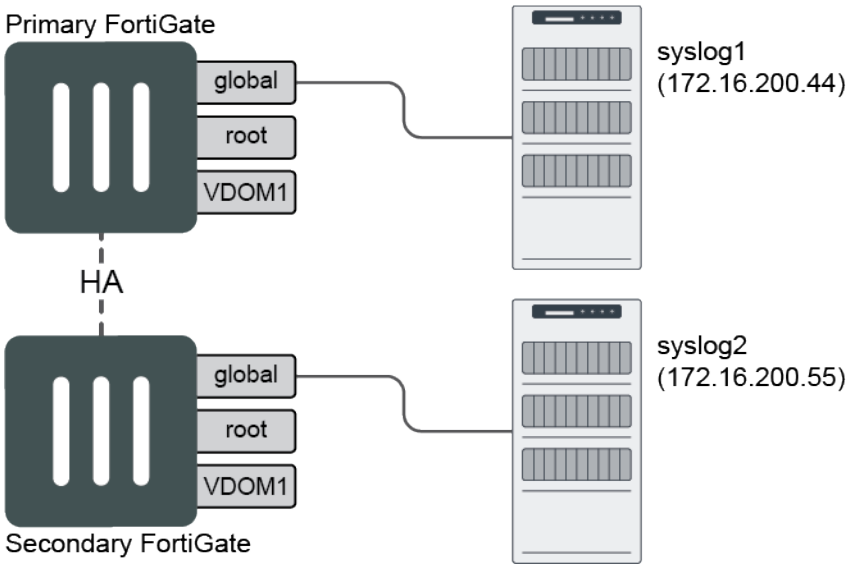
3. Log in as the administrator and confirm that they cannot access diagnose commands:

```
$ ?
config     Configure object.
get        Get dynamic and system information.
show       Show configuration.
execute    Execute static commands.
alias      Execute alias commands.
exit       Exit the CLI.
```

Override FortiAnalyzer and syslog server settings

In an HA cluster, secondary devices can be configured to use different FortiAnalyzer devices and syslog servers than the primary device. VDOMs can also override global syslog server settings.

Configure a different syslog server on a secondary HA device



To configure the primary HA device:

1. Configure a global syslog server:

```
config global
    config log syslog setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

2. Set up a VDOM exception to enable setting the global syslog server on the secondary HA device:

```
config global
    config system vdom-exception
        edit 1
            set object log.syslogd.setting
        next
    end
end
```

To configure the secondary HA device:

1. Configure a global syslog server:

```
config global
    config log syslogd setting
        set status enable
        set server 172.16.200.55
        set facility local5
    end
end
```

2. After the primary and secondary device synchronize, generate logs on the secondary device.

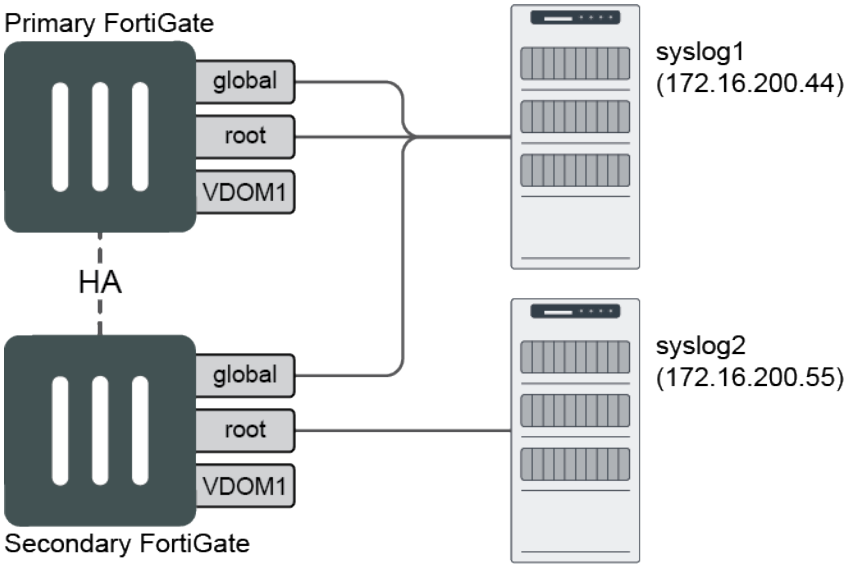
To confirm that logs are been sent to the syslog server configured on the secondary device:

1. On the primary device, retrieve the following packet capture from the secondary device's syslog server:

```
# diagnose sniffer packet any "host 172.16.200.55" 6
interfaces=[any]
filters=[host 172.16.200.55]

266.859494 port2 out 172.16.200.2.7434 -> 172.16.200.55.514: udp 278
0x0000   0000 0000 0000 0009 0f09 0004 0800 4500        ..............E.
0x0010   0132 f3c7 0000 4011 9d98 ac10 c802 ac10        .2....@.........
0x0020   c837 1d0a 0202 011e 4b05 3c31 3734 3e64        .7......K.<174>d
0x0030   6174 653d 3230 3230 2d30 332d 3134 2074        ate=2020-03-14.t
0x0040   696d 653d 3132 3a30 303a 3035 2064 6576        ime=12:00:05.dev
0x0050   6e61 6d65 3d22 466f 7274 6947 6174 652d        name="FGT-81E-Sl
0x0060   3831 455f 4122 2064 6576 6964 3d22 4647        ave-A".devid="FG
0x0070   5438 3145 3451 3136 3030 3030 3438 2220        T81E4Q16000048".
0x0080   6c6f 6769 643d 2230 3130 3030 3230 3032        logid="010002002
0x0090   3722 2074 7970 653d 2265 7665 6e74 2220        7".type="event".
0x00a0   7375 6274 7970 653d 2273 7973 7465 6d22        subtype="system"
0x00b0   206c 6576 656c 3d22 696e 666f 726d 6174        .level="informat
0x00c0   696f 6e22 2076 643d 2276 646f 6d31 2220        ion".vd="vdom1".
0x00d0   6576 656e 7474 696d 653d 3135 3834 3231        eventtime=158421
0x00e0   3234 3035 3835 3938 3335 3639 3120 747a        2405859835691.tz
0x00f0   3d22 2d30 3730 3022 206c 6f67 6465 7363        ="-0700".logdesc
0x0100   3d22 4f75 7464 6174 6564 2072 6570 6f72        ="Outdated.repor
0x0110   7420 6669 6c65 7320 6465 6c65 7465 6422        t.files.deleted"
0x0120   206d 7367 3d22 4465 6c65 7465 2031 206f        .msg="Delete.1.o
0x0130   6c64 2072 6570 6f72 7420 6669 6c65 7322        ld.report.files"
```

Configure a different syslog server in the root VDOM on a secondary HA device



To configure the primary HA device:

1. Configure a global syslog server:

```
config global
    config log syslog setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

2. Set up a VDOM exception to enable `syslog-override` in the secondary HA device root VDOM:

```
config global
    config system vdom-exception
        edit 1
            set object log.syslogd.override-setting
            set scope inclusive
            set vdom root
        next
    end
end
```

3. In the VDOM, enable `syslog-override` in the log settings, and set up the override syslog server:

```
config root
    config log setting
        set syslog-override enable
    end
    config log syslog override-setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

After `syslog-override` is enabled, an override syslog server must be configured, as logs will not be sent to the global syslog server.

To configure the secondary HA device:

1. Configure an override syslog server in the root VDOM:

```
config root
    config log syslogd override-setting
        set status enable
        set server 172.16.200.55
        set facility local5
        set format default
    end
end
```

2. After the primary and secondary device synchronize, generate logs in the root VDOM on the secondary device.

To confirm that logs are been sent to the syslog server configured for the root VDOM on the secondary device:

1. On the primary device, retrieve the following packet capture from the syslog server configured in the root VDOM on the secondary device:

```
# diagnose sniffer packet any "host 172.16.200.55" 6
interfaces=[any]
filters=[host 172.16.200.55]

156.759696 port2 out 172.16.200.2.1165 -> 172.16.200.55.514: udp 277
0x0000   0000 0000 0000 0009 0f09 0004 0800 4500        ..............E.
0x0010   0131 f398 0000 4011 9dc8 ac10 c802 ac10        .1....@.........
0x0020   c837 048d 0202 011d af5f 3c31 3734 3e64        .7......._<174>d
0x0030   6174 653d 3230 3230 2d30 332d 3134 2074        ate=2020-03-14.t
0x0040   696d 653d 3131 3a33 353a 3035 2064 6576        ime=11:35:05.dev
0x0050   6e61 6d65 3d22 466f 7274 6947 6174 652d        name="FGT-81E-Sl
0x0060   3831 455f 4122 2064 6576 6964 3d22 4647        ave-A".devid="FG
0x0070   5438 3145 3451 3136 3030 3030 3438 2220        T81E4Q16000048".
0x0080   6c6f 6769 643d 2230 3130 3030 3230 3032        logid="010002002
0x0090   3722 2074 7970 653d 2265 7665 6e74 2220        7".type="event".
0x00a0   7375 6274 7970 653d 2273 7973 7465 6d22        subtype="system"
0x00b0   206c 6576 656c 3d22 696e 666f 726d 6174        .level="informat
0x00c0   696f 6e22 2076 643d 2272 6f6f 7422 2065        ion".vd="root".e
0x00d0   7665 6e74 7469 6d65 3d31 3538 3432 3130        venttime=1584210
0x00e0   3930 3537 3539 3334 3132 3632 2074 7a3d        905759341262.tz=
0x00f0   222d 3037 3030 2220 6c6f 6764 6573 633d        "-0700".logdesc=
0x0100   224f 7574 6461 7465 6420 7265 706f 7274        "Outdated.report
0x0110   2066 696c 6573 2064 656c 6574 6564 2220        .files.deleted".
0x0120   6d73 673d 2244 656c 6574 6520 3220 6f6c        msg="Delete.2.ol
0x0130   6420 7265 706f 7274 2066 696c 6573 22          d.report.files"
```

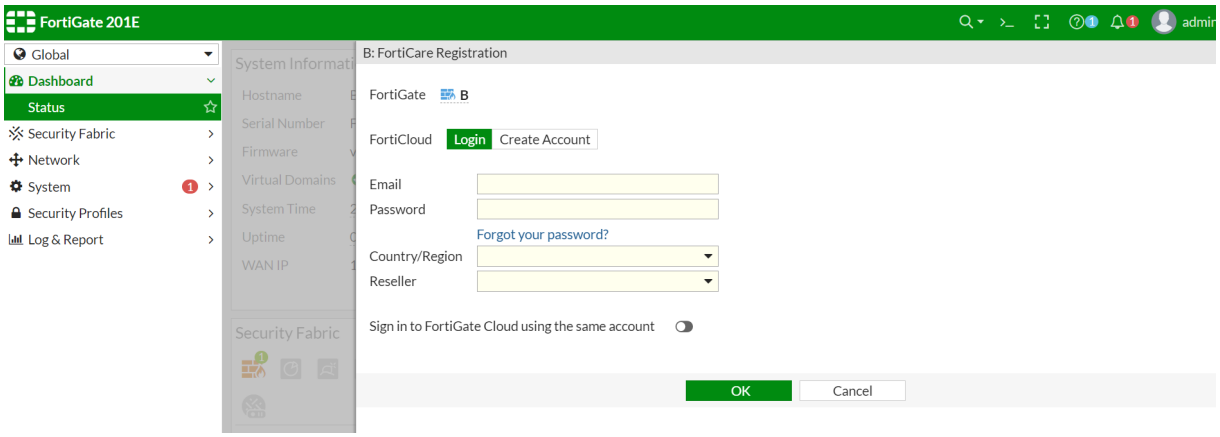
FortinetOne renamed FortiCloud

FortinetOne has been renamed FortiCloud in the FortiGate Cloud widget.

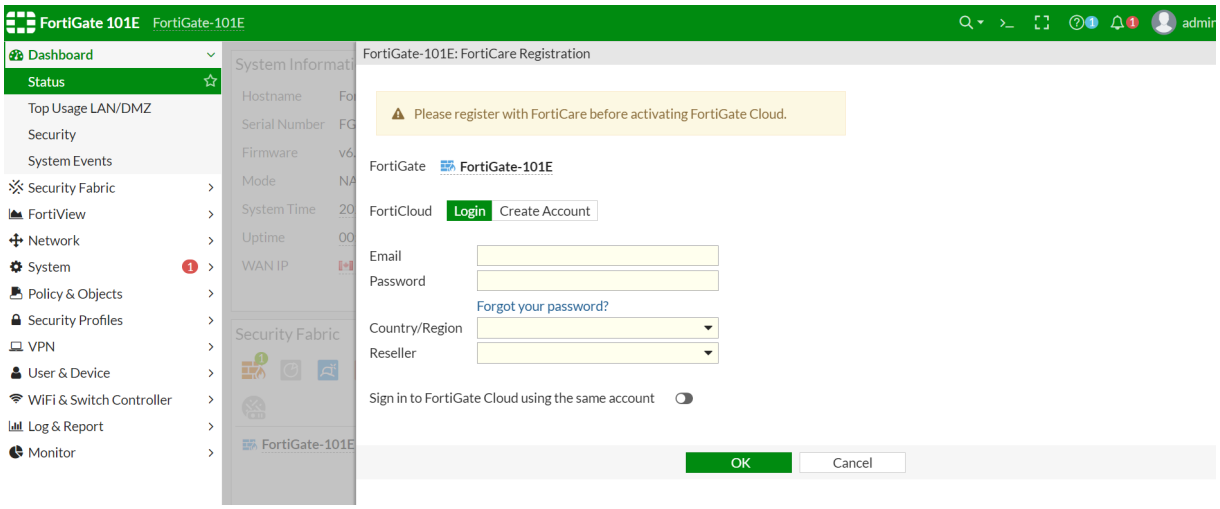
To activate FortiGate Cloud and register with FortiCloud at the same time:

1. Go to *Dashboard* > *Status*.
2. In the *FortiGate Cloud* widget, click *Not Activated* > *Activate*.

3. In the *FortiCloud* area, click *Login*.



You must register with FortiCare before activating FortiCloud.



4. Enter your FortiCloud account credentials, and click *OK*.

Security Fabric automation

This section includes Security Fabric automation features added to FortiOS:

- SDN connector for Cisco ACI northbound API integration on page 154
- Support multiple SDN connector instances for Cisco ACI and Nuage on page 157
- Automation stitches on page 163
- Slack notification action on page 172
- Simplify Azure Fabric connector configuration for a FortiGate-VM deployed on Azure on page 176
- Add multifunction tooltip for Fabric connectors on page 178
- Integrate FortiAnalyzer management into the Security Fabric using SAML SSO on page 180
- Group address objects synchronized from FortiManager on page 182
- Simplify FortiClient EMS setup on page 184
- Simplify the synchronization of EMS tags and configurations on page 187

- Allow FortiNAC to join the Security Fabric on page 189
- Exchange Server connector with Kerberos KDC auto-discovery on page 192
- Redesign Security Rating scorecards on page 193
- Redesign Fortinet Fabric Connectors and Fabric setup pages on page 195
- Display endpoints in Topology using donut chart on page 198
- Support filtering on AWS autoscaling group for dynamic address objects on page 200
- Support dynamic address objects in real servers under virtual server load balance on page 201
- Consolidate Monitor and FortiView pages on page 202
- Using the root FortiGate with disk to store historic user and device information on page 208
- Synchronizing objects across the Security Fabric on page 208

SDN connector for Cisco ACI northbound API integration

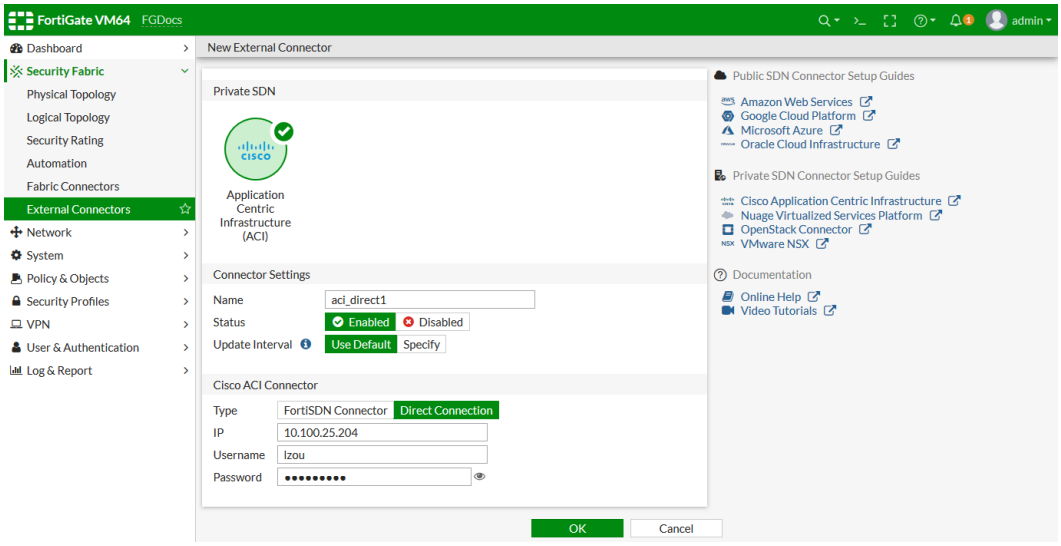
A new SDN connector type has been added for Cisco ACI (Application Centric Infrastructure) northbound API integration. Administrators can define a dynamic firewall addresses for Cisco ACI directly. Deploying an SDN connector through an external VM between the FortiGate and Cisco ACI is no longer required.

The following address filters are supported:

- Tenant
- Application
- Endpoint group
- Tag

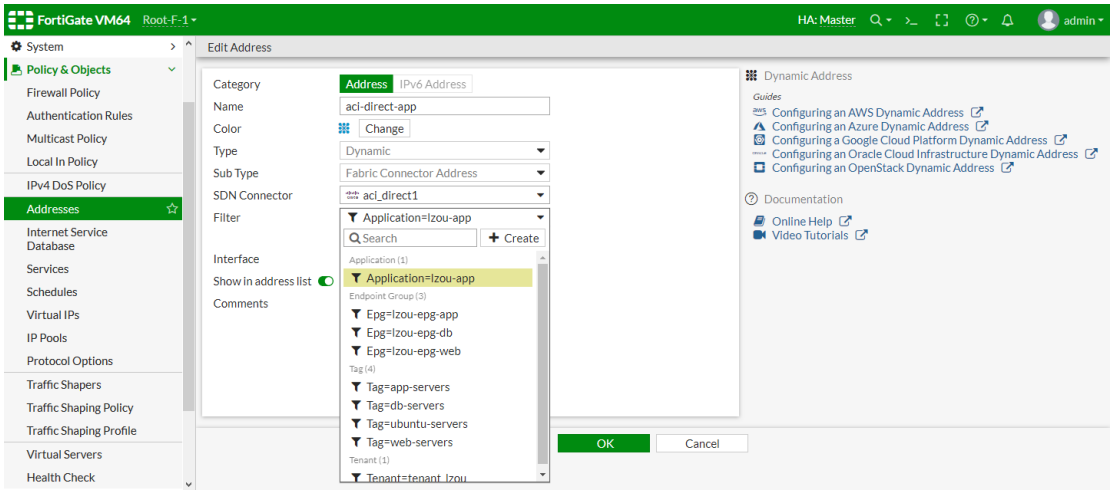
To configure a Cisco ACI connector in the GUI:

1. Configure the Cisco ACI SDN connector:
   a. Go to *Security Fabric* > *External Connectors*.
   b. Click *Create New*, and select *Application Centric Infrastructure (ACI)*.
   c. Configure the *Connector Settings* as needed. The update interval is in seconds.
   d. In the *Cisco ACI Connector* section, for *Type*, select *Direct Connection* and configure the remaining settings based on your deployment.



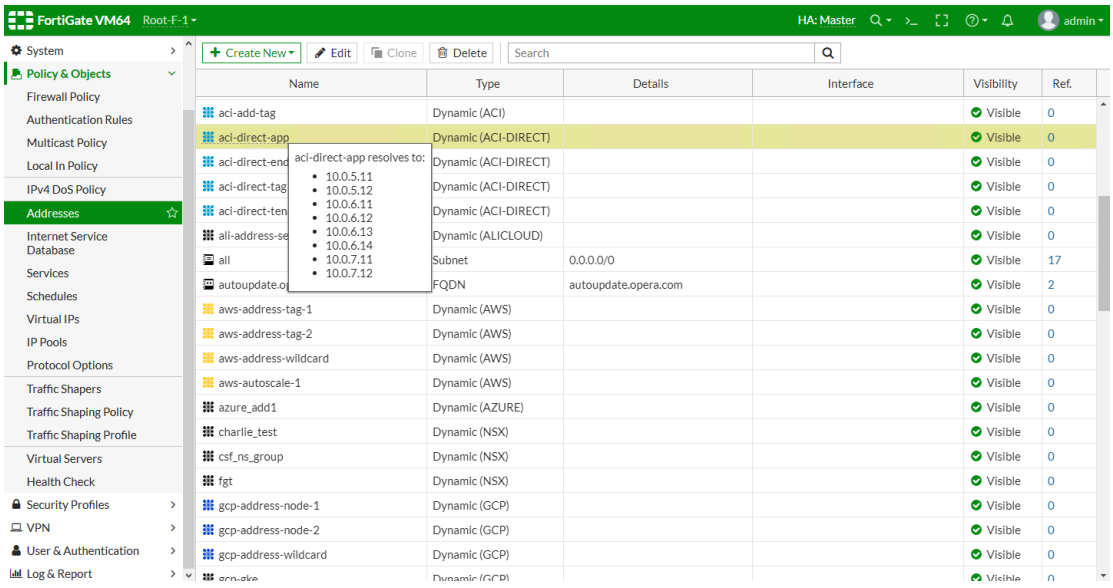
e. Click *OK*.

2. Create a dynamic firewall address for the connector:
   a. Go to *Policy & Objects* > *Addresses*.
   b. Click *Create New* > *Address* and enter a name.
   c. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the connector created in step 1.
      iv. For *Filter*, select an entry from the dropdown list. In this example, *Application* is selected.



   d. Click *OK*.

3. Confirm that the connector resolves the dynamic firewall IP addresses:
   a. Go to *Policy & Objects* > *Addresses*.
   b. In the address table, hover over the address created in step 2 to view which IPs it resolves to:



To configure a Cisco ACI connector in the CLI:

1. Configure the Cisco ACI SDN connector:

```
config system sdn-connector
    edit "aci_direct1"
        set status enable
        set type aci-direct
        set server "10.100.25.204"
        set username "lzou"
        set password xxxxxxx
        set update-interval 60
    next
end
```

2. Create a dynamic firewall address for the connector:

```
config firewall address
    edit "aci-direct-app"
        set type dynamic
        set sdn "aci_direct1"
        set color 17
        set filter "Application=lzou-app"
    next
end
```

3. Confirm that the connector resolves the dynamic firewall IP addresses:

```
config firewall address
    edit "aci-direct-app"
        show
            config firewall address
                edit "aci-direct-app"
                    set uuid 794aaf20-3e33-51ea-57e1-10b5badf3fc7
                    set type dynamic
                    set sdn "aci_direct1"
                    set color 17
                    set filter "Application=lzou-app"
                    config list
                        edit "10.0.5.11"
                        next
                        edit "10.0.5.12"
                        next
                        edit "10.0.6.11"
                        next
                        edit "10.0.6.12"
                        next
                        edit "10.0.6.13"
                        next
                        edit "10.0.6.14"
                        next
                        edit "10.0.7.11"
                        next
                        edit "10.0.7.12"
                        next
                    end
                next
            end
```

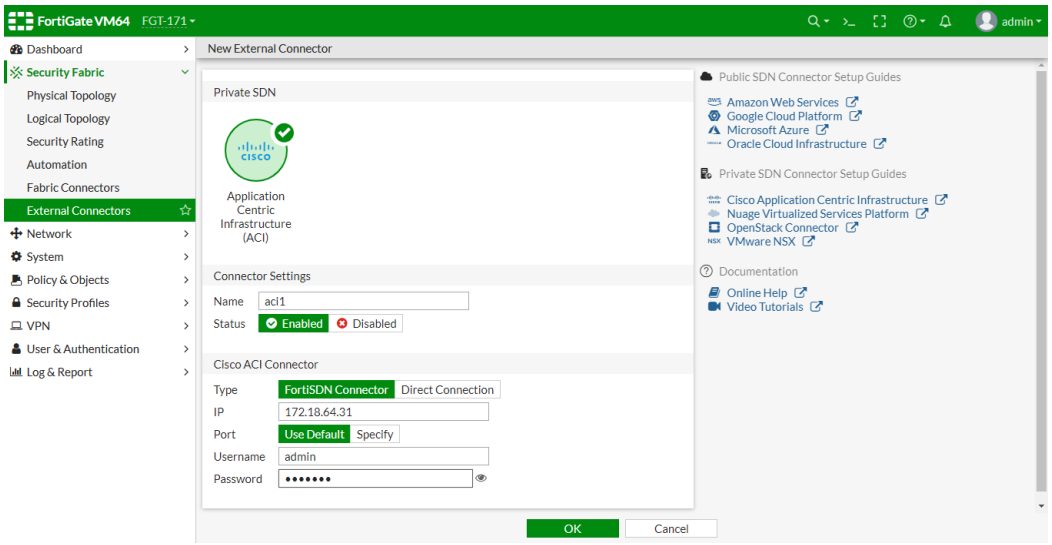
```
    next
end
```

Support multiple SDN connector instances for Cisco ACI and Nuage

Users can configure multiple Cisco ACI (Application Centric Infrastructure) and Nuage SDN connectors, which can be used in dynamic firewall addresses. The following examples configure two Cisco ACI and two Nuage SDN connectors.

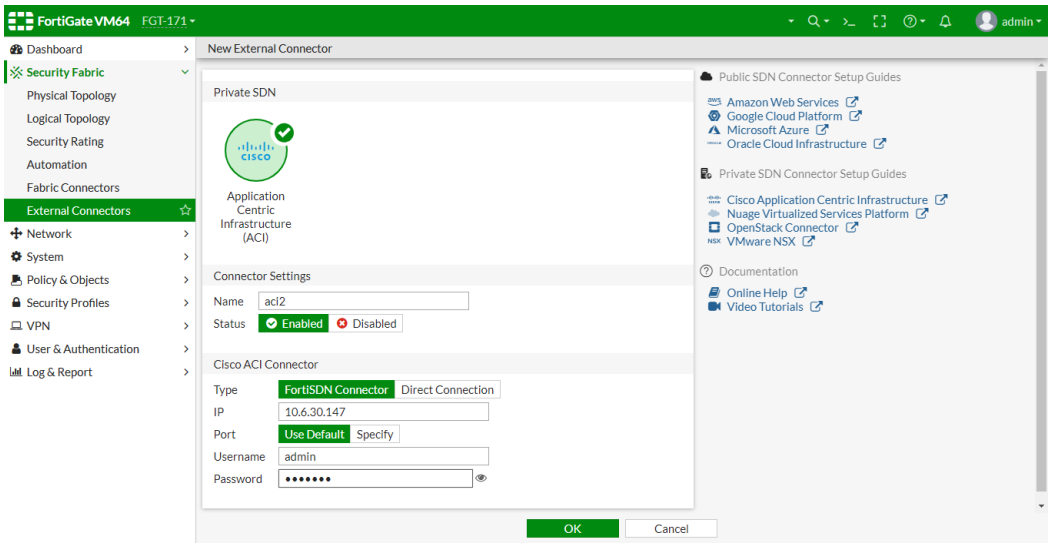
To configure Cisco ACI connectors in the GUI:

1. Configure the Cisco ACI SDN connectors:
   a. Go to *Security Fabric* > *External Connectors* and click *Create New*.
   b. In the *Private SDN* section, click *Application Centric Infrastructure (ACI)*.
   c. In the *Cisco ACI Connector* section, for *Type*, select *Fortinet SDN Connector* and configure the remaining settings as needed.

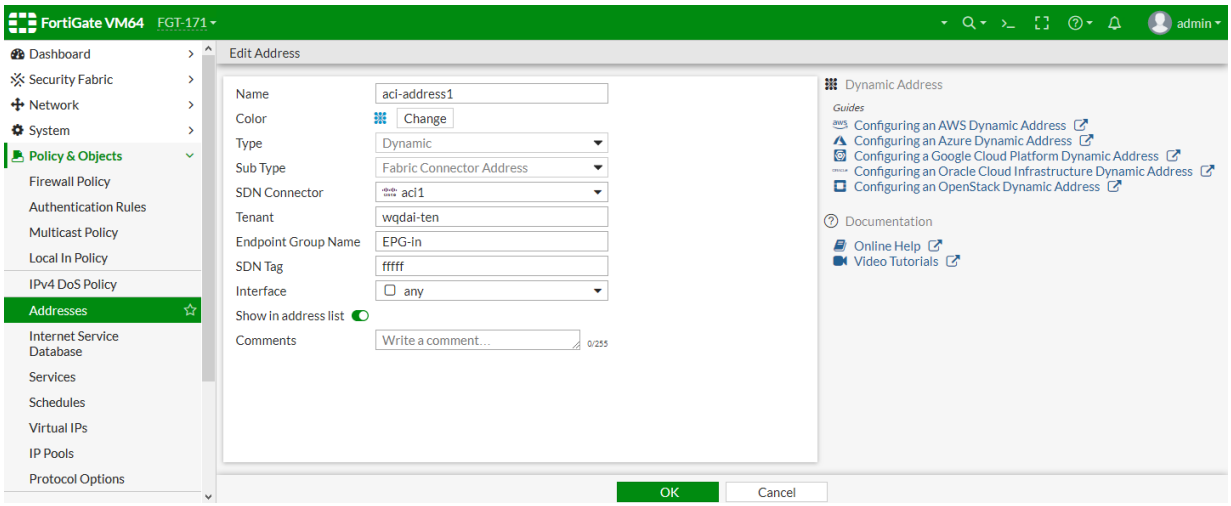


   d. Click *OK*.

e. Repeat these steps for the second connector.

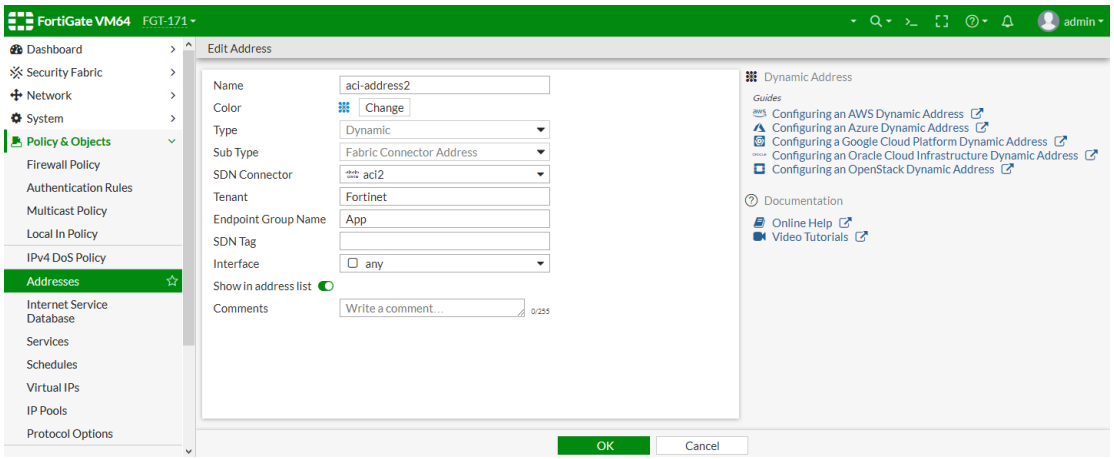


2. Create dynamic firewall addresses for the connectors:
   a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
   b. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the first ACI connector.
      iv. Configure the remaining settings as needed.



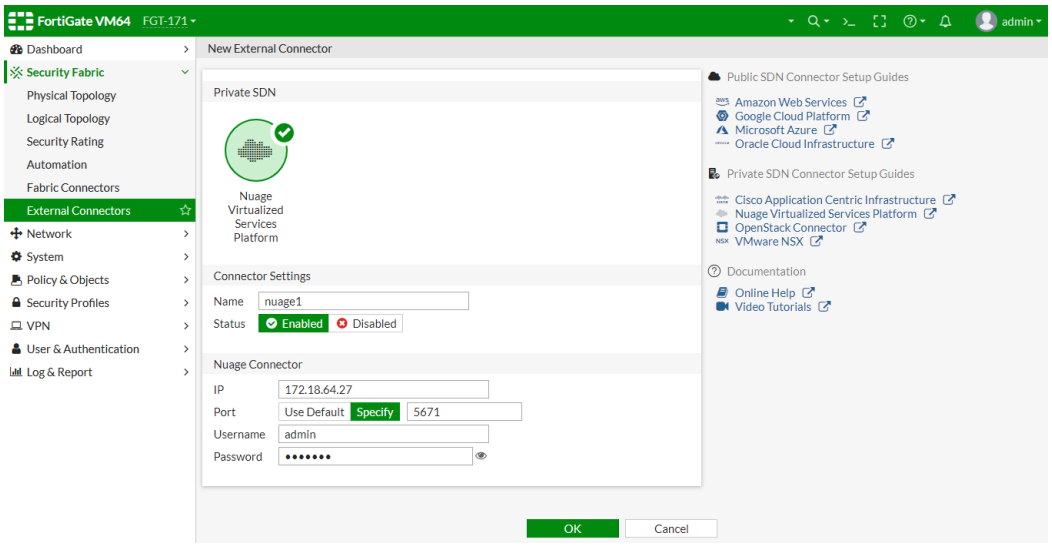
   c. Click *OK*.

d. Repeat these steps for the second connector.



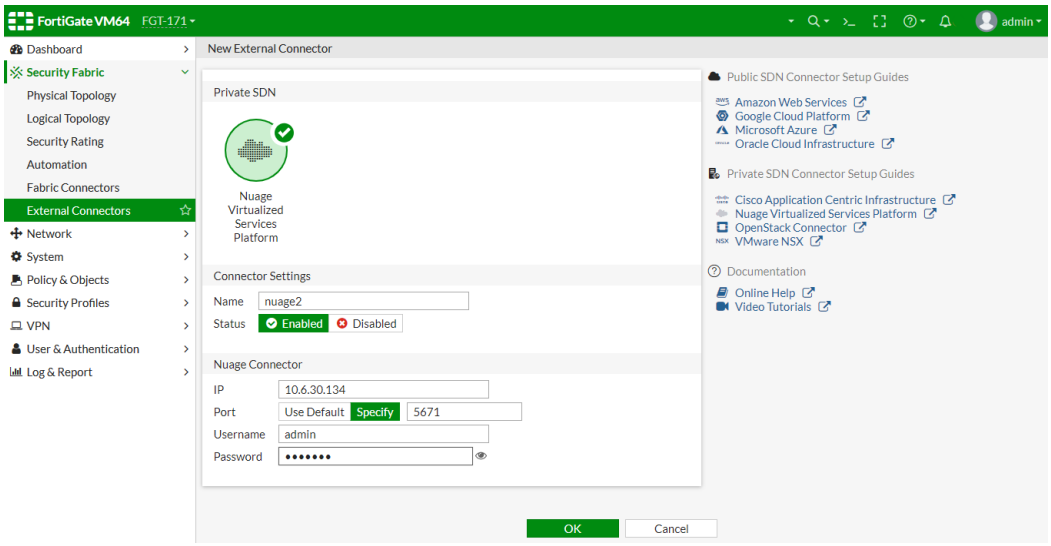
To configure Nuage connectors in the GUI:

1. Configure the Nuage SDN connectors:
   a. Go to *Security Fabric* > *External Connectors* and click *Create New*.
   b. In the *Private SDN* section, click *Nuage Virtualized Services Platform*.
   c. Configure the settings as needed.

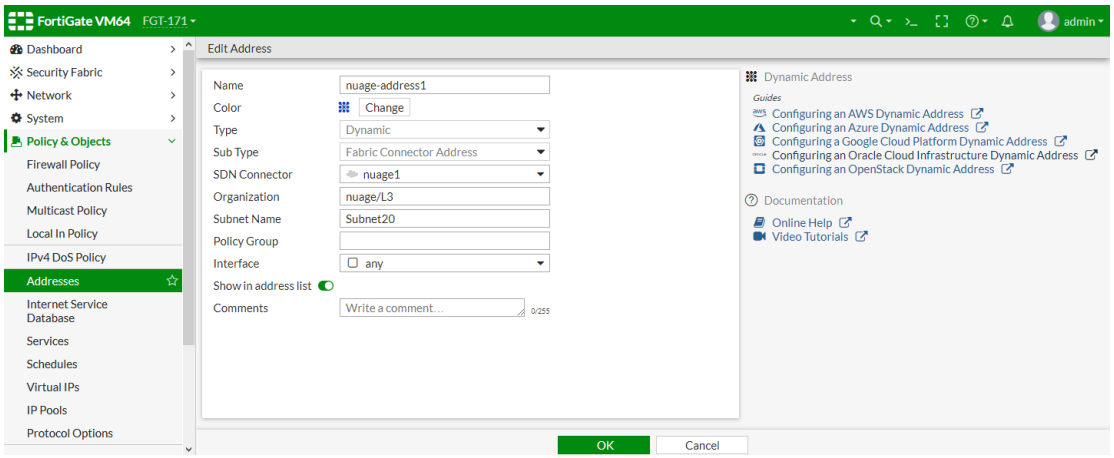


   d. Click *OK*.

e. Repeat these steps for the second connector.

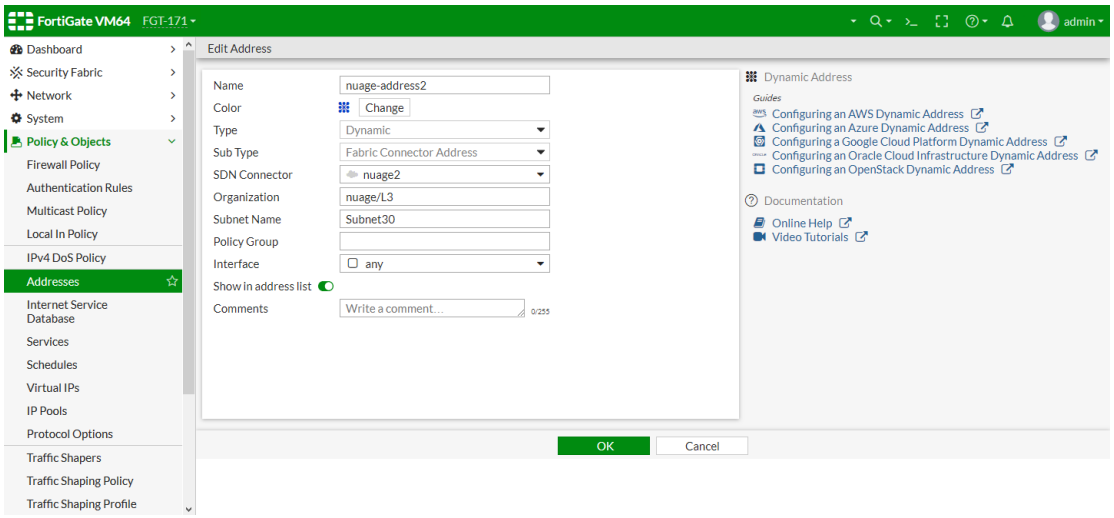


2. Create dynamic firewall addresses for the connectors:
   a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
   b. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the first the first Nuage connector.
      iv. Configure the remaining settings as needed.



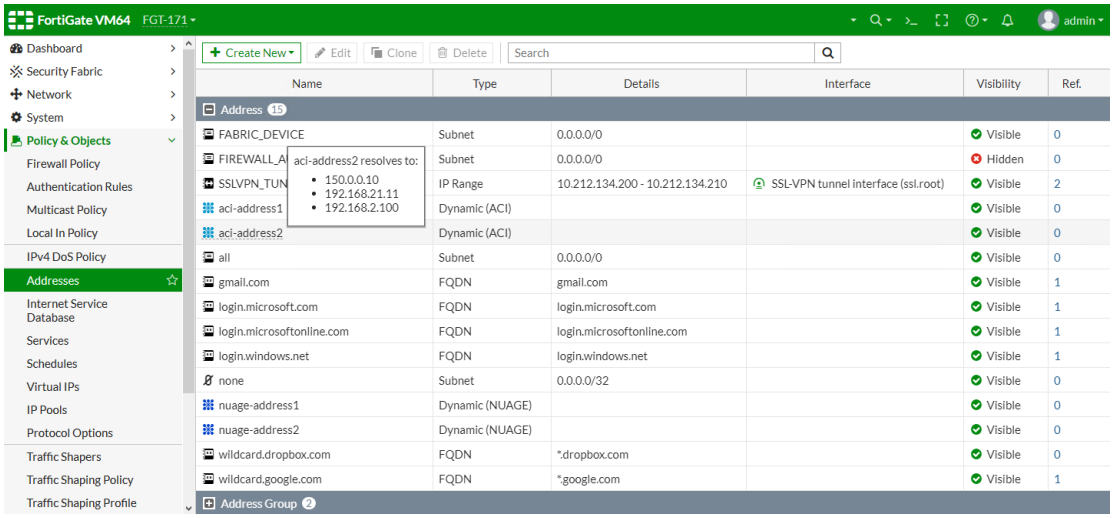
c. Click *OK*.

d. Repeat these steps for the second connector.



To verify the dynamic firewall IPs are resolved by the SDN connector in the GUI:

1. Go to *Policy & Objects* > *Addresses*.
2. In the address table, hover over an address to view which IPs it resolves to:



To configure Cisco ACI connectors in the CLI:

1. Configure the SDN connectors:

```
config system sdn-connector
    edit "aci1"
        set type aci
        set server "172.18.64.31"
        set username "admin"
        set password xxxxxxx
    next
    edit "aci2"
        set type aci
        set server "10.6.30.147"
```

```
        set username "admin"
        set password xxxxxxx
    next
end
```

2. Create dynamic firewall addresses for the connectors:

```
config firewall address
    edit "aci-address1"
        set type dynamic
        set sdn "aci1"
        set color 17
        set tenant "wqdai-ten"
        set epg-name "EPG-in"
        set sdn-tag "fffff"
    next
    edit "aci-address2"
        set type dynamic
        set sdn "aci2"
        set color 17
        set tenant "Fortinet"
        set epg-name "App"
    next
end
```

To configure Nuage connectors in the CLI:

1. Configure the SDN connectors:

```
config system sdn-connector
    edit "nuage1"
        set type nuage
        set server "172.18.64.27"
        set server-port 5671
        set username "admin"
        set password xxxxxxx
    next
    edit "nuage2"
        set type nuage
        set server "10.6.30.134"
        set server-port 5671
        set username "admin"
        set password xxxxxxx
    next
end
```

2. Create dynamic firewall addresses for the connectors:

```
config firewall address
    edit "nuage-address1"
        set type dynamic
        set sdn "nuage1"
        set color 19
        set organization "nuage/L3"
        set subnet-name "Subnet20"
    next
    edit "nuage-address2"
        set type dynamic
```

```
            set sdn "nuage2"
            set color 19
            set organization "nuage/L3"
            set subnet-name "Subnet30"
        next
    end
```

To verify the dynamic firewall IPs are resolved by the SDN connector in the CLI:

```
# diagnose firewall dynamic list

List all dynamic addresses:
aci1.aci.wqdai-ten.EPG-in.fffff: ID(171)
        ADDR(192.168.100.20)

nuage1.nuage.nuage/L3.Subnet20.*: ID(196)
        ADDR(192.168.20.92)
        ADDR(192.168.20.240)

nuage2.nuage.nuage/L3.Subnet30.*: ID(198)
        ADDR(192.168.30.92)

aci2.aci.Fortinet.App.*: ID(218)
        ADDR(150.0.0.10)
        ADDR(192.168.21.11)
        ADDR(192.168.2.100)
```

Automation stitches

Eight new webhook automation stitches were added to the *Automation* menu. The additional stitches include a new *Incoming Webhook Quarantine* trigger for API calls to the FortiGate, as well as a predefined *License Expired Notification* that replaces the existing license expiry alerts.

The automation stitches are available in new FortiGate installations by default. To install the stitches on an existing device, perform a factory reset.



Performing a factory reset will wipe the existing configurations from the ForttGate.

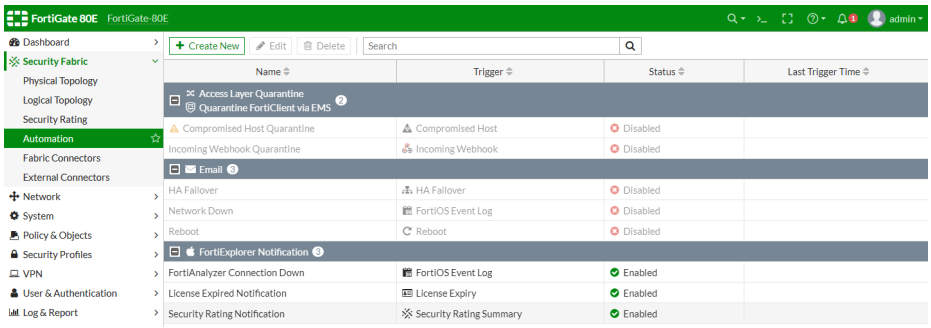
Before performing a factory reset, backup the existing configuration. Contact Fortinet support for additional assistance.

The following webhook stitches were added to the *Automation* menu:

- Compromised Host Quarantine
- Incoming Webhook quarantine
- HA Failover
- Network Down
- Reboot
- FortiAnalyzer Connection Down
- License Expired Notification
- Security rating Notification

GUI

To view the new automation stitches in the GUI, go to *Security Fabric* > *Automation*.



After the factory reset, the email alert feature will be removed from the GUI (*Log & Report* > *Email Alert Settings*), and replaced with the *Email* automation stitches.

You can continue using the email alert feature with the CLI console.

CLI

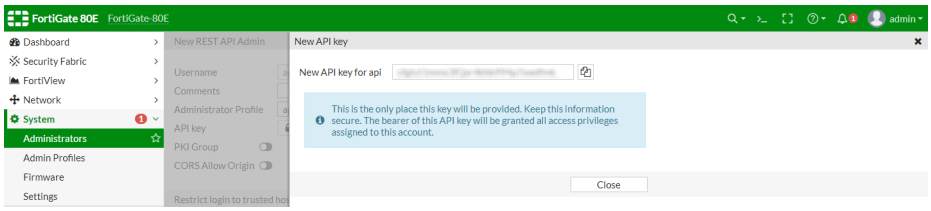
To configure the new automation stitches in the CLI console, use the following commands:

```
config system automation-action
config system automation-trigger
config system automation-stitch
```

To view the configurations for the new automation stitches, see the CLI reference at the bottom of the page.

To trigger an Incoming Webhook Quarantine stitch with the GUI:

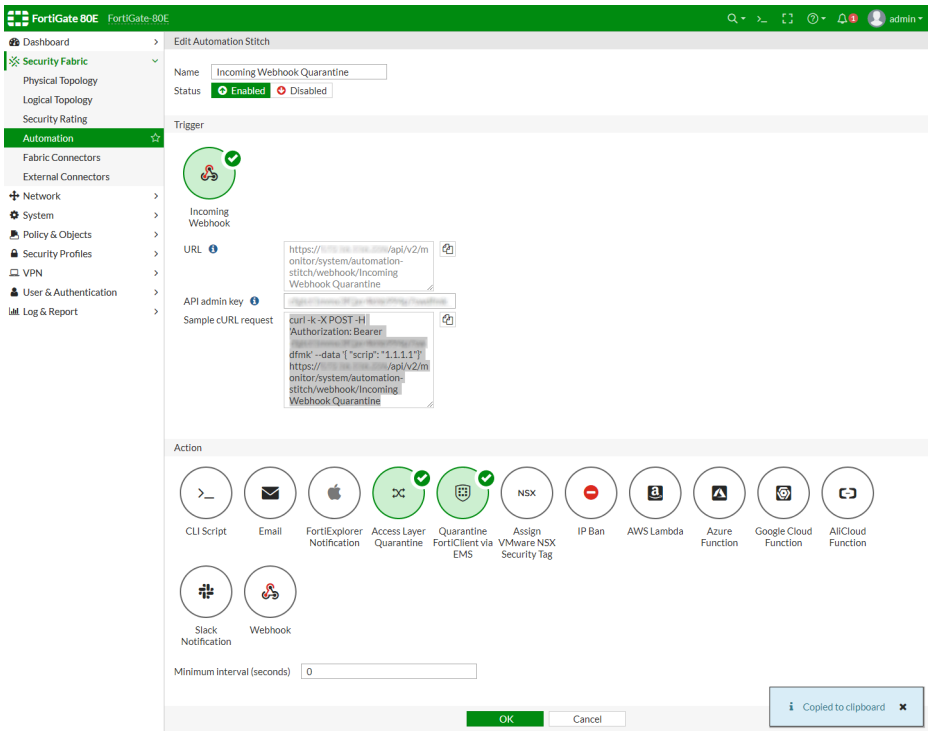
1. Create new API user.
   a. Go to *System* > *Administrators*.
   b. Click *Create New* > *REST API Admin*.
   c. Configure the *New REST API Admin* settings, and record the API key.



2. Get the sample cURL request.
   a. Go to *Security Fabric* > *Automation*.
   b. Under *Incoming Webhook*, right-click *Incoming Webhook Quarantine*, and select *Edit*.
   c. Click *Enabled*, to enable the rule.

d. In the *API admin key* field, enter the API key you recorded in the previous step. A *Sample cURL request* is created.

e. Copy the *Sample cURL request*.



3. Execute the request:

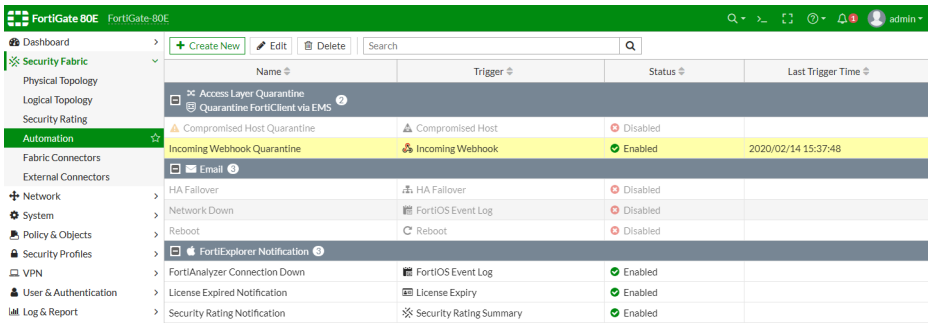
a. Edit the sample cURL you recorded in the previous step.

b. Add parameters to the `data` field (`"mac"` and *"fctuid"*), and then execute the request.

```
root@pc:~# curl -k -X POST -H 'Authorization: Bearer
cfgtct1mmx3fQxr4khb994p7swdfmk' --data '{ "mac":"0c:0a:00:0c:ce:b0", "fctuid":
"0000BB0B0ABD0D00B0D0A0B0E0F0B00B"}'
https://172.16.116.226/api/v2/monitor/system/automation-
stitch/webhook/Incoming%20Webhook%20Quarantine
{
   "http_method":"POST",
   "status":"success",
   "http_status":200,
   "serial":"FGT00E0Q00000000",
   "version":"v6.4.0",
   "build":1545
```

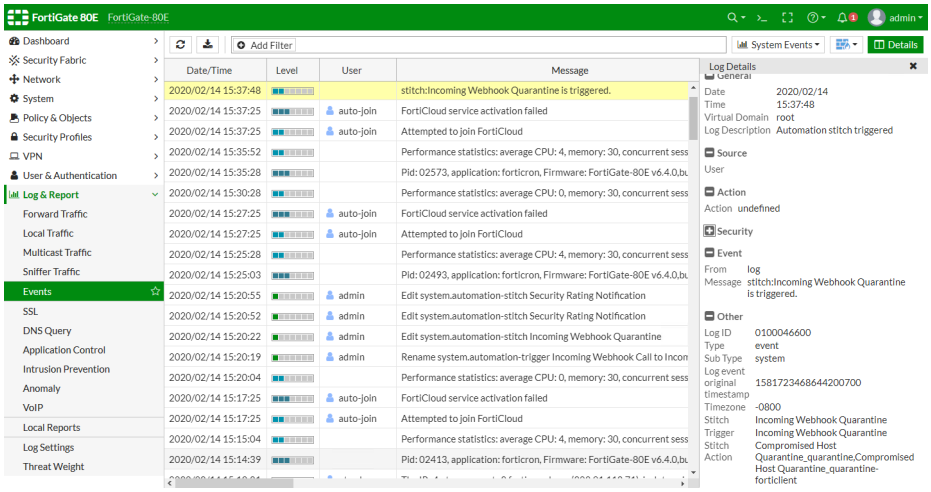
Encode the spaces in the automation-stitch name with `%20`. For example, `Incoming%20Webhook%20Quarantine`

The automation rule *Incoming Webhook Quarantine* is triggered.



The MAC address is quarantined in FortiGate and an event log is created.

The FortiClient UUID is quarantined by EMS on the server side.



To create an automated stitch with the CLI:

1. Create new API user and record the API key.

```
config system api-user
   edit "api"
      set api-key ENC SH00vqP0GKWKyZNz0FP0/jq00O0Ka/DHVEKdxUi+0kRDNKPpZppnnMk0KeunBI=
      set accprofile "api_profile"
      set vdom "root"
      config trusthost
         edit 1
            set ipv4-trusthost 10.6.30.0 200.200.200.0
         next
      end
   next
end
```

2. Configure the automation stitch, `Incoming Webhook Quarantine`.

```
config system automation-stitch
   edit "Incoming Webhook Quarantine"
      set status enable
      set trigger "Incoming Webhook Quarantine"
      set action "Compromised Host Quarantine_quarantine" "Compromised Host Quarantine_
          quarantine-forticlient"
   next
end
```

3. Add parameters in the `data` field (`"mac"` and `"fctuid"`), then execute the request on a device.

```
root@pc56:~# curl -k -X POST -H 'Authorization: Bearer
cfgtct1mmx0fQxr4khb000p70wdfmk' --data '{ "mac":"0c:0a:00:0c:ce:b0", "fctuid":
"3000BB0B0ABD0D00B0D0A0B0E0F0B00B"}'
https://100.10.100.200/api/v2/monitor/system/automation-
stitch/webhook/Incoming%20Webhook%20Quarantine
{
   "http_method":"POST",
   "status":"success",
   "http_status":200,
   "serial":"FGT80E0Q00000000",
   "version":"v6.4.0",
   "build":1545
```



Encode the spaces in the automation-stitch name with `%20`. For example, `Incoming%20Webhook%20Quarantine`

The automation rule "Incoming Webhook Quarantine" is triggered. The MAC address is quarantined in FortiGate, and an event log is created. The FortiClient UUID will be quarantined on the EMS server side.

```
config user quarantine
   config targets
      edit "0c:0a:00:0c:ce:b0"
         config macs
            edit 0c:0a:00:0c:ce:b0
               set description "Quarantined by automation stitch: Incoming Webhook
                    Quarantine"
            next
         end
      next
   end
end
date=2020-02-14 time=15:37:48 logid="0100046600" type="event" subtype="system"
level="notice" vd="root" eventtime=1581723468644200712 tz="-0800"
logdesc="Automation stitch triggered" stitch="Incoming Webhook Quarantine"
trigger="Incoming Webhook Quarantine" stitchaction="Compromised Host
Quarantine_quarantine,Compromised Host Quarantine_quarantine-forticlient"
from="log" msg="stitch:Incoming Webhook Quarantine is triggered."
```

CLI Reference

Network down

config system automation-action

```
config system automation-action
   edit "Network Down_email"
      set action-type email
      set email-from ''
      set email-subject "Network Down"
      set minimum-interval 0
      set delay 0
```

```
      set required disable
      set message "%%log%%"
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "Network Down"
      set trigger-type event-based
      set event-type event-log
      set logid 20099
      config fields
         edit 1
            set name "status"
            set value "DOWN"
         next
      end
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "Network Down"
      set status disable
      set trigger "Network Down"
      set action "Network Down_email"
   next
end
```

HA failover

config system automation-action

```
config system automation-action
   edit "HA Failover_email"
      set action-type email
      set email-from ''
      set email-subject "HA Failover"
      set minimum-interval 0
      set delay 0
      set required disable
      set message "%%log%%"
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "HA Failover"
      set trigger-type event-based
      set event-type ha-failover
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "HA Failover"
      set status disable
      set trigger "HA Failover"
      set action "HA Failover_email"
   next
end
```

Reboot

config system automation-action

```
config system automation-action
   edit "Reboot_email"
      set action-type email
      set email-from ''
      set email-subject "Reboot"
      set minimum-interval 0
      set delay 0
      set required disable
      set message "%%log%%"
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "Reboot"
      set trigger-type event-based
      set event-type reboot
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "Reboot"
      set status disable
      set trigger "Reboot"
      set action "Reboot_email"
   next
end
```

Connection down

config system automation-action

```
config system automation-action
   edit "FortiAnalyzer Connection Down_ios-notification"
      set action-type ios-notification
      set minimum-interval 0
      set delay 0
      set required disable
```

```
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "FortiAnalyzer Connection Down"
      set trigger-type event-based
      set event-type event-log
      set logid 22902
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "FortiAnalyzer Connection Down"
      set status enable
      set trigger "FortiAnalyzer Connection Down"
      set action "FortiAnalyzer Connection Down_ios-notification"
   next
end
```

License expired

config system automation-action

```
config system automation-action
   edit "License Expired Notification_ios-notification"
      set action-type ios-notification
      set minimum-interval 0
      set delay 0
      set required disable
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "License Expired Notification"
      set trigger-type event-based
      set event-type license-near-expiry
      set license-type any
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "License Expired Notification"
      set status enable
      set trigger "License Expired Notification"
      set action "License Expired Notification_ios-notification"
   next
end
```

Compromised host

config system automation-action

```
config system automation-action
   edit "Compromised Host Quarantine_quarantine"
      set action-type quarantine
      set minimum-interval 0
      set delay 0
      set required disable
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "Compromised Host Quarantine"
      set trigger-type event-based
      set event-type ioc
      set ioc-level high
   next
end
```

config system automation-stitch

```
config system automation-stitch
   edit "Compromised Host Quarantine"
      set status disable
      set trigger "Compromised Host Quarantine"
      set action "Compromised Host Quarantine_quarantine" "Compromised Host Quarantine_
           quarantine-forticlient"
   next
end
```

Quarantine FortiClient

config system automation-action

```
config system automation-action
   edit "Compromised Host Quarantine_quarantine-forticlient"
      set action-type quarantine-forticlient
      set minimum-interval 0
      set delay 0
      set required disable
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "Compromised Host Quarantine"
      set trigger-type event-based
      set event-type ioc
      set ioc-level high
   next
```

```
end
```

config system automation-stitch

```
config system automation-stitch
   edit "Compromised Host Quarantine"
      set status disable
      set trigger "Compromised Host Quarantine"
      set action "Compromised Host Quarantine_quarantine" "Compromised Host Quarantine_
           quarantine-forticlient"
   next
end
```

Security rating

config system automation-action

```
config system automation-action
   edit "Security Rating Notification_ios-notification"
      set action-type ios-notification
      set minimum-interval 0
      set delay 0
      set required disable
   next
end
```

config system automation-trigger

```
config system automation-trigger
   edit "Security Rating Notification"
      set trigger-type event-based
      set event-type security-rating-summary
   next
end
```

config system automation-stitch

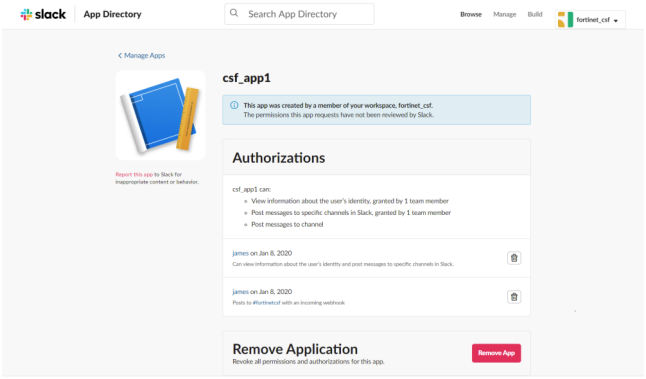
```
config system automation-stitch
   edit "Security Rating Notification"
      set status enable
      set trigger "Security Rating Notification"
      set action "Security Rating Notification_ios-notification"
   next
end
```

Slack notification action

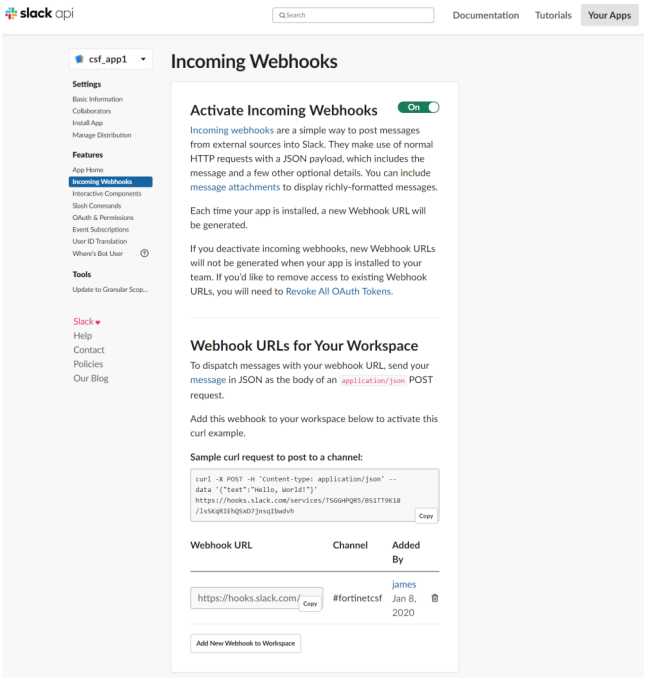
Automation stitches now include a Slack notification in the *Action* menu. To configure the automation stitch, create an Incoming Webhook in Slack, and then enter the Webhook URL in the corresponding field of the notification action in FortiGate.

To create an Incoming Webhook in Slack:

1. Go to the Slack website, and create a workspace.
2. Create a Slack application for the workspace.



3. Add an *Incoming Webhook* to a channel in the workspace.
4. Activate the *Incoming Webhook*, and record the *Webhook URL*.

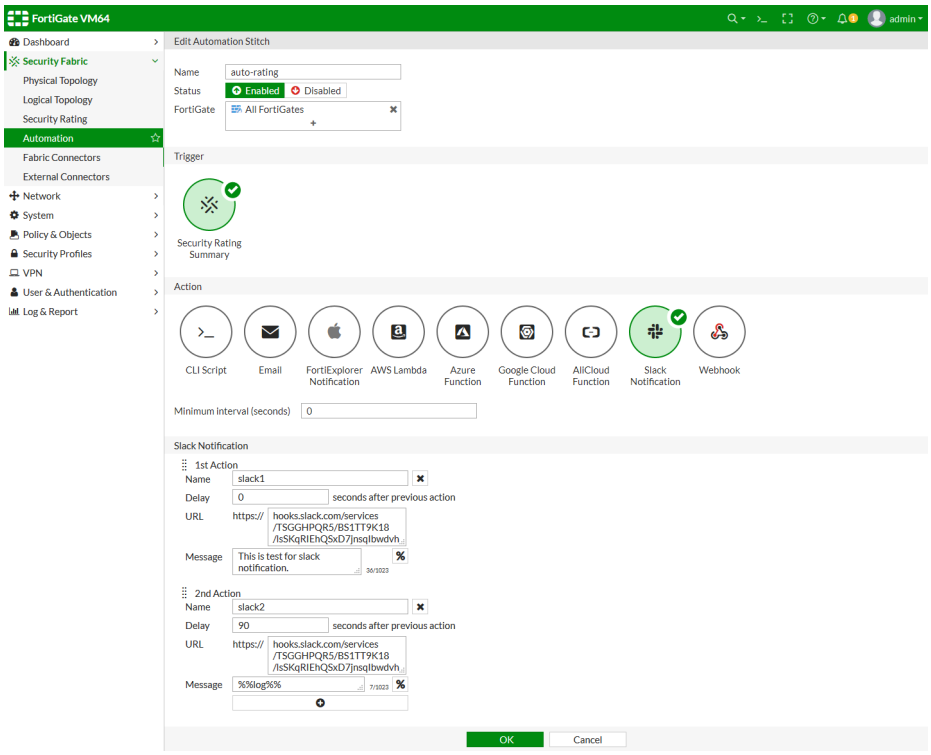


For information about using Incoming Webhooks in Slack, see https://api.slack.com/incoming-webhooks.

To configure a Slack notification in the GUI:

1. Go to *Security Fabric* > *Automation*.
2. Choose an automation stitch, and click *Edit*.

3. Select *Slack Notification*, and configure the notification settings.



Name	Enter a name for the notification.
Delay	Enter the number of seconds to delay the notification after the previous action is triggered.
URL	Enter the *Webhook URL* you recorded when you created the Incoming Webhook in Slack.
Message	Take one of the following actions: • Configure the message parameters. Click **%** to view a description of the available parameters. • Enter the message to display in the Slack channel.

4. (Optional) Click the plus (+) sign to add another action.
5. Click *OK*.
6. Run the automation stitch to trigger the action.

To configure a Slack notification in the CLI:

1. Add the webhook URL the Slack notification action.

```
config system automation-action
   edit "slack1"
      set action-type slack-notification
      set minimum-interval 0
      set delay 0
      set required disable
      set message "This is test for slack notification."
      set uri "hooks.slack.com/services/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"
```

```
    next
    edit "slack2"
       set action-type slack-notification
       set minimum-interval 0
       set delay 90
       set required disable
       set message "%%log%%"
       set uri "hooks.slack.com/services/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"
    next
end
```

2. Create the trigger for the notification.

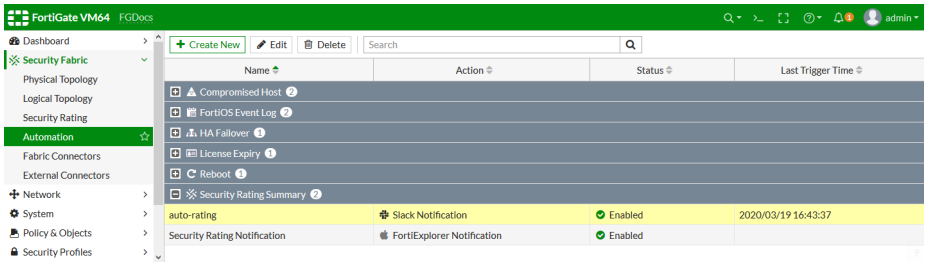
```
config system automation-trigger
   edit "auto-rating"
      set trigger-type event-based
      set event-type security-rating-summary
   next
end
```

3. Configure the action for the trigger.

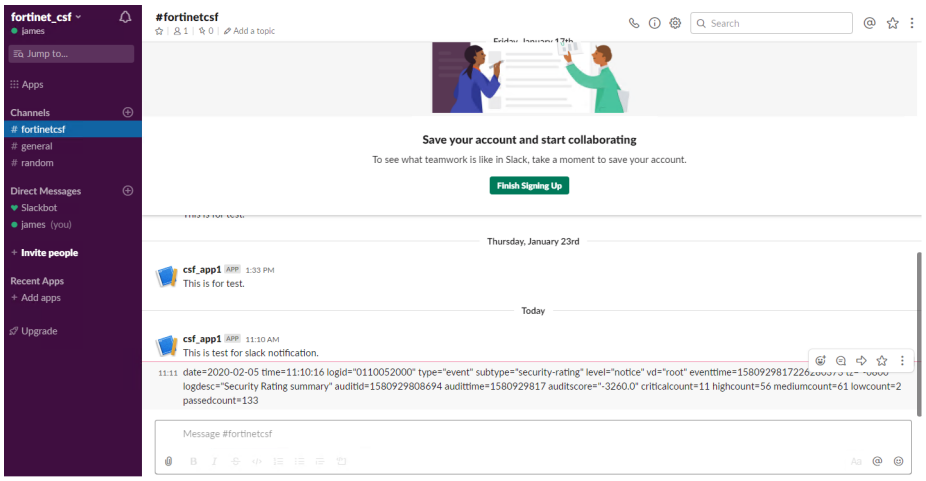
```
config system automation-stitch
   edit "auto-rating"
      set status enable
      set trigger "auto-rating"
      set action "slack1" "slack2"
   next
end
```

4. Trigger the notification.

The notification action is triggered in FortiGate.



The message you entered in the automation stitch is delivered to the Slack channel.



Simplify Azure Fabric connector configuration for a FortiGate-VM deployed on Azure

For a FortiGate-VM deployed on Azure, the new *Use managed identity* setting allows FortiOS to connect to Azure based on the FortiGate-VM's user-assigned managed identity. Using user-assigned managed identities enables a FortiGate-VM deployed on Azure to authenticate to cloud services without storing credentials in FortiOS.

When you enable *Use managed identity* for an Azure Fabric connector, you do not need to configure the *Tenant ID*, *Client ID*, and *Client secret* fields on the Fabric connector creation page. FortiOS hides these fields when you enable *Use managed identity* for an Azure Fabric connector.

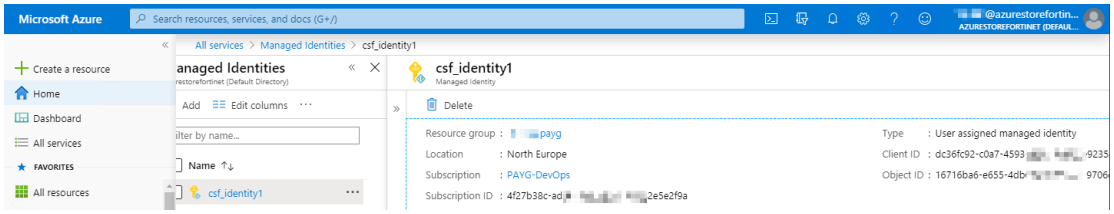
This feature only applies for a FortiGate-VM deployed on Azure. For a FortiGate that is not deployed on Azure, you must still configure the *Tenant ID*, *Client ID*, and *Client secret* fields for an Azure Fabric connector. This feature also does not apply for a FortiGate-VM deployed on Azure Stack.

This configuration consists of the following steps:

1. Configure a user-managed identity in Azure.
2. Configure an Azure Fabric connector in FortiOS:
   a. GUI instructions
   b. CLI instructions

To configure a user-managed identity in Azure:

1. In Azure, go to *All services* > *Managed Identities*. Create a managed identity.



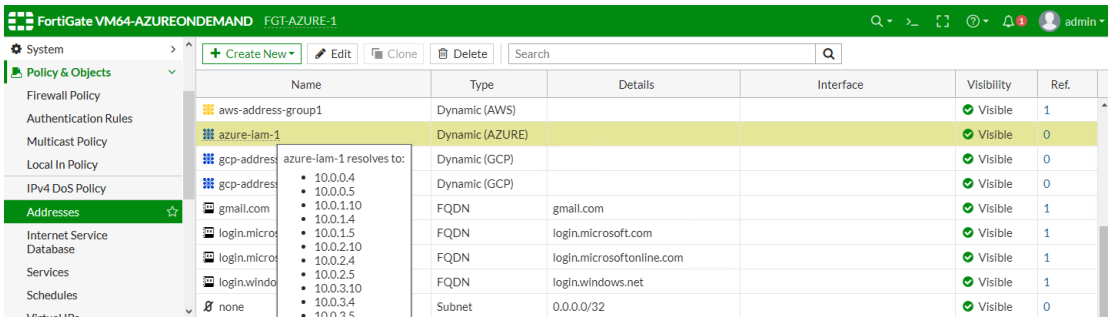
2. Go to the FortiGate-VM instance, then go to *Identity*. Set the managed identity created in step a as the user-assigned identity.
3. Search for subscriptions to assign the level of scope. Select the subscription, then go to *Access control (IAM)*. Click *Add role assignment*. From the *Role* dropdown list, select *Contributor*.

To configure an Azure Fabric connector in the FortiOS GUI:

1. Configure the Fabric connector in FortiOS:
   a. On the FortiGate-VM deployed on Azure, go to *Security Fabric* > *External Connectors*.
   b. Click *Create New*.
   c. Under *Public SDN*, select *Microsoft Azure*.
   d. Enable *Use managed identity*.
   e. Configure other settings as desired.
   f. Click *OK*.
2. Create a dynamic firewall address associated to the Fabric connector:
   a. Go to *Policy & Objects* > *Addresses*.
   b. From the *Type* dropdown list, select *Dynamic*.
   c. From the *Sub Type* dropdown list, select *Fabric Connector Address*.

d. From the *SDN Connector* dropdown list, select the Fabric connector that you created in step 1.
e. Configure other settings as desired.
f. Click *OK*.

3. To confirm that the Fabric connector resolves the dynamic firewall IP addresses with the supported filter, go to *Policy & Objects* > *Addresses*. Hover over the address that you created in step 2.



To configure an Azure Fabric connector in the FortiOS CLI:

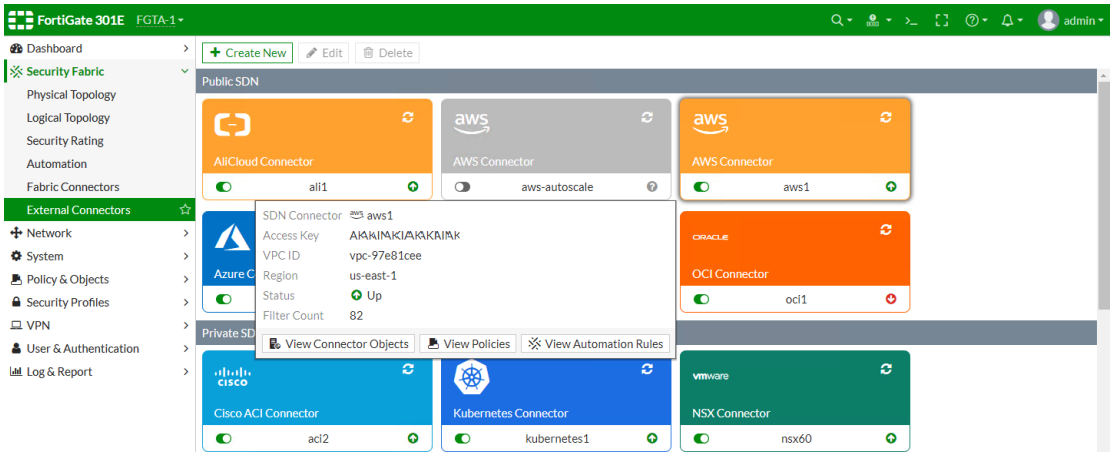
1. Configure the Fabric connector in FortiOS:
```
config system sdn-connector
   edit "azure"
      set status enable
      set type azure
      set azure-region global
      set use-metadata-iam enable
   next
end
```
2. Create a dynamic firewall address associated to the Fabric connector:
```
config firewall address
   edit "azure-iam-1"
      set type dynamic
      set sdn "azure"
      set color 2
      set filter "ResourceGroup=azuretest"
   next
end
```
3. Confirm that the Fabric connector resolves the dynamic firewall IP addresses with the supported filter:
```
config firewall address
   edit "azure-iam-1"
      set type dynamic
      set sdn "azure2"
      set color 2
      set filter "ResourceGroup=azuretest"
      config list
         edit "10.0.0.4"
         next
         edit "10.0.0.5"
         next
         edit "10.0.1.10"
         next
         edit "10.0.1.4"
         next
         edit "10.0.1.5"
         next
```

```
            edit "10.0.2.10"
            next
            edit "10.0.2.4"
            next
            edit "10.0.2.5"
            next
            edit "10.0.3.10"
            next
            edit "10.0.3.4"
            next
            edit "10.0.3.5"
            next
            edit "10.5.0.4"
            next
            edit "10.5.0.5"
            next
            edit "10.8.0.5"
            next
            edit "10.8.1.6"
            next
        end
    next
end
```

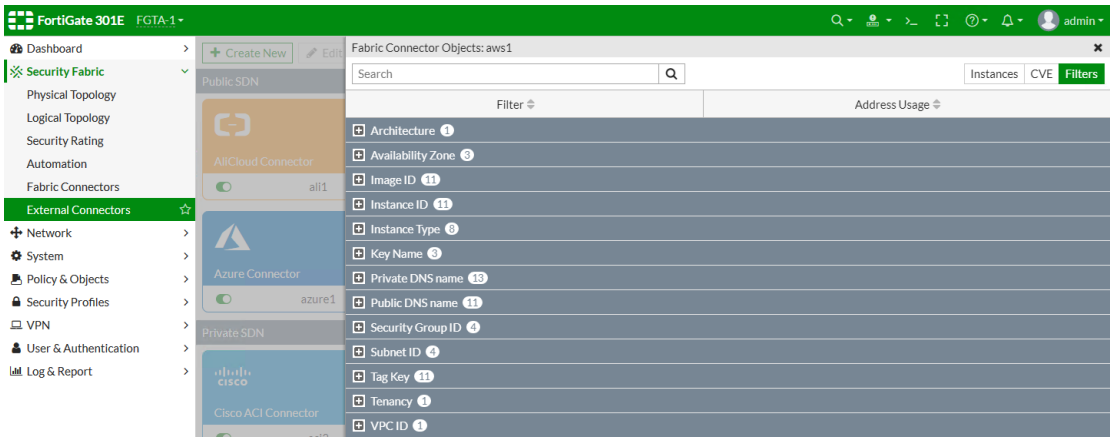
Add multifunction tooltip for Fabric connectors

FortiOS 6.4.0 adds tooltips to Fabric connector cards and other areas that reference the Fabric connector. Tooltips provide information on the connector, associated actions, and policies and objects defined against the connectors, driven primarily from tooltips throughout FortiOS.

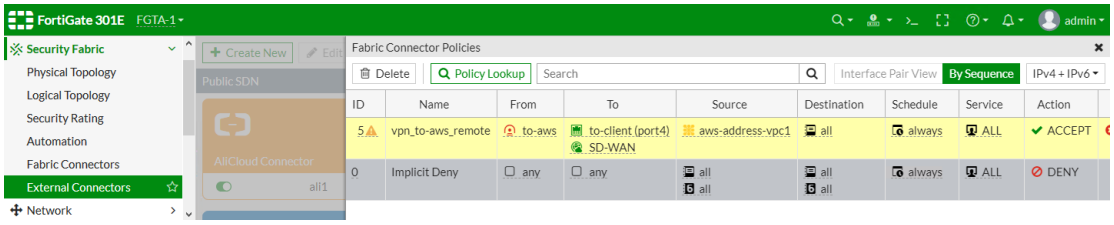
In *Security Fabric > External Connectors*, when you hover over a Fabric connector, a tooltip appears that shows basic information on its configuration.



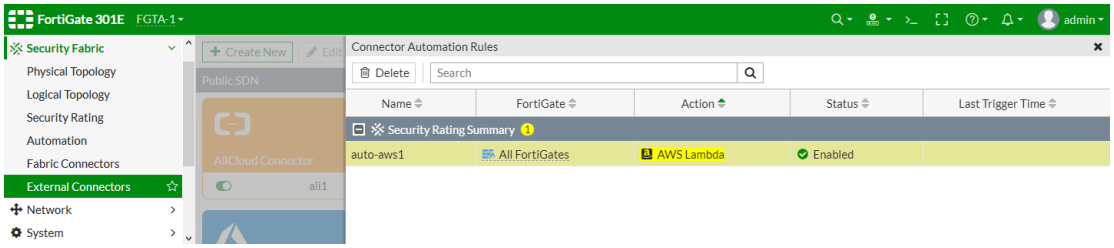
If you click *View Connector Objects* from this tooltip, the *Fabric Connector Objects* pane shows this Fabric connector's dynamic objects, such as filters. For an AWS Fabric connector, this pane also shows instance and CVE information.



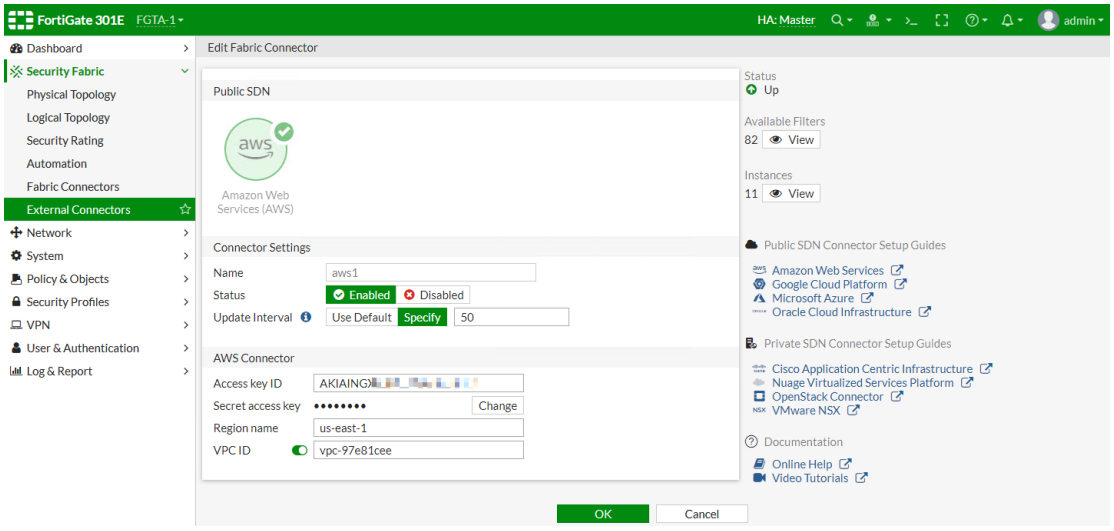
If you click *View Policies* from the tooltip, the *Fabric Connector Policies* pane shows policies that are using dynamic addresses from this Fabric connector.



If you click *View Automation Rules* from the tooltip, the *Connector Automation Rules* pane shows automation actions that are using this Fabric connector.



When you edit an existing Fabric connector, the connector status, filter, and instance information displays.



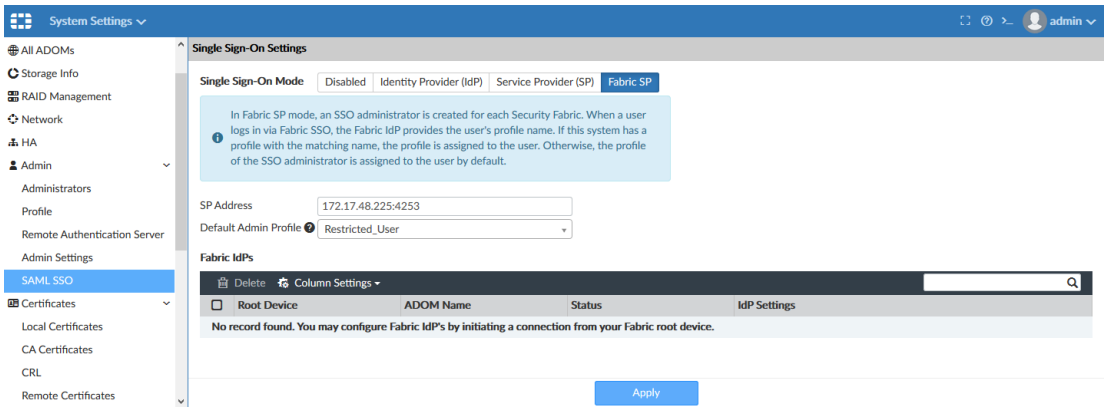
Integrate FortiAnalyzer management into the Security Fabric using SAML SSO

When a FortiGate is configured as the SAML SSO IdP, FortiAnalyzer can register itself as the SP (FortiAnalyzer must be running version 6.4.0). Once registered, FortiAnalyzer will be added automatically to the Security Fabric navigation in FortiOS. A similar dropdown navigation is displayed in FortiAnalyzer where users can navigate to the FortiGate using SAML SSO.

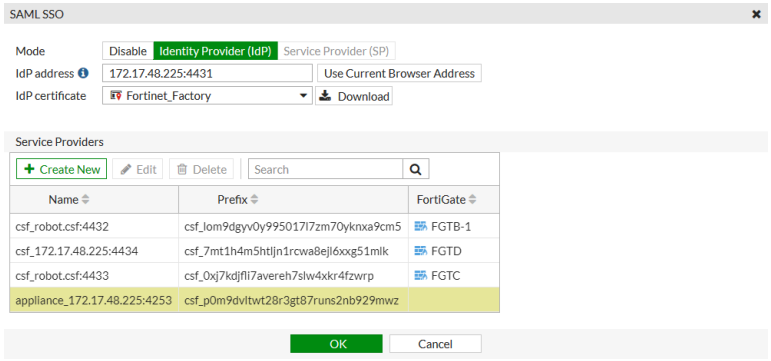
The following example assumes the root FortiGate (FGTA-1, server address 172.17.48.225:4431) has been configured as the SAML SSO IdP, and FortiAnalyzer logging has been enabled in the Security Fabric settings.

To enable FortiAnalyzer as a Fabric SP in the GUI:

1. In FortiAnalyzer, go to *System Settings* > *Admin* > *SAML SSO*.
2. For *Single Sign-On Mode*, click *Fabric SP* and enter the *SP Address*.



3. Click *Apply*.
   FortiAnalyzer will automatically register itself on the FortiGate as an appliance visible in the list of SPs. Go to *Security Fabric* > *Fabric Connectors*, edit the *Security Fabric Setup* connector, then click *Advanced Options* to view the list of SPs.



To enable FortiAnalyzer as a Fabric SP in the CLI:

1. In FortiAnalyzer, enable the device as a Fabric SP:

```
config system saml
    set status enable
    set role FAB-SP
```

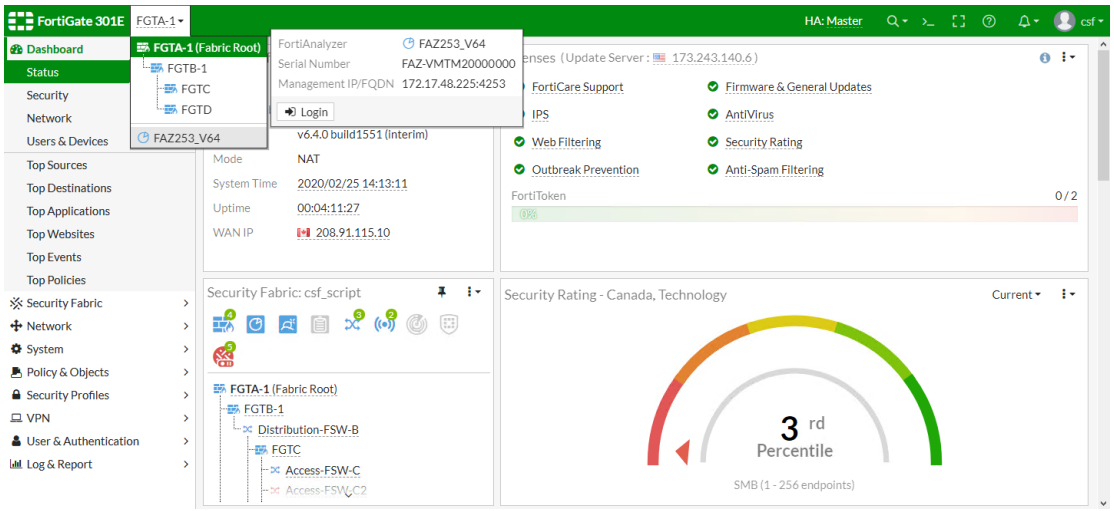
```
    set server-address "172.17.48.225:4253"
end
```

FortiAnalyzer will register itself on the FortiGate as an appliance. To view the configuration in FortiOS:

```
show system saml
    config service-providers
        edit "appliance_172.17.48.225:4253"
            set prefix "csf_p0m9dvltwt28r3gt87runs2nb929mwz"
            set sp-entity-id "http://172.17.48.225:4253/metadata/"
            set sp-single-sign-on-url "https://172.17.48.225:4253/saml/?acs"
            set sp-single-logout-url "https://172.17.48.225:4253/saml/?sls"
            set sp-portal-url "https://172.17.48.225:4253/saml/login/"
            config assertion-attributes
                edit "username"
                next
                edit "profilename"
                    set type profile-name
                next
            end
        next
    end
```

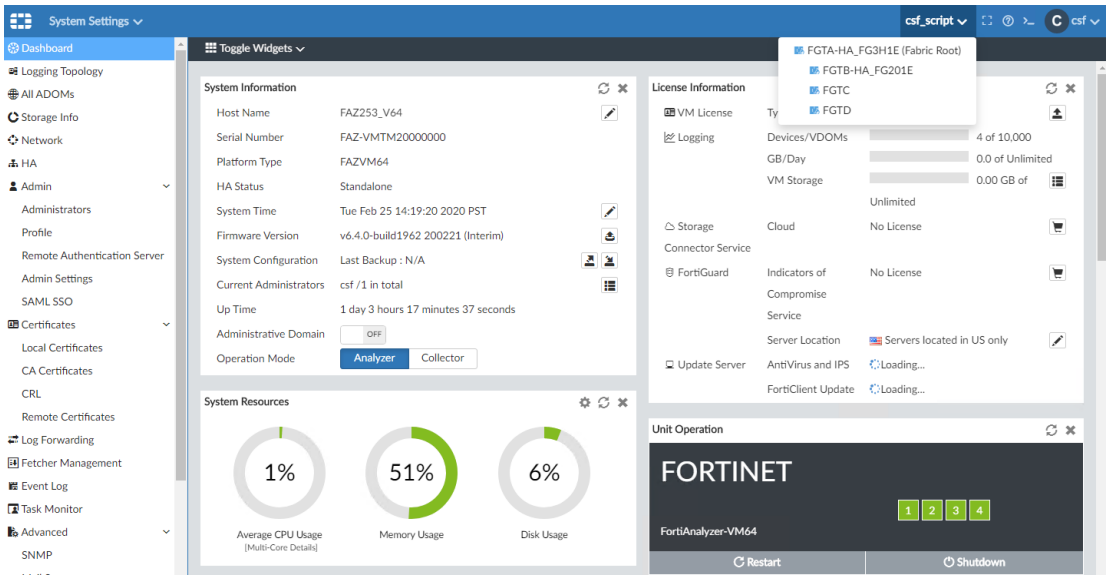
To navigate between devices using SAML SSO:

1. Log in to the root FortiGate.
2. In the toolbar, click the device name to display the Security Fabric members dropdown.
3. Hover over the FortiAnalyzer and click *Login*.



4. Log in to the FortiAnalyzer using SAML SSO.

5. In the toolbar, click the Security Fabric members dropdown to navigate between other FortiGates.



Group address objects synchronized from FortiManager

Address objects from external connectors that are learned by FortiManager are synchronized to FortiGate. These objects can be grouped together with the FortiGate CLI to simplify selecting connector objects in the FortiGate GUI. Multiple groups can be created.

This option is only available for objects that are synchronized from FortiManager.

To add an object to a connector group:

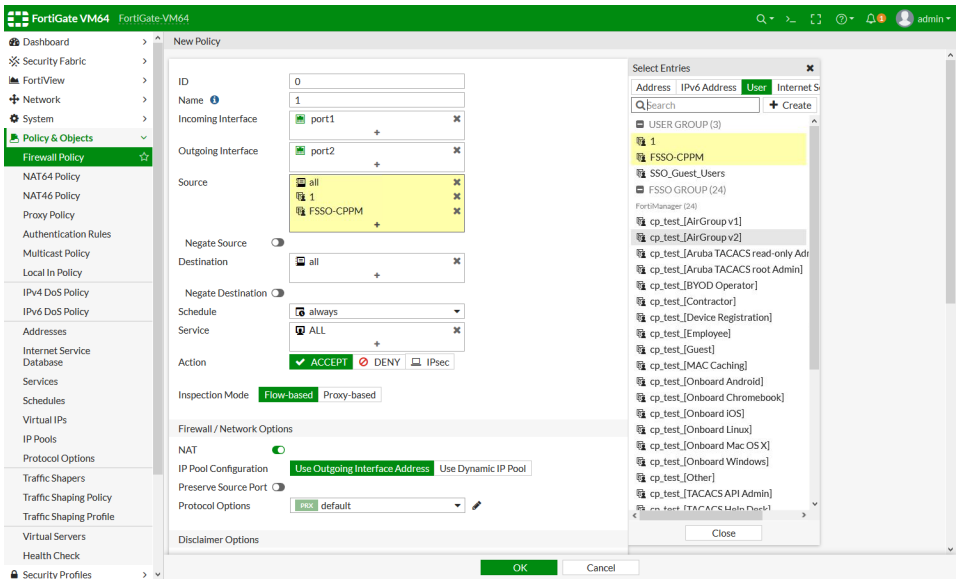
```
config user adgrp
    edit <object_name>
        set server-name "FortiManager"
        set connector-source <group_name>
    next
end
```

Example

In this example, objects learned by the FortiManager from an Aruba ClearPass device are synchronized to the FortiGate. Some of the objects are then added to a group called *ClearPass* to make them easier to find in the object list when creating a firewall policy.



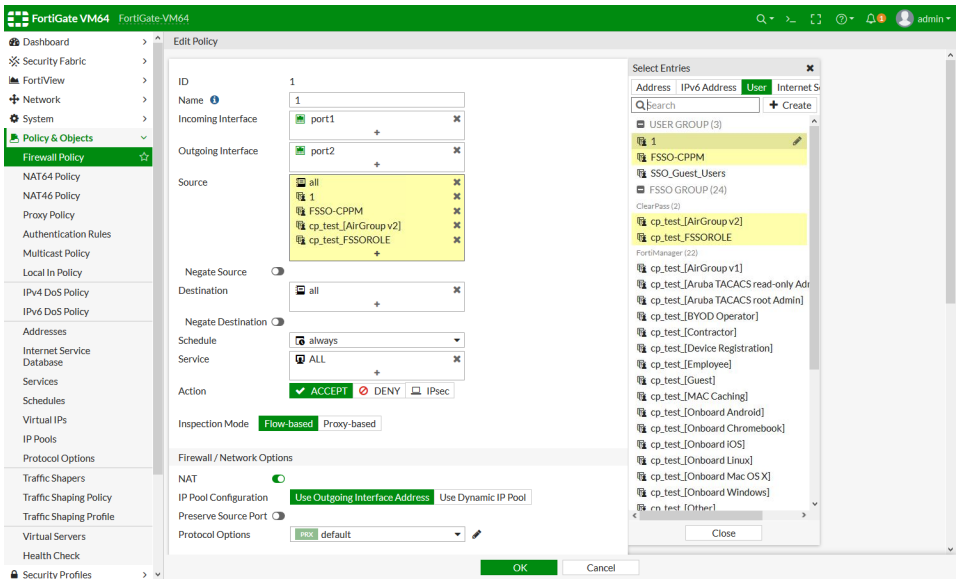
Prior to being grouped, the synchronized objects are listed under the FortiManager heading in the object lists.



To add some of the objects to a group:

```
config user adgrp
    edit "cp_test_FSSOROLE"
        set server-name "FortiManager"
        set connector-source "ClearPass"
    next
    edit "cp_test_[AirGroup v2]"
        set server-name "FortiManager"
        set connector-source "ClearPass"
    next
end
```

The objects are now listed under the *ClearPass* heading.



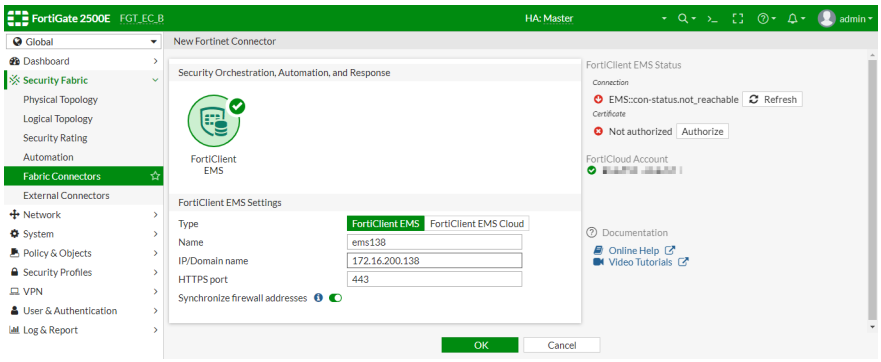
Simplify FortiClient EMS setup

EMS configurations are now centralized under one configuration card on the *Fabric Connectors* page. Certificates are the main mode of authentication and authorization. The certificate validity is verified against the issuer CA, and then presented to the user to authorize. A certificate attribute has been added to `endpoint-control fctems`, and EMS certificates can be verified with `execute fctems verify`.

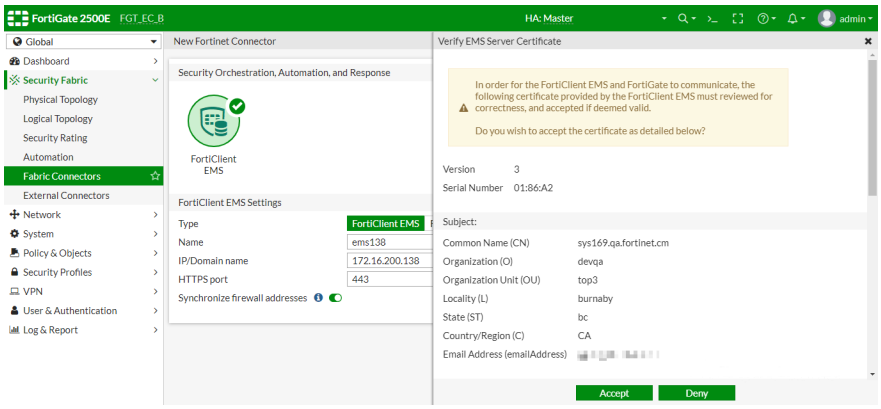
The following examples presume the EMS certificate has already been configured.

To configure an on-premise FortiClient EMS server to the Security Fabric in the GUI:

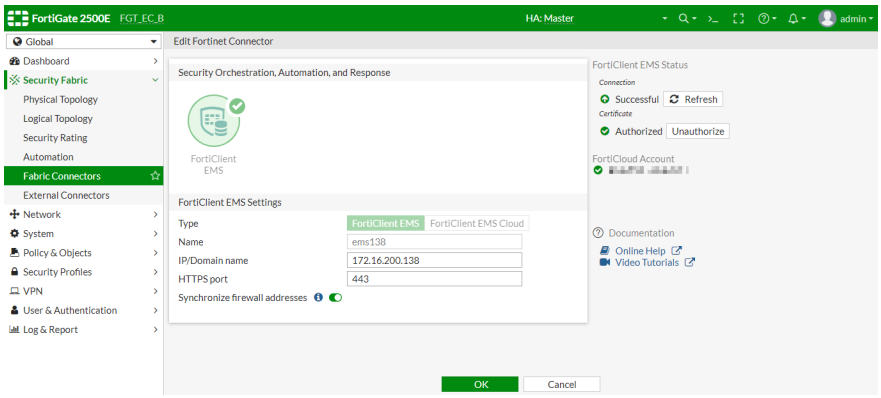
1. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors*.
2. Click *Create New* and click *FortiClient EMS*.
3. For *Type*, click *FortiClient EMS*.
4. Enter a name and IP address.
5. Click *OK*.



A window appears to verify the EMS server certificate:

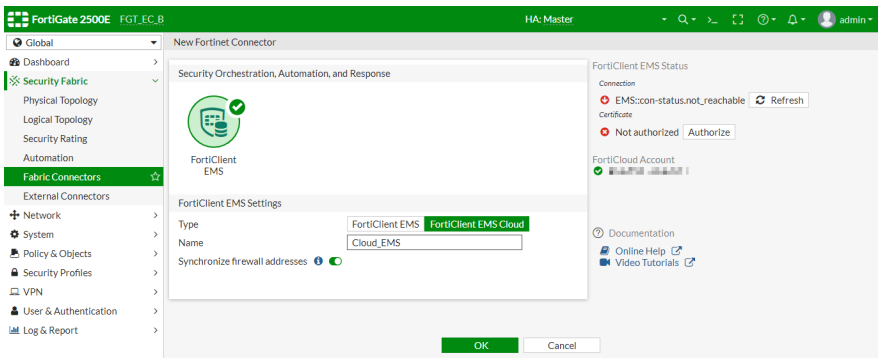


6. Click *Accept*.
   The *FortiClient EMS Status* section displays a *Successful* connection and an *Authorized* certificate:



To configure a FortiClient EMS Cloud server to the Security Fabric in the GUI:

1. Go to *Security Fabric* > *Fabric Connectors*.
2. Click *Create New* and click *FortiClient EMS*.
3. For *Type*, click *FortiClient EMS Cloud*.
4. Enter a name.
5. Click *OK*.



A window appears to verify the EMS server certificate.

6. Click *Accept*.
The *FortiClient EMS Status* section displays a *Successful* connection and an *Authorized* certificate.

To configure an on-premise FortiClient EMS server to the Security Fabric in the CLI:

```
config endpoint-control fctems
    edit "ems138"
        set server "172.16.200.138"
        set certificate "REMOTE_Cert_1"
    next
end
```

To configure a FortiClient EMS Cloud server to the Security Fabric in the CLI:

```
config endpoint-control fctems
    edit "Cloud_EMS"
        set fortinetone-cloud-authentication enable
        set certificate "REMOTE_Cert_1"
```

```
        next
    end
```

To verify an EMS certificate in the CLI:

```
# execute fctems verify ems137

        Subject:     C = CA, ST = bc, L = burnaby, O = devqa, OU = top3, CN = sys169.qa.-
fortinet.cm, emailAddress = xxxx@xxxxxxxx.xxx
        Issuer:      CN = 155-sub1.fortinet.com
        Valid from:  2017-12-05 00:37:57  GMT
        Valid to:    2027-12-02 18:08:13  GMT
        Fingerprint: D3:7A:1B:84:CC:B7:5C:F0:A5:73:3D:BB:ED:21:F2:E0
        Root CA:     No
        Version:     3
        Serial Num:
                01:86:a2
        Extensions:
                Name:     X509v3 Basic Constraints
                Critical: yes
                Content:
                CA:FALSE

                Name:     X509v3 Subject Key Identifier
                Critical: no
                Content:
                35:B0:E2:62:AF:9A:7A:E6:A6:8E:AD:CB:A4:CF:4D:7A:DE:27:39:A4

                Name:     X509v3 Authority Key Identifier
                Critical: no
                Content:
                keyid:66:54:0F:78:78:91:F2:E4:08:BB:80:2C:F6:BC:01:8E:3F:47:43:B1
DirName:/C-
C=CA/ST=bc/L=burnaby/O=devqa/OU=top3/CN=fac155.fortinet.com/emailAddress=xyguo@fortinet.com
serial:01:86:A4


                Name:     X509v3 Subject Alternative Name
                Critical: no
                Content:
                DNS:sys169.qa.fortinet.cm

                Name:     X509v3 Key Usage
                Critical: no
                Content:
                Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment, Key
Agreement, Certificate Sign, CRL Sign, Encipher Only, Decipher Only

                Name:     X509v3 Extended Key Usage
                Critical: no
                Content:
                TLS Web Server Authentication, TLS Web Client Authentication

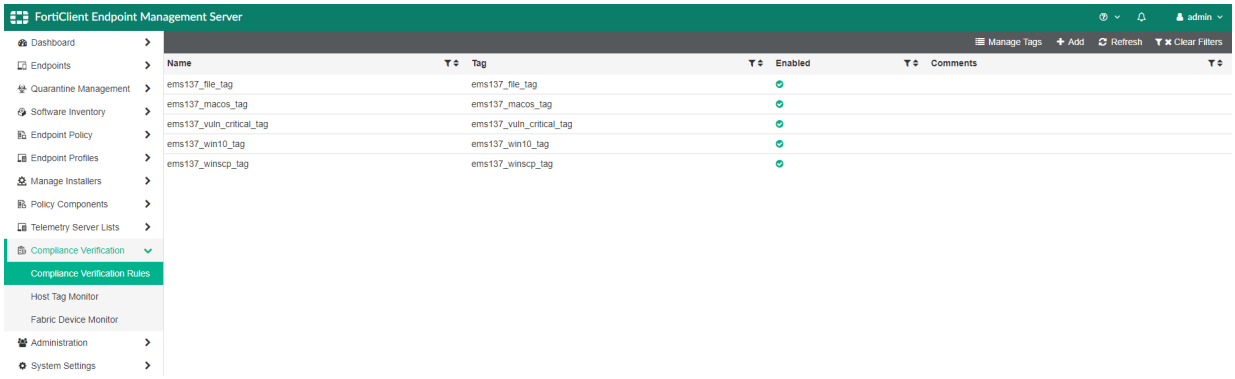
EMS configuration needs user to confirm server certificate.
Do you wish to add the above certificate to trusted remote certificates? (y/n)y
```

Simplify the synchronization of EMS tags and configurations

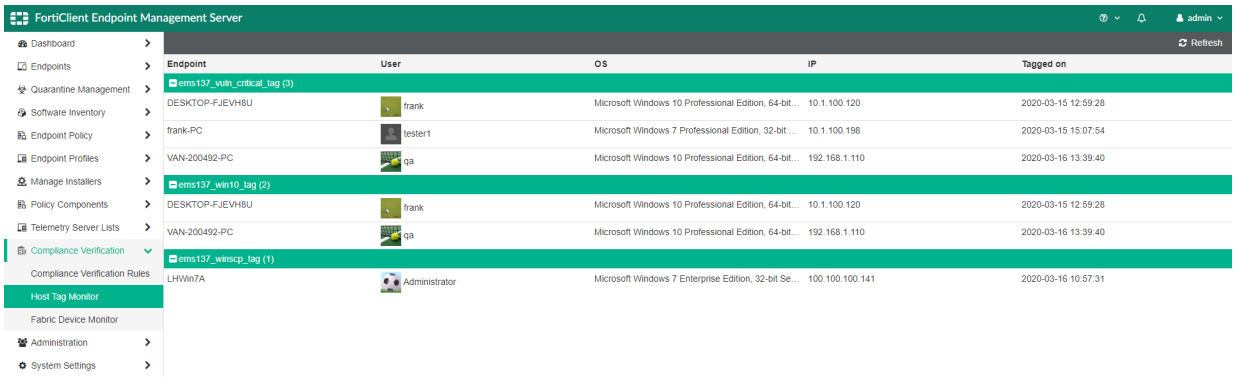
A new option under the FortiClient EMS settings consolidates the setup of EMS connectors to support EMS tags. EMS tags are pulled and automatically synced with the EMS server. They are converted into read-only dynamic firewall addresses that can be used in firewall policies, routing, and so on.

These examples presume the following have been configured in FortiClient EMS:

- Tags have been created on the *Compliance Verification* > *Compliance Verification Rules* page.

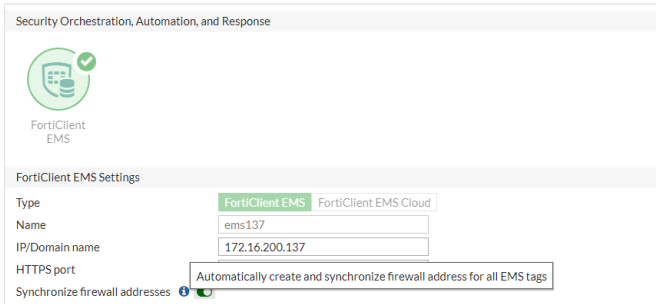


- There are registered users who match the defined tags that are visible on the *Compliance Verification* > *Host Tag Monitor* page.



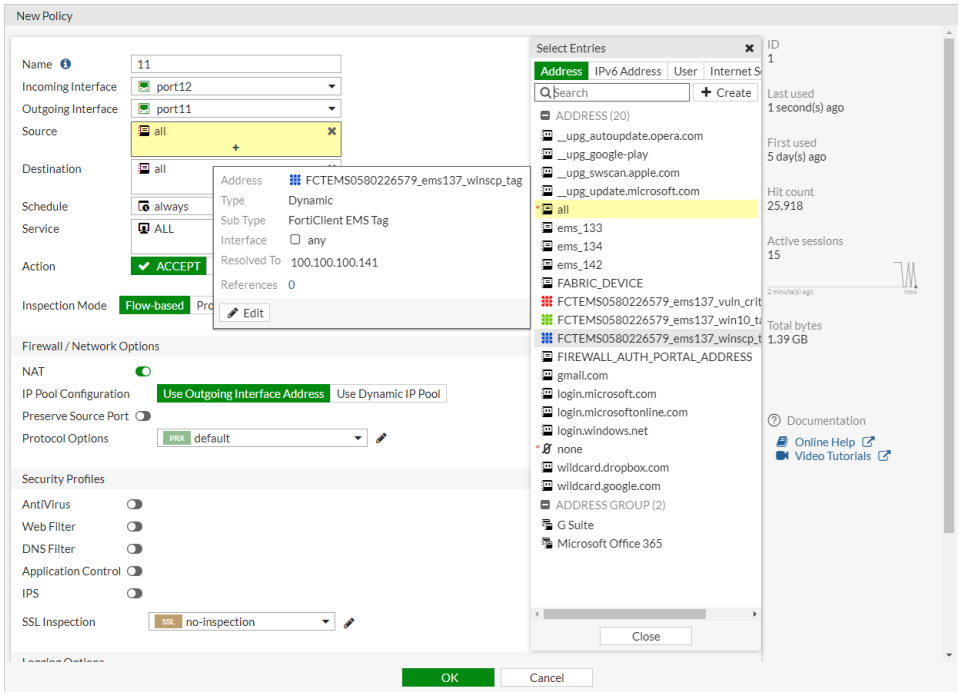
To configure FortiClient EMS with tag synchronization in the GUI:

1. Configure the EMS Fabric Connector:
   a. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors*.
   b. Click *Create New* and click *FortiClient EMS*.
   c. Enable *Synchronize firewall addresses*.



d. Configure the other settings as needed and validate the certificate.
e. Click *OK*.

2. Go to *Policy & Objects* > *Addresses* and hover over the EMS tag to view which IPs it resolves to.
3. Configure a firewall policy:
   a. Go to *Policy & Objects* > *Firewall Policy* and create a new policy.
   b. For the *Source Address*, add the EMS tag dynamic address.



   c. Configure the other settings as needed.
   d. Click *OK*.

To configure FortiClient EMS with tag synchronization in the CLI:

1. Configure the EMS Fabric Connector:

```
config endpoint-control fctems
    edit "ems137"
        set fortinetone-cloud-authentication disable
        set server "172.16.200.137"
        set https-port 443
        set source-ip 0.0.0.0
        set pull-sysinfo enable
        set pull-vulnerabilities enable
        set pull-avatars enable
        set pull-tags enable
        set call-timeout 5000
        set certificate "REMOTE_Cert_1"
    next
end
```

2. Verify which IPs the dynamic firewall address resolves to:

```
# diagnose firewall dynamic list
List all dynamic addresses:
FCTEMS0580226579_ems137_vuln_critical_tag: ID(118)
        ADDR(10.1.100.120)
        ADDR(10.1.100.198)

FCTEMS0580226579_ems137_winscp_tag: ID(155)
        ADDR(100.100.100.141)

FCTEMS0580226579_ems137_win10_tag: ID(182)
        ADDR(10.1.100.120)

# diagnose firewall dynamic address FCTEMS0580226579_ems137_vuln_critical_tag
FCTEMS0580226579_ems137_vuln_critical_tag: ID(118)
        ADDR(10.1.100.120)
        ADDR(10.1.100.198)

Total dynamic list entries: 1.
Total dynamic addresses: 2
Total dynamic ranges: 0
```

3. Configure a firewall policy that uses the EMS tag dynamic firewall address as a source.

Allow FortiNAC to join the Security Fabric

A FortiNAC device can be added to the Security Fabric on the root FortiGate. After the device has been added and authorized, you can log in to the FortiNAC from the FortiGate topology views.



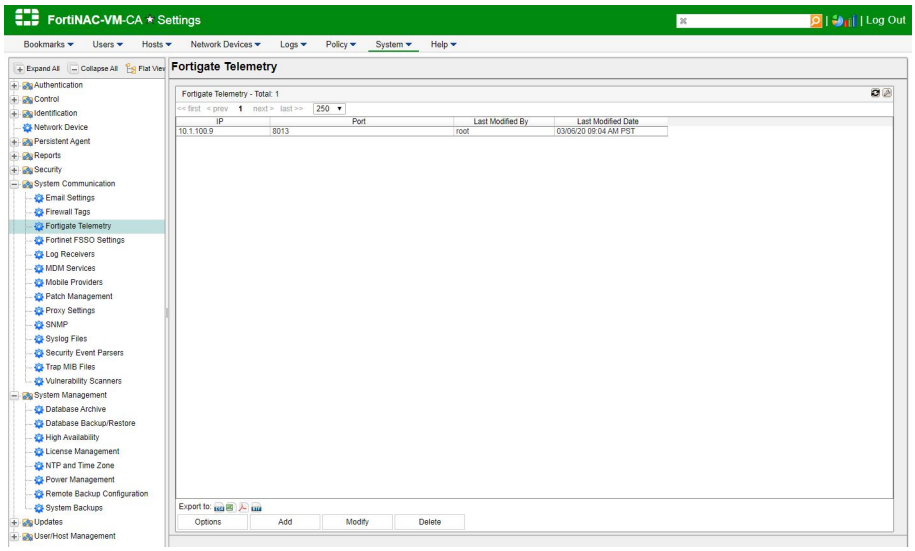
Adding a FortiNAC to the Security Fabric requires a FortiNAC with a license issued in the year 2020 that includes an additional certificate. The device cannot be added if it has an older license. Use the `licensetool` in the FortiNAC CLI to determine if your license includes the additional certificate

To add a FortiNAC to the Security Fabric:

1. On the FortNAC, configure telemetry and input the IP address of the root FortiGate.
2. On the root FortiGate, authorize the FortiNAC.
3. Verify the connection status in the topology views.

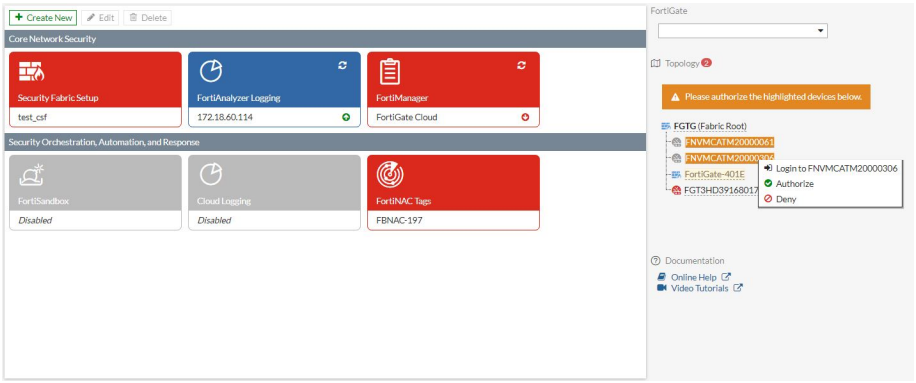
To configure the FortiNAC:

1. Go to *Settings* > *System Communication* > *FortiGate Telemetry*.
2. Add a new entry with the root FortiGate device's IP address. The default port is 8013.



To authorize the FortiNAC on the root FortiGate in the GUI:

1. Go to *Security Fabric* > *Fabric Connectors*.
2. The FortiNAC device will be highlighted in the topology list in the right panel with the status *Waiting for Authorization*.
3. Click on the highlighted FortiNAC and select *Authorize*.



Optionally, you can also deny authorization to the FortiNAC to remove it from the list.



Joining a FortiNAC to the Security Fabric is not related to FortiNAC Tags FSSO connectors. See FortiNAC endpoint connector for information about the FSSO connector.

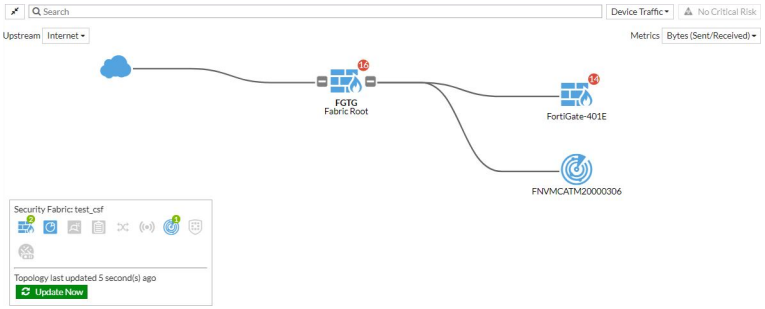
To authorize the FortiNAC on the root FortiGate in the CLI:

```
config system csf
    config trusted-list
```

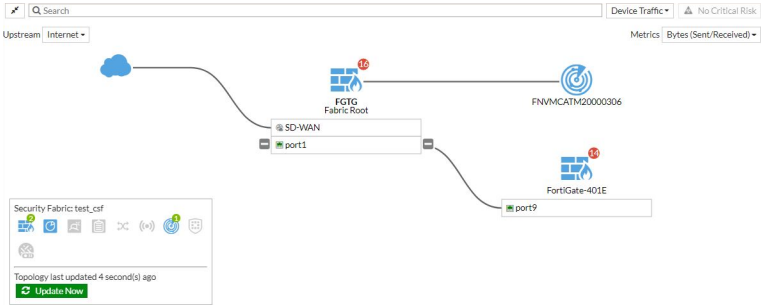
```
        edit "FNVMCATM20000306"
            set action accept
        next
    end
end
```

To verify the connection status:

1. After the FortiNAC is authorized, go to *Security Fabric* > *Physical Topology* and confirm that it is included in the topology.



2. Go to *Security Fabric* > *Logical Topology* and confirm the FortiNAC is also displayed there.

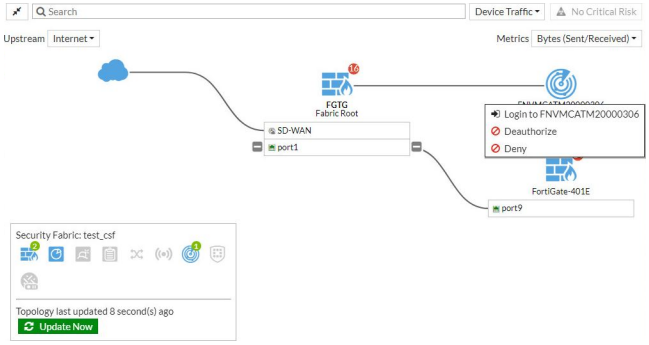


3. Run the following command in the CLI to view information about the FortiNAC device's status:

```
# diagnose sys csf downstream-devices fortinac
{
  "path":"FG5H1E5818900126:FNVMCATM20000306",
  "mgmt_ip_str":"10.1.100.197",
  "mgmt_port":0,
  "admin_port":8443,
  "serial":"FNVMCATM20000306",
  "host_name":"adnac",
  "device_type":"fortinac",
  "upstream_intf":"port2",
  "upstream_serial":"FG5H1E5818900126",
  "is_discovered":true,
  "ip_str":"10.1.100.197",
  "downstream_intf":"eth0",
  "authorizer":"FG5H1E5818900126",
  "idx":1
}
```

To log in to the FortiNAC from the FortiGate:

1. On the FortiGate, go to *Security Fabric* > *Physical Topology* or *Security Fabric* > *Logical Topology*.
2. Click on the FortiNAC and select *Login to <serial_number>*.



A new tab will open to the FortiNAC log in page.

3. Enter the username and password to log in to the FortiNAC.

Exchange Server connector with Kerberos KDC auto-discovery

FortiOS takes the domains learned from LDAP user authentication, and uses DNS to discover the IP addresses of Kerberos KDC servers for those domains.

The Exchange User connector is used to connect to Exchange, and other domain, servers and collect information about users. The connector can be used in conjunction with an LDAP server. The Kerberos KDC service in the domain server accepts queries to provide access and information about users in the domain.

By default, KDC discovery is automatic. If auto-discovery is disabled, the KDC IP address must be manually configured.

To configure an Exchange connector with automatic KDC discovery:

```
config user exchange
    edit "exchange140"
        set server-name "W2K8-SERV1"
        set domain-name "FORTINET-FSSO.COM"
        set username "Administrator"
        set password ENC XXXXXXXXXXXXXXXXXXXXXXX
        set ip 10.1.100.140
        set auto-discover-kdc enable
    next
end
```

To verify that auto-discovery is working:

```
# diagnose wad debug enable category all
# diagnose wad debug enable level verbose
# diagnose debug enable
# diagnose wad user exchange test-auto-discover

wad_diag_session_acceptor(3115): diag socket 20 accepted.
__wad_fmem_open(557): fmem=0x12490bd8, fmem_name='cmem 9188 bucket', elm_sz=9188, block_
sz=73728, overhead=0, type=advanced
Starting auto-discover test for all configured user-exchanges.
```

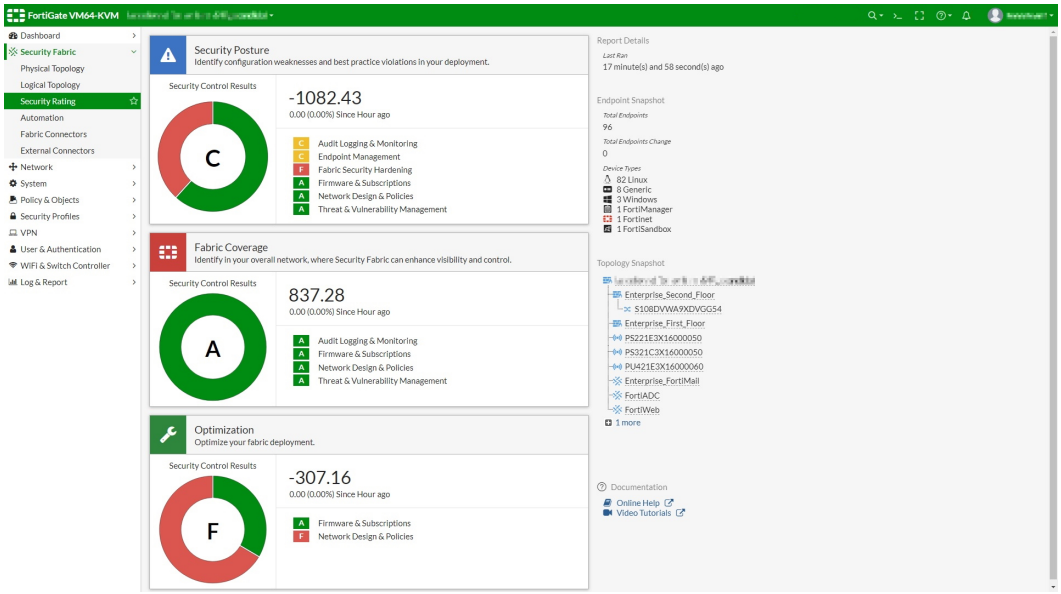
```
[NOTE]: If any errors are returned, try manually configuring IPs for the reported errors.

wad_rpc_nspi_test_autodiscover_kdc(1835): Starting DNS SRV request for srv(0x7f938e052050)
query(_kerberos._udp.FORTINET-FSSO.COM)
wad_dns_send_srv_query(705): 1:0: sending DNS SRV request for remote peer _kerberos._
udp.FORTINET-FSSO.COM id=0
1: DNS response received for remote host _kerberos._udp.FORTINET-FSSO.COM req-id=0
wad_dns_parse_srv_resp(409): _kerberos._udp.FORTINET-FSSO.COM: resp_type(SUCCESS)
  srv[0]: name(w2k12-serv1.fortinet-fsso.com) port(88) priority(0) weight(100)
    addr[0]: 10.1.100.131
    addr[1]: 10.6.30.131
    addr[2]: 172.16.200.131
    addr[3]: 2003::131
    addr[4]: 2001::131
  srv[1]: name(fsso-core-DC.Fortinet-FSSO.COM) port(88) priority(0) weight(100)
    addr[0]: 10.6.30.16
    addr[1]: 172.16.200.16
  srv[2]: name(w2k12-serv1.Fortinet-FSSO.COM) port(88) priority(0) weight(100)
    addr[0]: 10.1.100.131
    addr[1]: 172.16.200.131
    addr[2]: 10.6.30.131
    addr[3]: 2001::131
    addr[4]: 2003::131
wad_rpc_nspi_dns_on_discover_kdc_done(1787): Received response for DNS autodiscover req
(0x7f938dfe8050) query(_kerberos._udp.FORTINET-FSSO.COM) n_rsp(3)

Completed auto-discover test for all configured user-exchanges.
```

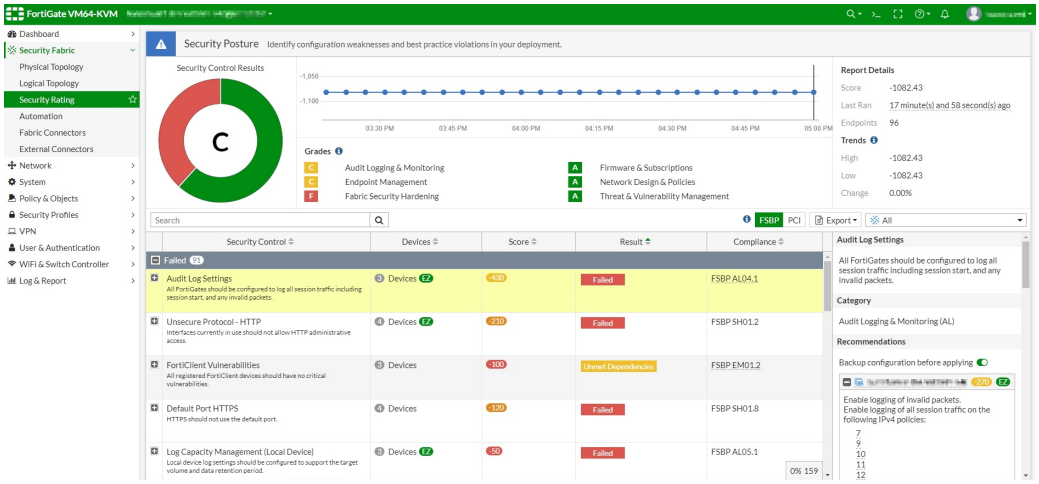
Redesign Security Rating scorecards

The *Security Rating* page is separated into three major scorecards: *Security Posture*, *Fabric Coverage*, and *Optimization*, which provide an executive summary of the three largest areas of security focus in the Security Fabric.



This page is only visible on the root FortiGate or a standalone FortiGate. It is not visible on downstream FortiGates.

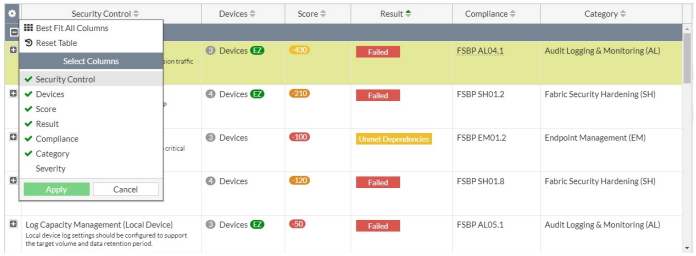
The scorecards show an overall letter grade and breakdown of the performance in sub-categories. Clicking a scorecard drills down to a detailed report of itemized results and compliance recommendations. The point score represents the net score for all passed and failed items in that area. The report includes the security controls that were tested against, linking to specific FSBP or PCI compliance policies. Click the *FSBP* and *PCI* buttons to reference the corresponding standard.



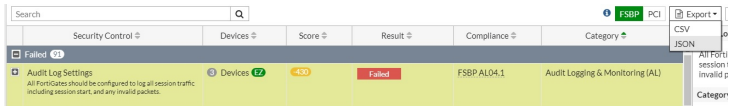
Certain remediations marked with an *EZ* symbol represent configuration recommendations that support *Easy Apply*. In the panel on the right, in the *Recommendations* section, click *Apply* to apply the changes to resolve the failed security control.



The report table can be customized by adding more columns, such as *Category*, to view, filter, or sort the results based on scorecard categories. Click the gear icon to customize the table.



Users can also export the reports as CSV or JSON files by clicking the *Export* dropdown.





To exit the current view, click the icon beside the scorecard title to return to the summary view.

Security rating check scheduling

Security rating checks by default are scheduled to run automatically every four hours.

To disable automatic security checks using the CLI:

```
config system global
    security-rating-run-on-schedule disable
end
```

To manually run a report using the CLI:

```
# diagnose report-runner trigger
```

Redesign Fortinet Fabric Connectors and Fabric setup pages

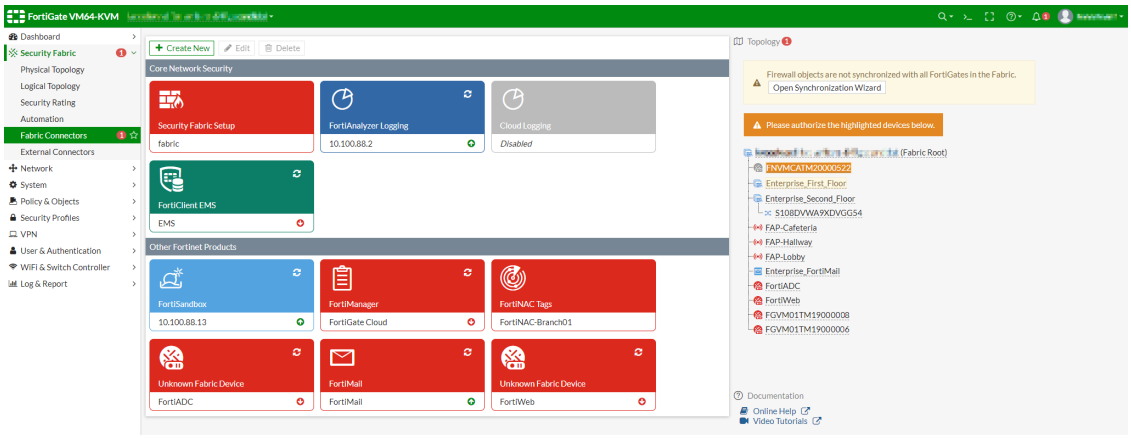
In FortiOS 6.4.0 there have been several changes to simplify the GUI for the Security Fabric menu.

- The *Security Fabric Settings* page has been renamed to *Fabric Connectors* and all the settings under it now show up as separate cards. The cards that appear by default are: *Security Fabric Setup*, *FortiAnalyzer Logging*, *FortiManager*, *FortiSandbox*, and *Cloud Logging*.
- The new *Fabric Connectors* menu contains a card view similar to *External Connectors* for various Fortinet products (FortiSandbox, FortiManager, Cloud Logging, and so on).
- The *Fabric Connectors* menu has been renamed to *External Connectors* where third-party connectors are configured.
- Each card has a separate page with its own edit dialog view.
- The topology tree, various statistics, and connectivity results have been moved from the main dialog to the gutter.

Fabric Connectors page

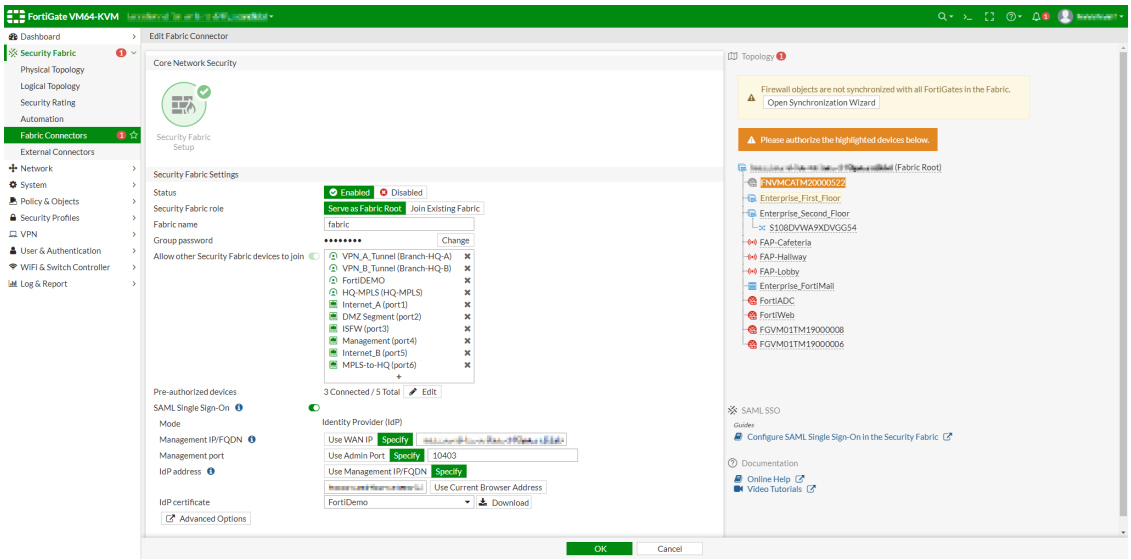
This page displays all configured Security Fabric Devices with their own card. The card also display the device connectivity status with a green up arrow or a red down arrow.

The topology tree and notifications are displayed in the gutter. In this example, there is a device that requires authorization.



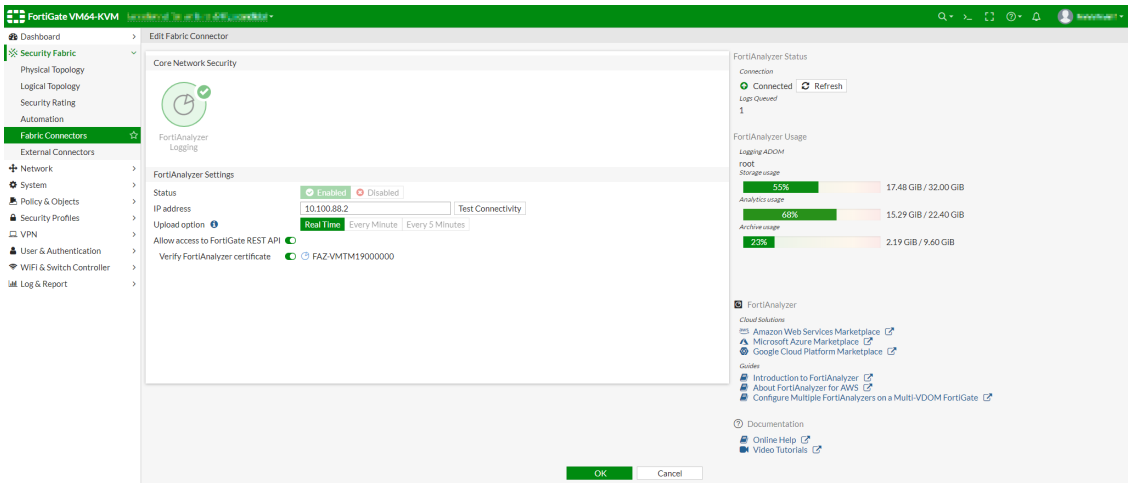
Security Fabric Setup page

This page displays the Security Fabric settings, including SAML SSO. In previous versions these settings were under the *FortiTelemetry* section. The topology tree is also displayed in the gutter.



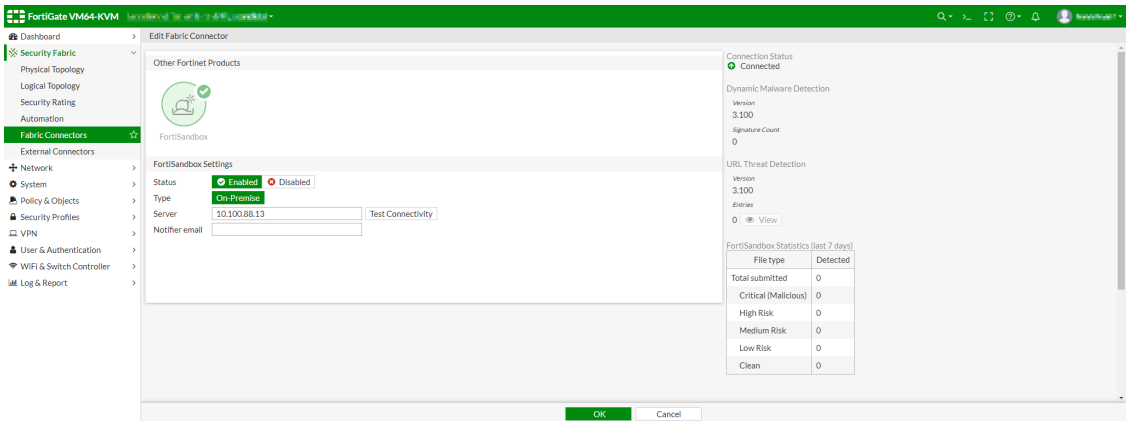
FortiAnalyzer Logging page

The gutter in this page displays the connection status and usage information.



FortiSandbox page

The gutter in this page displays the connection status, dynamic malware and URL threat detection information, and FortiSandbox statistics.

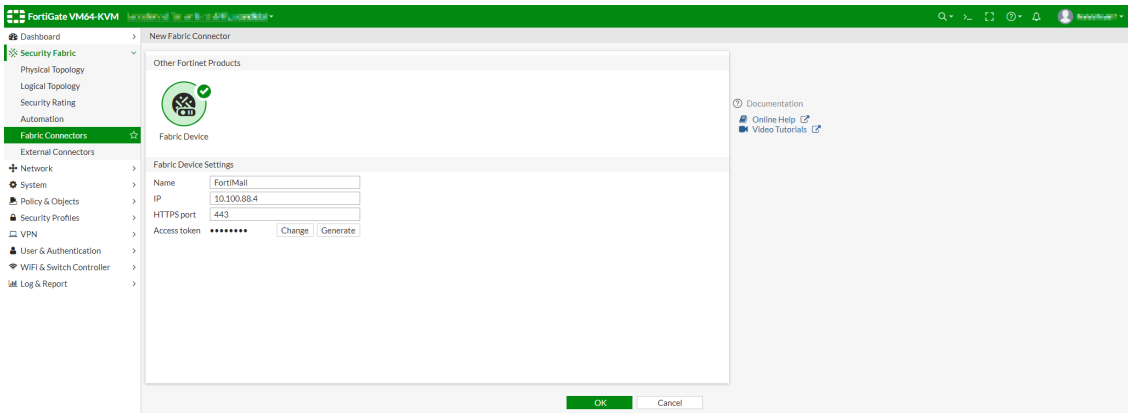


Adding a new Fabric connector

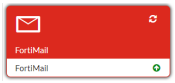
To configure a FortiMail connector:

1. Go to *Security Fabric* > *Fabric Connectors* and click *Create New*.
2. Click *Fabric Device*.

3. Enter the FortiMail settings as needed.



4. Click *OK*.
   The FortiGate will attempt to connect to the device to authorize it. Once the device is authorized, the device name will no longer be displayed as *Unknown Fabric Device* in the card. The corresponding device name and icon are displayed in the card.

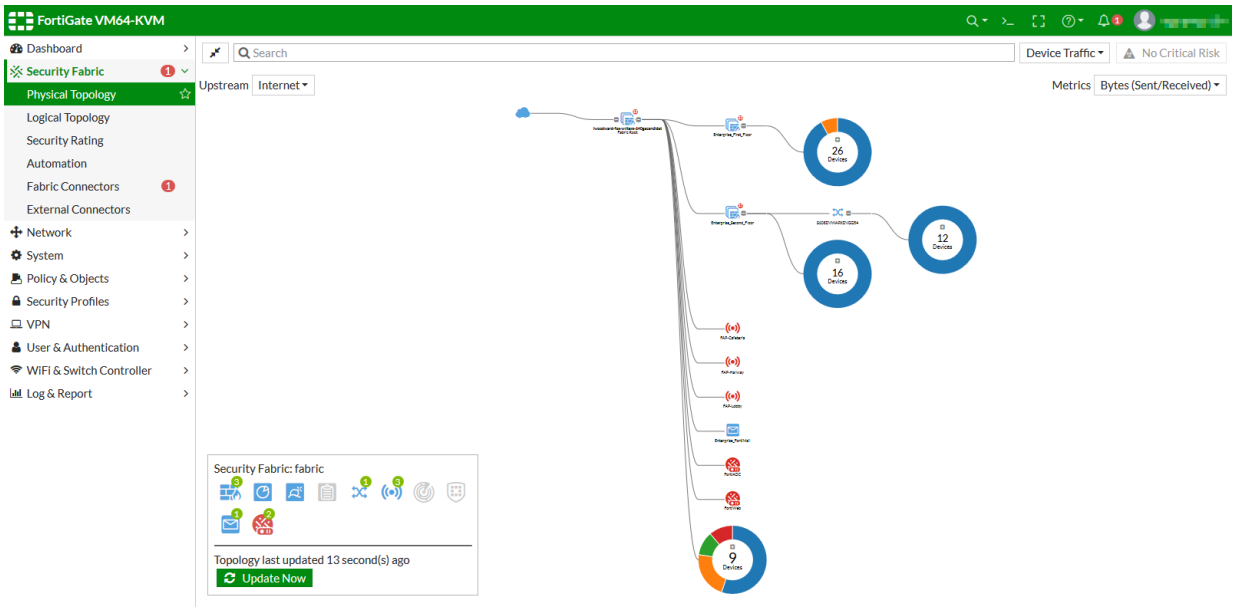


Display endpoints in Topology using donut chart

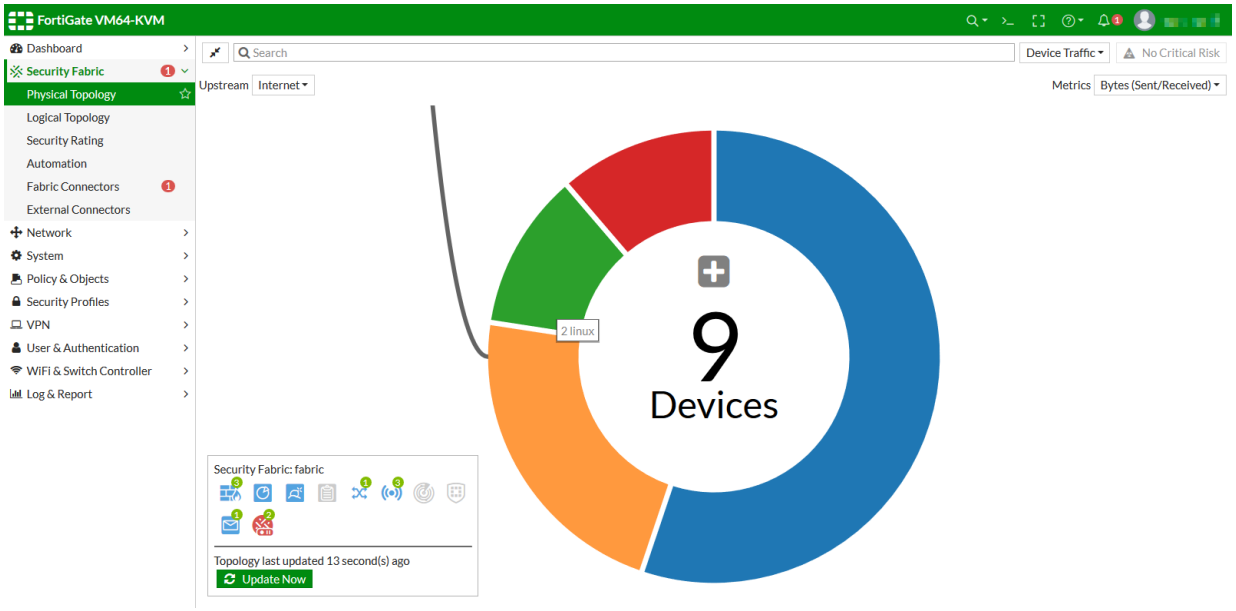
On the Physical and Logical Topology pages, the Device Traffic and Device Count views now display endpoint groups as donut charts. Each sector of the donut chart represents a different endpoint operating system. This new donut chart display allows for faster loading times and provides more stability with zoom operations. This improvement is especially useful for deployments with a large number of endpoints, while retaining the bubble pack endpoint view from earlier versions of FortiOS.

To view endpoints in Topology using donut charts:

1. Go to *Security Fabric* > *Physical Topology* or *Logical Topology*.
2. From the *Endpoint Option* dropdown list, select *Device Traffic* or *Device Count*. FortiOS represents each endpoint group as a donut chart, with the total number of endpoints in the group in the center of the chart.



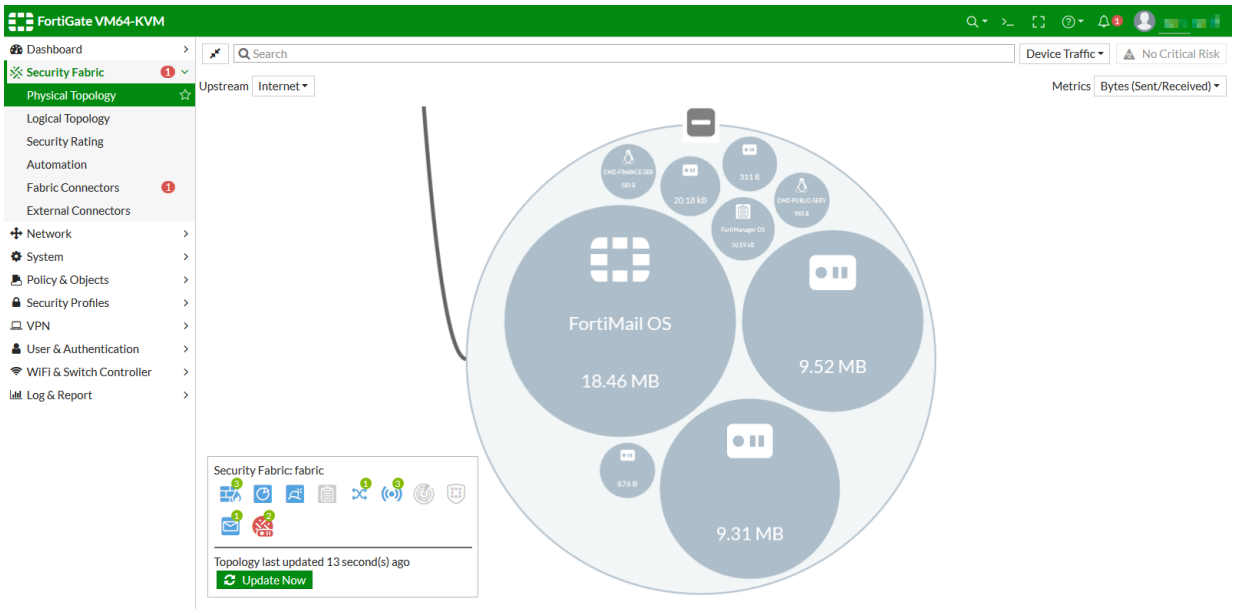
To zoom in on a donut chart, click any chart sector. Each sector represents a different endpoint OS. Hovering over each sector allows you to see the OS that the sector represents and the number of endpoints that have that OS installed.



In this example, the endpoint group contains a total of nine endpoints, with the following OSes installed:

Donut sector color	OS	Number of endpoints
Orange	Linux	2
Green	FortiMail	1
Red	FortiManager	1
Blue	Other	5

To view the endpoint group in a bubble pack display, click the + button in the center of the donut chart. You can view each individual endpoint in the bubble pack view.



To return to the donut chart display, click the - button at the top of the bubble.

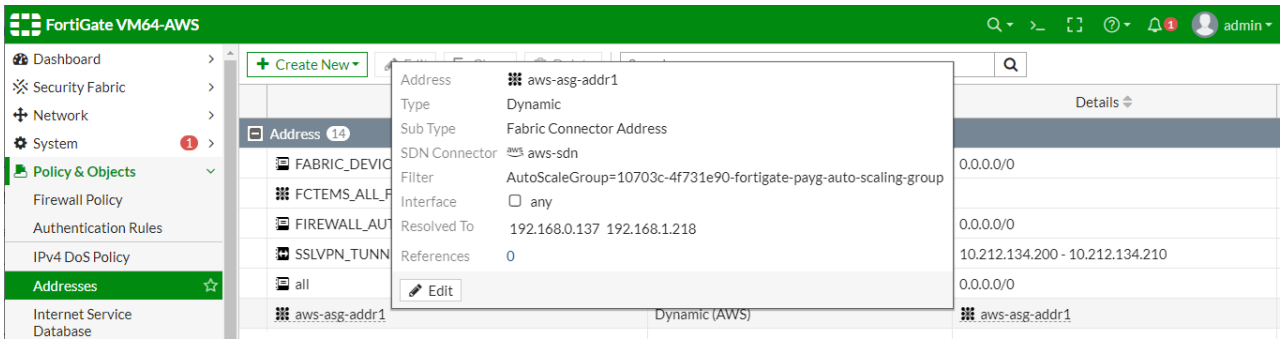
Support filtering on AWS autoscaling group for dynamic address objects

A FortiGate-VM deployed on AWS can create a dynamic address based on an AWS Fabric connector and use an autoscaling group (ASG) filter to obtain ASG members' primary IP addresses or NICs. You can use this feature for load balancing to optimize network efficiency.

To create an address with an ASG filter using the GUI:

1. In FortiOS, go to *Policy & Objects* > *Addresses*.
2. Click *Create New*, then select *Address*.
3. Enter the address name. From the *Type* dropdown list, select *Dynamic*.
4. From the *Sub Type* dropdown list, select *Fabric Connector Address*.
5. From the *SDN Connector* dropdown list, select the AWS Fabric connector.
6. In the *Filter* fields, enter the desired filter. In this example, you would enter AutoScaleGroup=<ASG ID> in the *Filter* field.
7. From the *Interface* dropdown list, select an interface where the Fabric connector covers where relevant.

8. Click *OK*. Once saved, FortiOS lists the address under *Policy & Objects* > *Addresses*.



To create an address with an ASG filter using the CLI:

```
config firewall address
   edit "aws-asg-addr1"
      set uuid 82e26cea-756e-51ea-d322-4259d3db301b
      set type dynamic
      set sdn "aws-sdn"
      set filter "AutoScaleGroup=10703c-4f731e90-fortigate-payg-auto-scaling-group"
      config list
         edit "192.168.0.137"
         next
         edit "192.168.1.218"
         next
      end
   next
end
```

Support dynamic address objects in real servers under virtual server load balance

FortiOS supports using dynamic firewall addresses in real servers under a virtual server load balancing configuration. Combined with support for the autoscaling group filter (see Support filtering on AWS autoscaling group for dynamic address objects on page 200), this enables you to use the FortiGate as a load balancer in AWS for an autoscaling deployment. You do not need to manually change each server's IP address whenever a scale in/out action occurs, as FortiOS dynamically updates the IP addresses following each scale in/out action.

Consider a scenario where the FortiGate-VM is deployed on AWS and load balancing for three servers. The Fabric connector configured in FortiOS dynamically loads the server IP addresses. If a scale in action occurs, the load balancer dynamically updates to load balance to the two remaining servers.

The following instructions assume the following:

1. An AWS Fabric connector is configured and up.
2. An AWS dynamic firewall address with a filter is configured.

To configure a dynamic address object in a real server under virtual server load balance:

CLI commands introduced in FortiOS 6.4.0 are shown bolded below.

```
config firewall vip
   edit "0"
```

```
        set id 0
        set uuid 0949dfbe-7512-51ea-4671-d3a706b09657
        set comment ''
        set type server-load-balance
        set extip 0.0.0.0
        set extintf "port1"
        set arp-reply enable
        set server-type http
        set nat-source-vip disable
        set gratuitous-arp-interval 0
        set http-ip-header disable
        set color 0
        set ldb-method static
        set http-redirect disable
        set persistence none
        set extport 80
        config realservers
           edit 1
              set type address
              set address "aws addresses"
              set port 8080
              set status active
              set holddown-interval 300
              set healthcheck vip
              set max-connections 0
              unset client-ip
           next
        end
        set http-multiplex disable
        set max-embryonic-connections 1000
    next
end
```

Consolidate Monitor and FortiView pages

The Monitoring & FortiView consoles were removed from the tree menu and now appear as widgets in the Dashboards menu. The dashboard navigation has been improved for easy access to features such as creating a new dashboard. New widgets were added to the Add Widget menu to create custom dashboards.

Improved navigation and functionality

Dashboard options are now located in the dashboard banner for easier access and visibility. Users can use a widget to create a standalone dashboard.

New WiFi health monitor

- The existing WiFi Health Monitor widgets were consolidated into a new default WiFi dashboard in the tree menu. This dashboard contains the following widgets:
- FortiAP Status
- Channel Utilization
- Clients By FortiAP
- Signal Strength

- Rogue APs
- Historical Clients
- InterfereAPs
- Login Failures

New network monitor

A new Network dashboard was added to the tree menu. This dashboard contains the following widgets:

- Routing Monitor
- DHCP Monitor
- SD-WAN Monitor
- IPsec Monitor
- SSL-VPN Monitor

Consolidated Formative menu

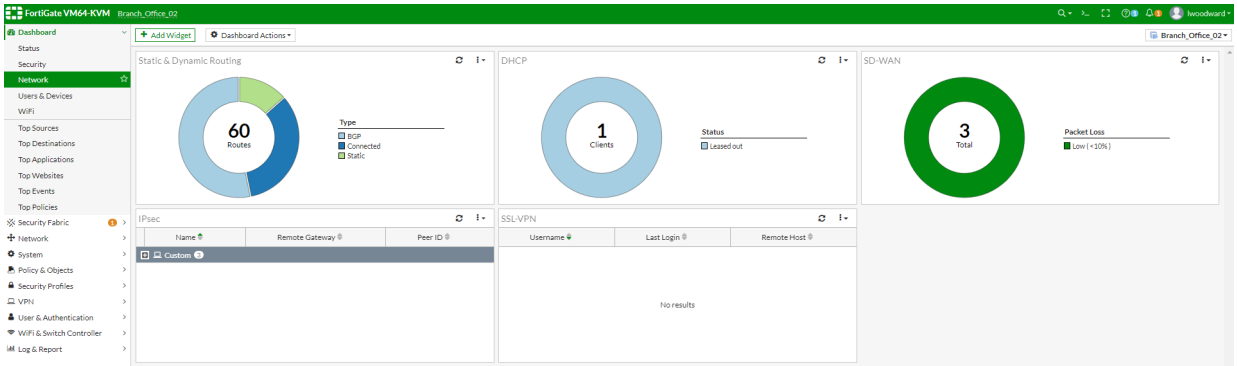
The FortiView module has been consolidated into dashboard widgets. The sections of the FortiView module that were not incorporated into widgets were moved to the Log & Reports module in the tree menu. The Threat Map was removed from the GUI.

The following FortiView widgets were created:

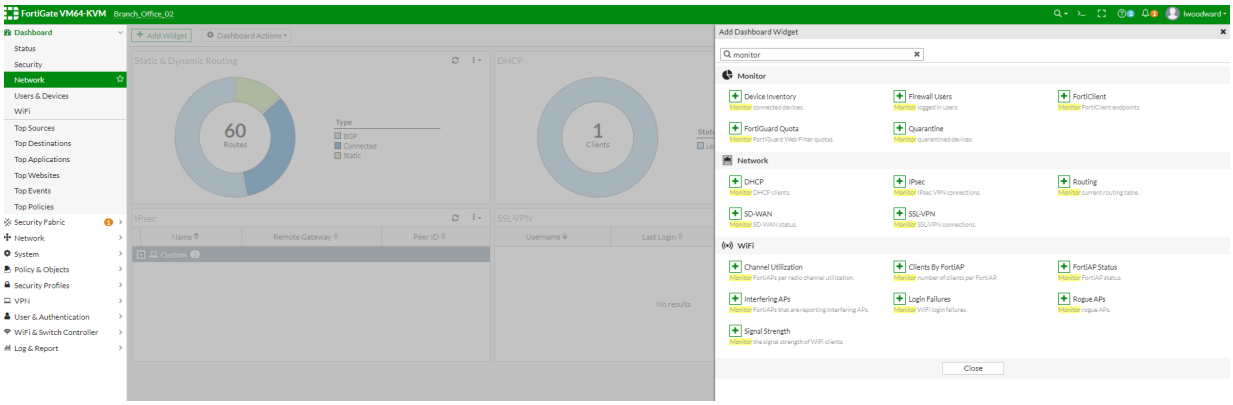
- FortiClient
- Firewall users
- Quarantine

To add a widget to a dashboard:

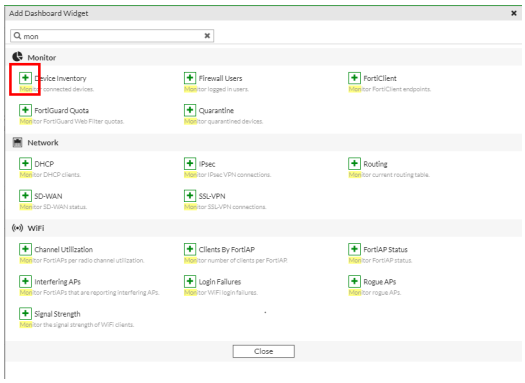
1. Open a dashboard in the tree menu.
2. In widget banner, click *Add Widget*. The *Add Dashboard Widget* window opens. The widgets are organized by category.



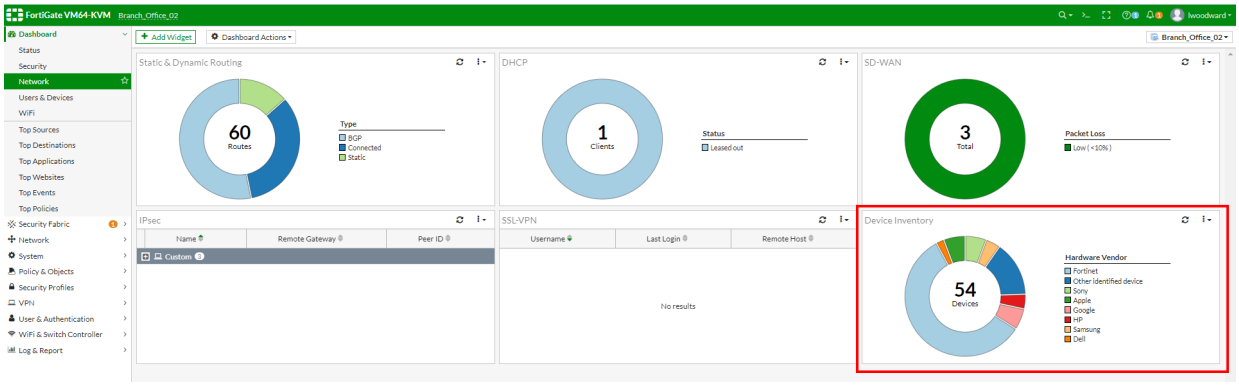
3. Enter a search term in the *Search* field to find a widget, or scroll through the window.



4. Click the *Add* icon to add the widget to the dashboard.

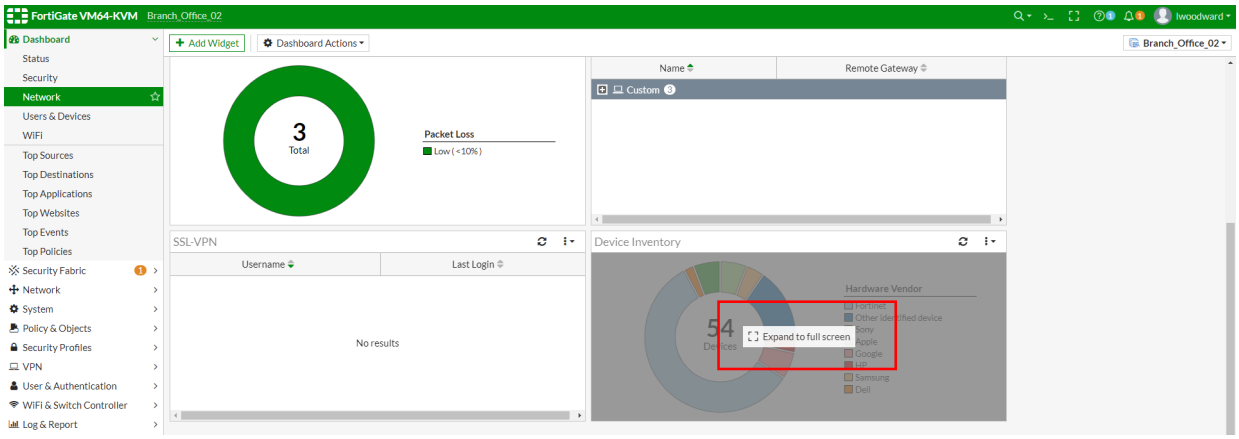


The widget is added to the dashboard.

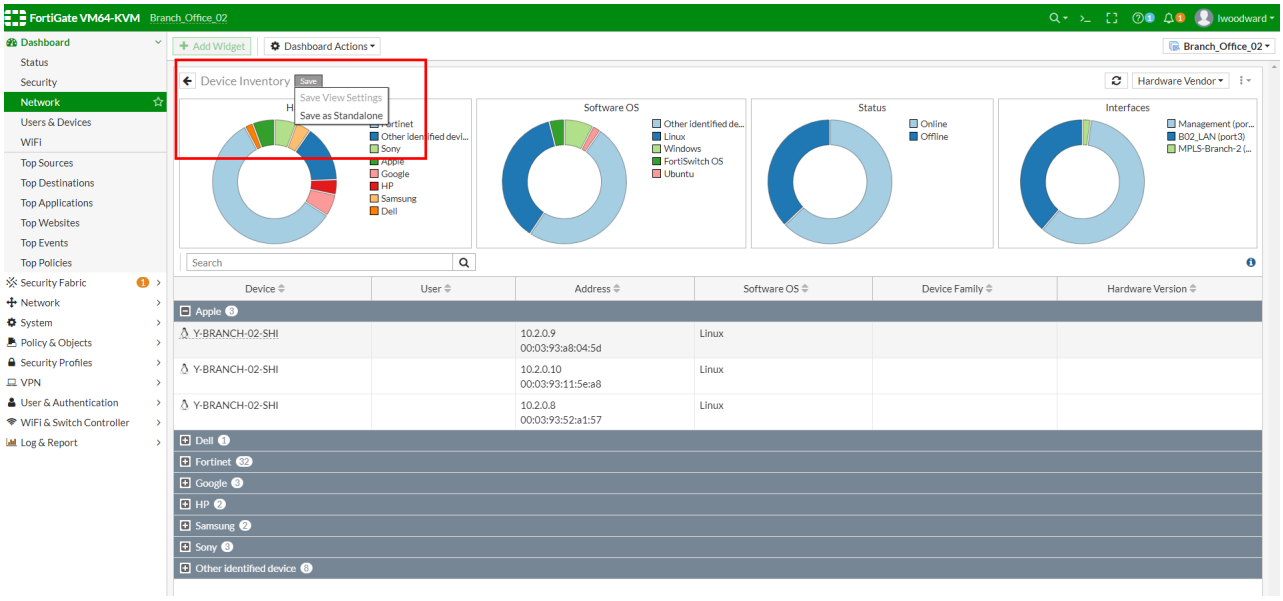


To create a dashboard with a widget:

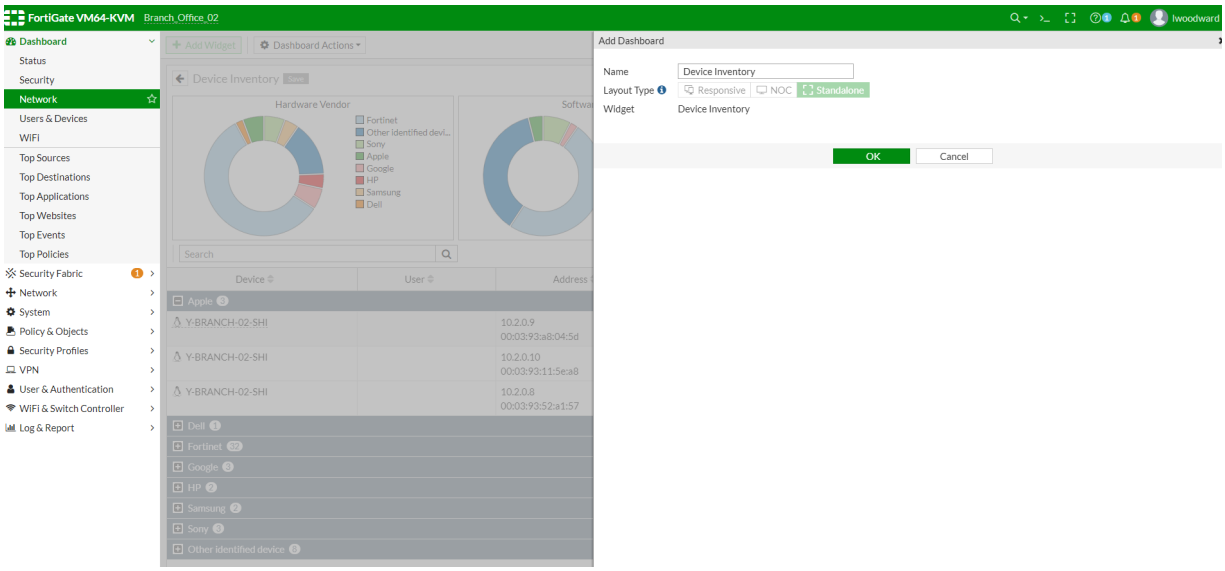
1. Hover over a widget in the dashboard, and click *Expand to Full Screen*.



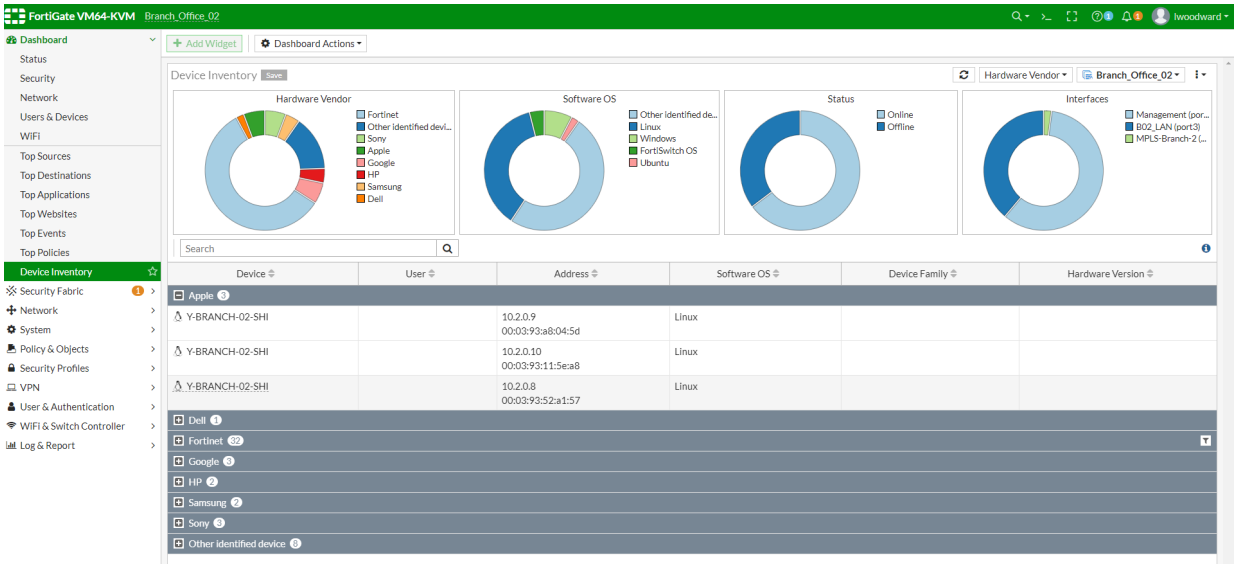
2. In the widget banner, click *Save* > *Save as Standalone*.



3. In the *Add Widget* window, click *OK*.

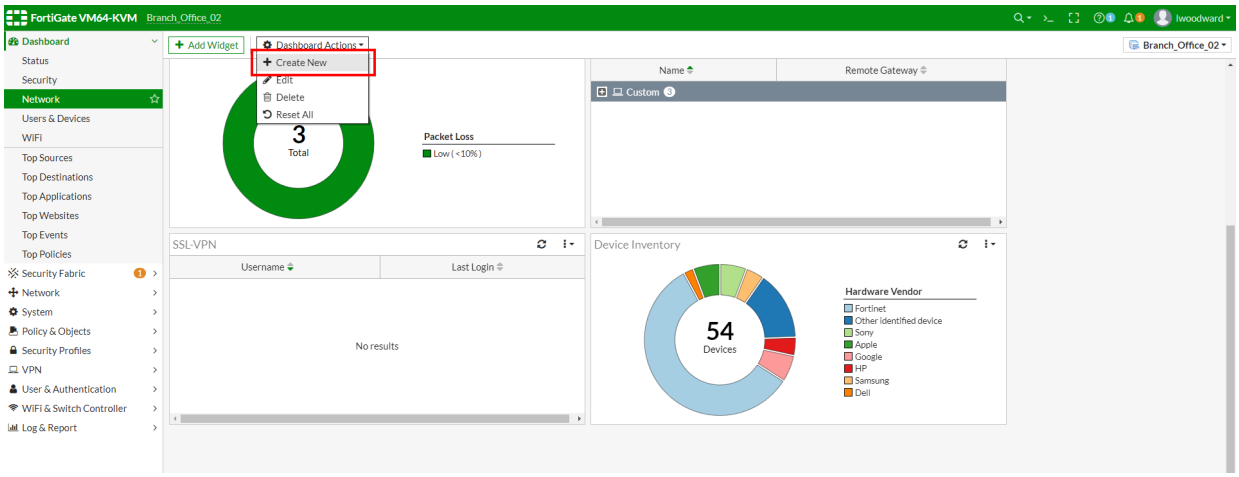


The new dashboard is added to the tree menu.

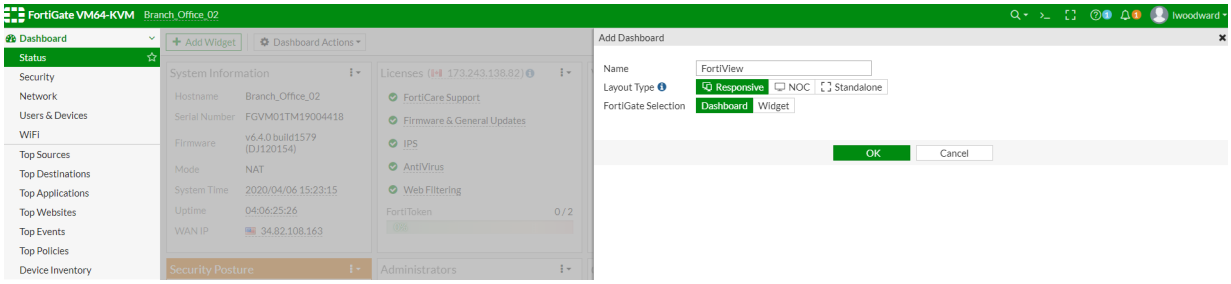


To create a new custom dashboard:

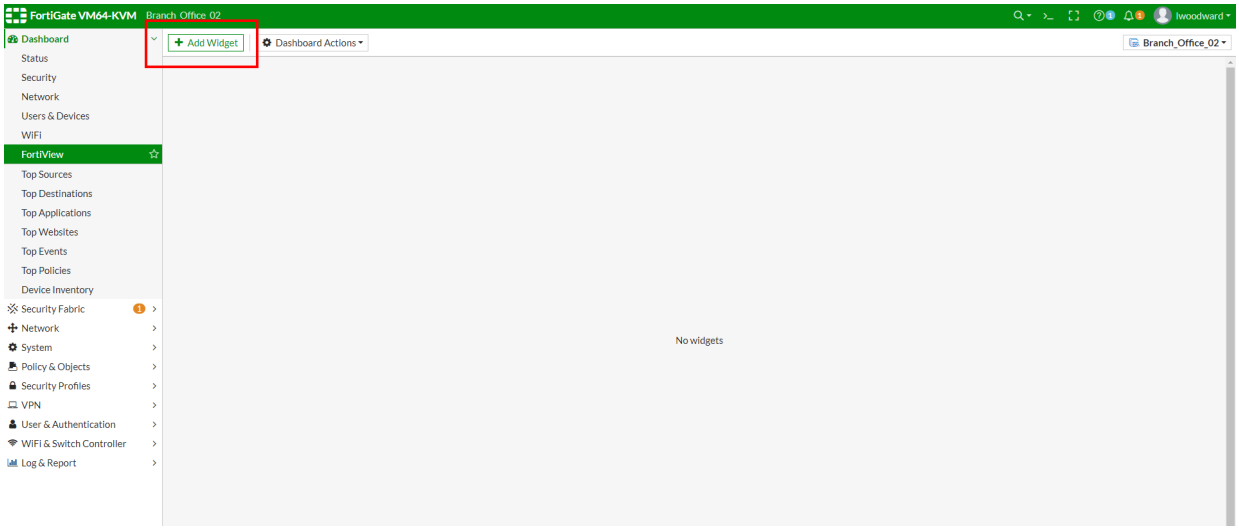
1. Open a dashboard, and click *Dashboard Actions* > *Create New*.



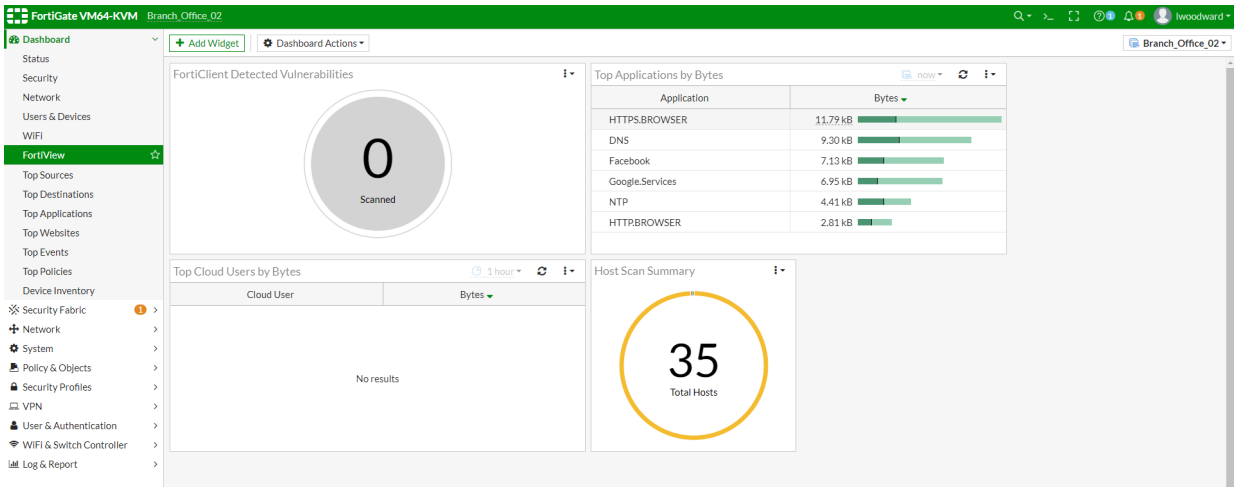
2. Configure the dashboard options, and click *OK*. The dashboard is added to the tree menu.



3. Click *Add Widget* and select the widgets you want to add to the new dashboard. Click *Close* when you are done.



The widgets are added to the dashboard.



Using the root FortiGate with disk to store historic user and device information

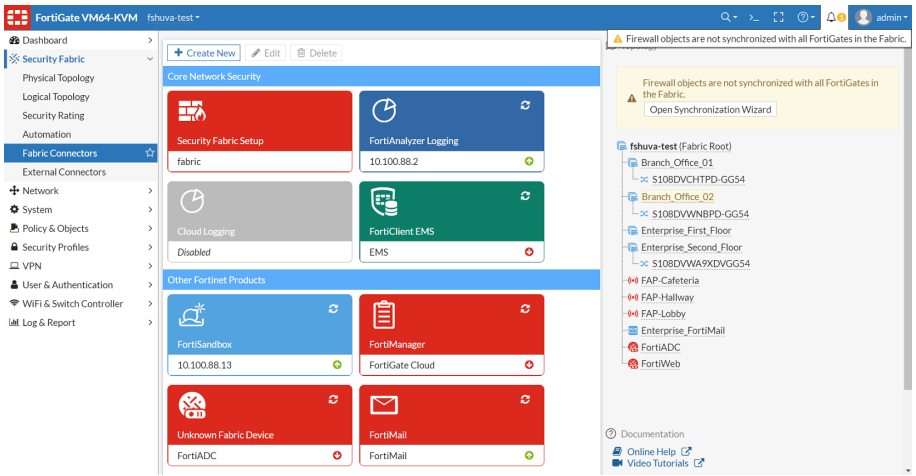
This backend implementation allows the root FortiGate in a Security Fabric to store historic user and device information in a database on its disk. This will allow administrators to visualize users and devices over a period of time.

A new daemon, user_info_history, stores this data on the disk. The information source for the historical data will be the user_info daemon, which would be recorded on the disk when user_info notifies user_info_history that a user has logged out or the device is no longer connected.

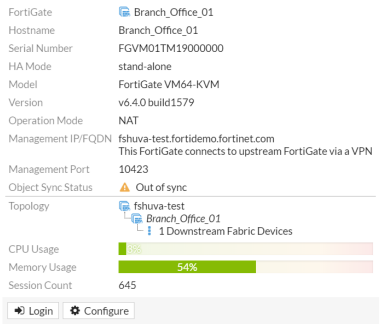
Synchronizing objects across the Security Fabric

When the Security Fabric is enabled, various objects such as addresses, services, and schedules are synced from the upstream FortiGate to all downstream devices by default. The firewall object synchronization wizard helps identify objects that are out of sync and resolves any conflicts. Objects that are out of sync are highlighted in yellow in the GUI.

In this example, the notifications icon displays a message that *Firewall objects are not synchronized with all the FortiGates in the Fabric*. In the topology tree, *Branch_Office_02* is highlighted in yellow because it is out of sync.

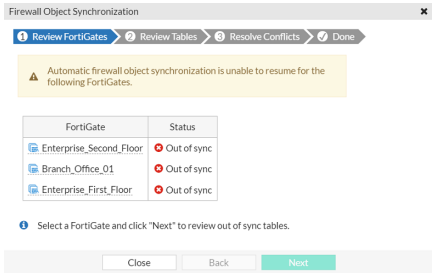


In this example, the tooltip displays a caution icon that the device is out of sync.

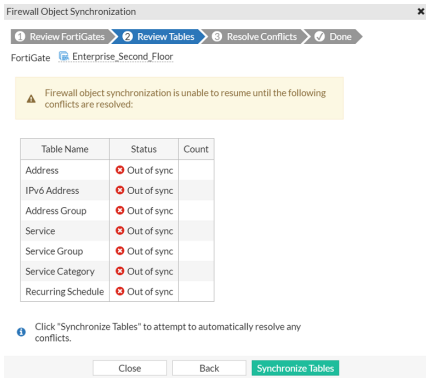


To use the firewall object synchronization wizard in the GUI:

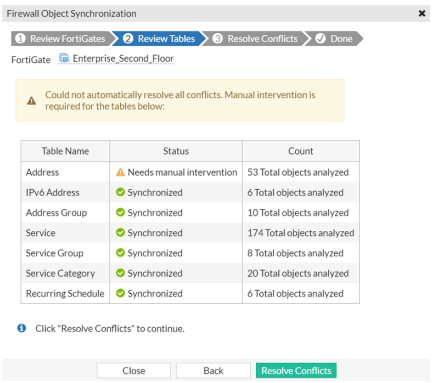
1. Go to *Security Fabric >Fabric Connectors* and click *Open Synchronization Wizard*.
   A list of FortiGates and their synchronization status displays.



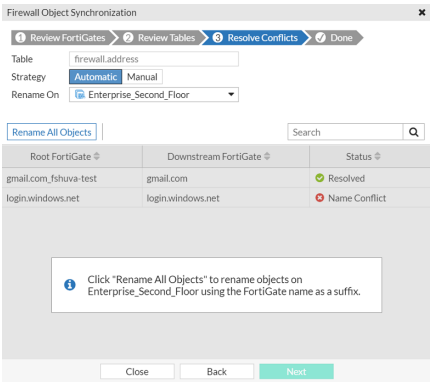
2. Select a FortiGate that is *Out of sync* and click *Next*.
   A list of tables and their synchronization status displays.
3. Click *Synchronize Tables*.



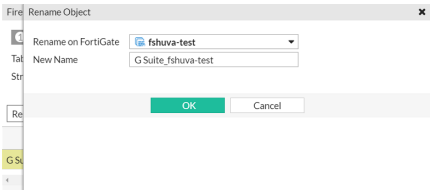
   The FortiGate attempts to automatically resolve the conflicts. In this example, the address table requires manual intervention.



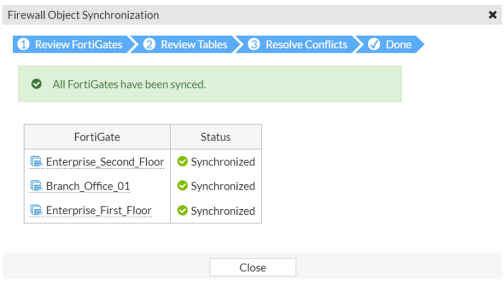
4. Click *Resolve Conflicts*.
5. For *Strategy*, choose one of the following.
   a. Automatic resolve (automatically resolves all the name conflicts and renames them on the selected FortiGate using the FortiGate name as a suffix):
      i. Click *Automatic*.
      ii. Click *Rename All Objects*.



   b. Manual resolve:
      i. Click *Manual*.
      ii. Double-click an object and re-name it.
      iii. Click *OK*.



6. Click *Next*.
   An updated list of FortiGates and their synchronization status displays.



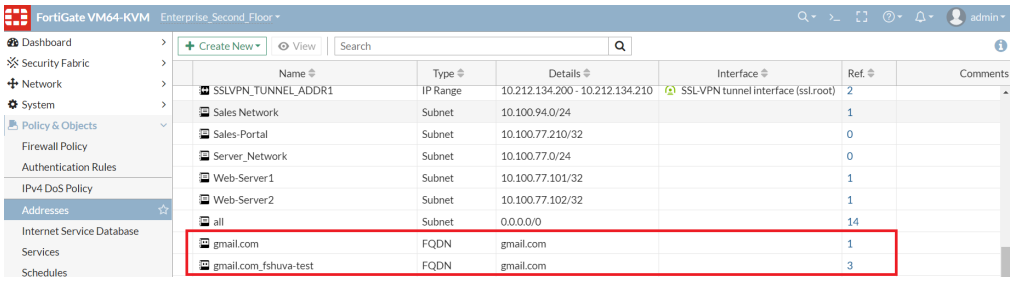
7. Click *Close*.

To verify object synchronization on downstream devices:

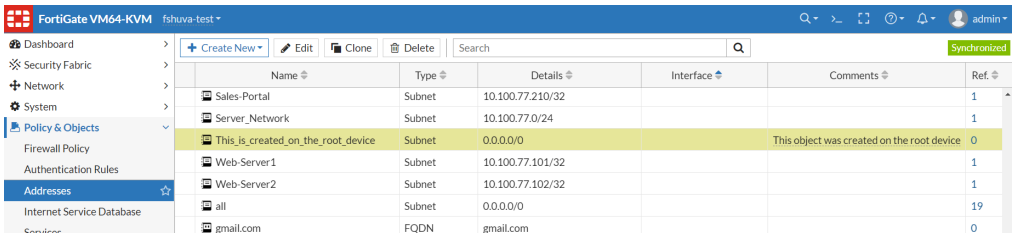
1. Log in to a downstream device.
2. Go to *Policy & Objects* > *Addresses*.
   - An information bubble displays the following: *All objects must be created/edited on the root FortiGate, and will be read-only on downstream FortiGates*.



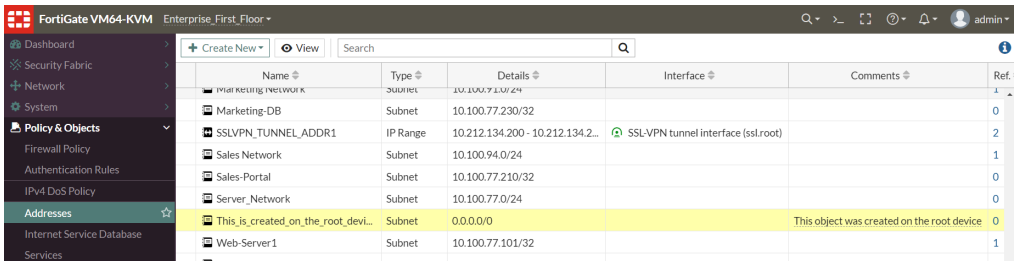
   - The following example shows an object that exists on both upstream (Enterprise_Second_Floor) and downstream (fshuva-test) FortiGates. On the downstream device, there is an existing *gmail.com*, and another object, *gmail.com_fshuva-test*, that was resolved by adding the suffix of the upstream FortiGate name to the end.



   - In this example, an object created on the upstream FortiGate is synchronized to a downstream FortiGate.



     The same object appears automatically on the downstream device.



CLI commands

Object synchronization can be configured with the following commands:

```
config system csf
    set fabric-object-unification [default | local]
    set configuration-sync [default | local]
    ...
    next
end
```

Parameter	Description
fabric-object-unification	*default*: Global CMDB objects will be synchronized in Security Fabric. *local*: Global CMDB objects will not be synchronized to and from this device.
configuration-sync	*default*: Synchronize configuration for FortiAnalyzer, FortiSandbox, and Central Management to root node. *local*: Do not synchronize configuration with root node.

Other

This section lists other new features added to FortiOS:

- IPv6 on page 213
- Set minimum RIP update timer to one second on page 228
- Redirect to WAD after handshake completion on page 220
- No session timeout on page 220
- Authentication support for upstream proxy in transparent proxy mode on page 224
- FortiOS image signing and verification on page 252
- ICAP response filtering on page 254
- UUID field added to all policy types on page 222
- Use CP9/SoC3 entropy source on page 224
- SNMP bridge MIB module support on page 225
- Support SHA-2 for SNMPv3 on page 227
- Dynamic address support for SSL VPN policies on page 228
- SNAT support for policies with virtual wire pairs on page 237
- Increase in maximum number of VIP real servers on page 239
- GUI support for FortiLink groups on page 239
- FortiSwitch link status visibility improvements on page 244
- WAN interface bandwidth log on page 240
- Source interface setting for NetFlow data on page 241
- Maintain radio SSID WLAN IDs on page 247
- Support for Okta RADIUS attributes filter-Id and class on page 250
- Support for FAP431F and FAP433F on page 256
- Enhanced autoscale clusters for FortiGate VM on page 259
- SNMP traps and query for monitoring DHCP pool on page 261
- Firmware upgrade notifications on page 262
- Identify the XAUI link used for a specific traffic stream on page 263
- DHCP client options on page 264
- NAS-IP support per SSL-VPN realm on page 265
- Matching multiple parameters on application control signatures on page 267
- Detecting IEC 61850 MMS protocol in IPS on page 269
- IP address tooltips on page 271
- Interface-based traffic shaping on page 273

IPv6

This section lists other new features added to FortiOS related to protocols.

- IPv6 geography-based address support on page 214
- Support for IPv6 in central SNAT table on page 216

- FQDN support for remote gateways on page 218

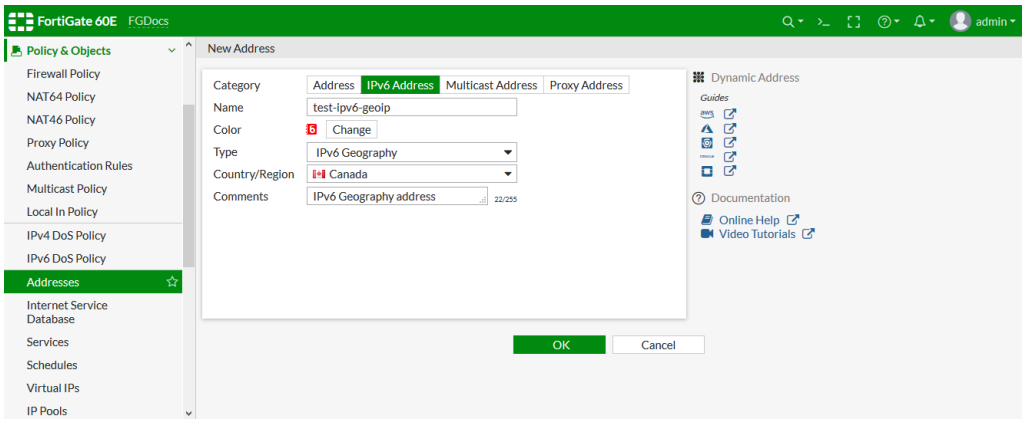
IPv6 geography-based address support

Geography-based IPv6 addresses can be created and applied to IPv6 firewall policies.

IPv6 geography-based addresses do not support `geoip-override` or `geoip-anycast`.

To create an IPv6 geography-based address in the GUI:

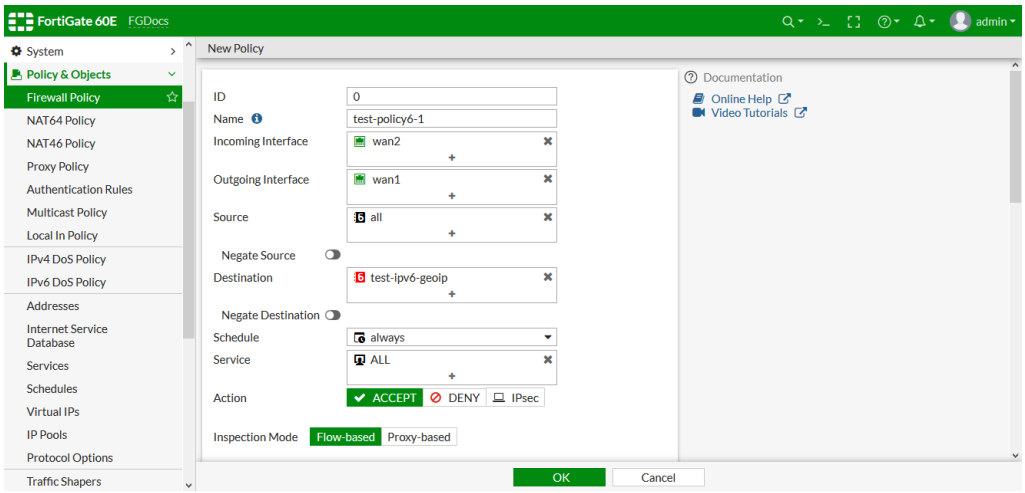
1. Go to *Policy and Objects* > *Addresses*.
2. Click *Create New* > *Address*.
3. Set *Category* to *IPv6 Address*.
4. Enter a name for the address.
5. Set *Type* to *IPv6 Geography*.
6. Select the *Country/Region* from the list.
7. Optionally, enter comments.



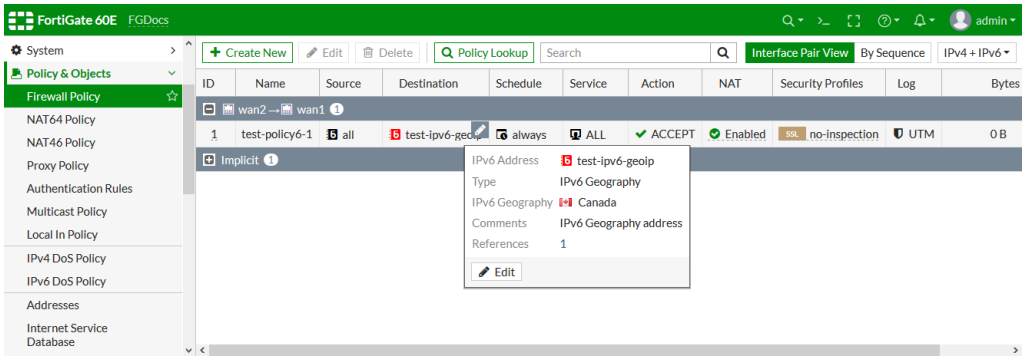
8. Click *OK*.

To use the IPv6 geography address in a policy:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Edit an existing policy, or create a new one, using the IPv6 geography address as the *Source* or *Destination Address*.



3. In the policy list, hover over the address to view details.



To configure an IPv6 geography-based address in the CLI:

1. Create an IPv6 geography-based address:

```
config firewall address6
    edit "test-ipv6-geoip"
        set type geography
        set color 6
        set comment "IPv6 Geography address"
        set country "CA"
    next
end
```

2. Use the IPv6 geography-based address in a policy:

```
config firewall policy
    edit 1
        set name "test-policy6-1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr6 "all"
        set dstaddr6 "test-ipv6-geoip"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
```

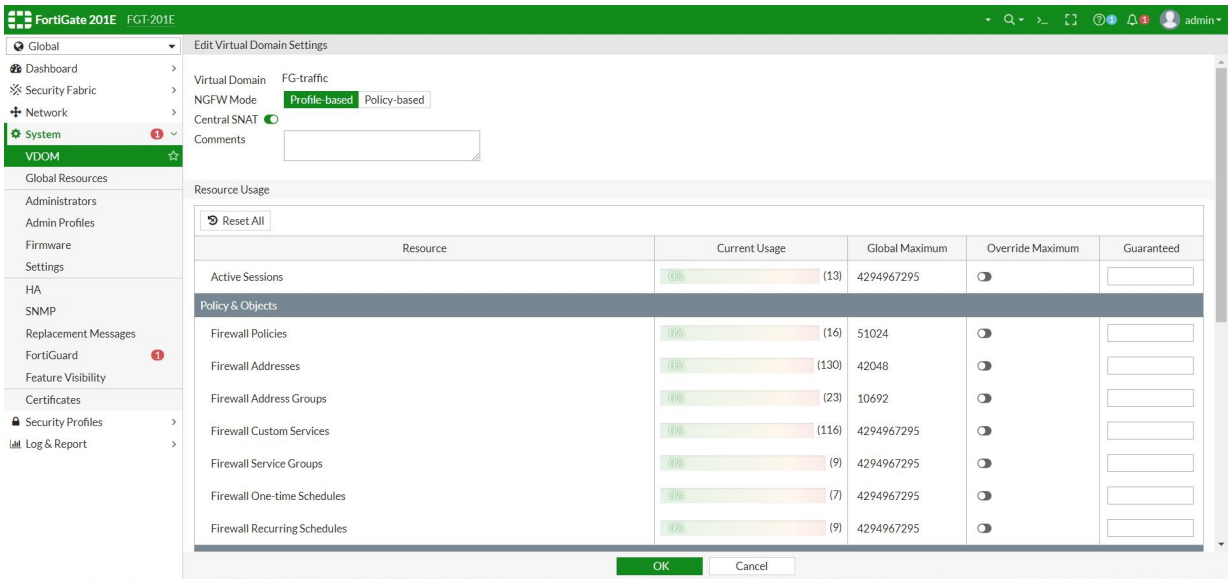
```
    next
end
```

Support for IPv6 in central SNAT table

IPv4 and IPv6 central SNAT maps are displayed in the same table.

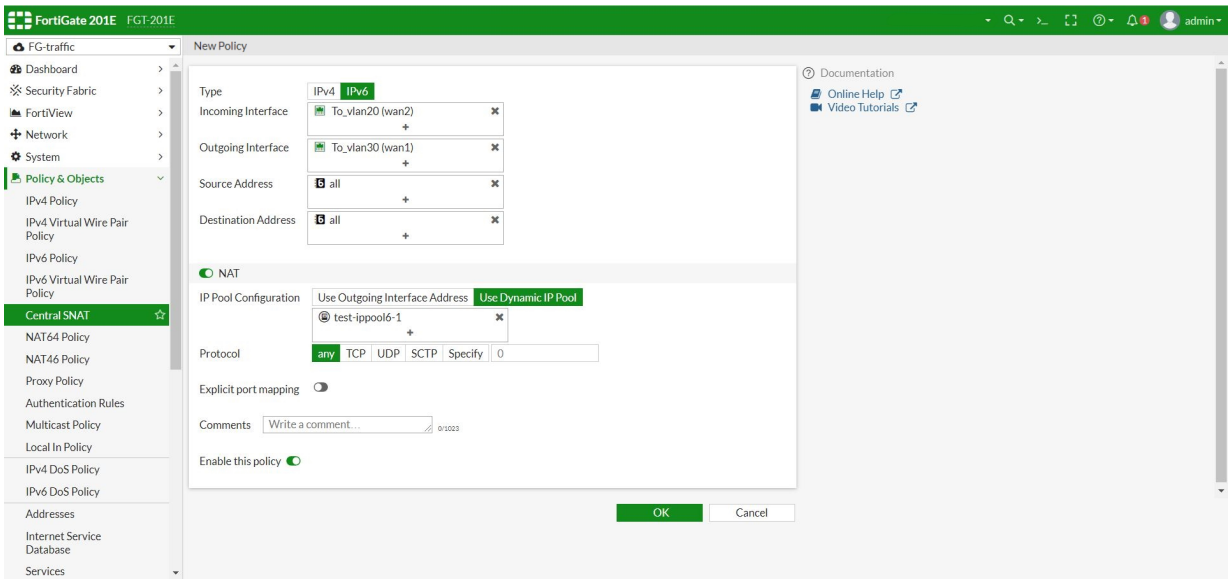
To configure an IPv6 policy with central SNAT in the GUI:

1. Enable central SNAT:
   a. In the Global VDOM, go to *System* > *VDOM*.
   b. Select a VDOM and click *Edit*. The *Edit Virtual Domain Settings* pane opens.
   c. Enable *Central SNAT*.
   d. Click *OK*.



2. Go in to the VDOM with central SNAT enabled (FG-traffic in this example).
3. Go *Policy & Objects* > *Central SNAT* and click *Create New*.
4. Configure the policy settings:
   a. For *Type*, select *IPv6*.
   b. Enter the interface, address, and IP pool information.
   c. Configure the other settings as needed.

d. Click *OK*.



The matching SNAT traffic will be handled by the IPv6 central SNAT map.

To configure an IPv6 policy with central SNAT in the CLI:

1. Enable central SNAT:

```
config vdom
    edit FG-traffic
        config system settings
            set central-nat enable
        end
    next
end
```

2. Create an IPv6 central SNAT policy:

```
config vdom
    edit FG-traffic
        config firewall central-snat-map
            edit 2
                set type ipv6
                set srcintf "wan2"
                set dstintf "wan1"
                set orig-addr6 "all"
                set dst-addr6 "all"
                set nat-ippool6 "test-ippool6-1"
            next
        end
    next
end
```

3. Verify the SNAT traffic:

```
(FG-traffic) # diagnose sniffer packet any icmp6 4
interfaces=[any]
filters=[icmp6]
3.602891 wan2 in 2000:10:1:100::41 -> 2000:172:16:200::55: icmp6: echo request seq 0
```

```
3.602942 wan1 out 2000:172:16:200::199 -> 2000:172:16:200::55: icmp6: echo request seq 0
3.603236 wan1 in 2000:172:16:200::55 -> 2000:172:16:200::199: icmp6: echo reply seq 0
3.603249 wan2 out 2000:172:16:200::55 -> 2000:10:1:100::41: icmp6: echo reply seq 0
4.602559 wan2 in 2000:10:1:100::41 -> 2000:172:16:200::55: icmp6: echo request seq 1
4.602575 wan1 out 2000:172:16:200::199 -> 2000:172:16:200::55: icmp6: echo request seq 1
4.602956 wan1 in 2000:172:16:200::55 -> 2000:172:16:200::199: icmp6: echo reply seq 1
4.602964 wan2 out 2000:172:16:200::55 -> 2000:10:1:100::41: icmp6: echo reply seq 1
^C
8 packets received by filter
0 packets dropped by kernel
```

FQDN support for remote gateways

FortiGate supports FQDN when defining an IPsec remote gateway with a dynamically assigned IPv6 address. When FortiGate attempts to connect to the IPv6 device, FQDN will resolve the IPv6 address even when the address changes.

Using FQDN to configure the remote gateway is useful when the remote end has a dynamic IPv6 address assigned by their ISP or DHCPv6 server.

1. Set the VPN to DDNS and configure FQDN

```
config vpn ipsec phase1-interface
   edit "ddns6"
      set type ddns
      set interface "agg1"
      set ip-version 6
      set ike-version 2
      set peertype any
      set net-device disable
      set proposal aes128-sha256 aes256-sha256 aes128gcm-prfsha256 aes256gcm-prfsha384
           chacha20poly1305-prfsha256
      set dpd on-idle
      set remotegw-ddns "rgwa61.vpnlab.org"
      set psksecret xxxxxxx
   next
end
config vpn ipsec phase2-interface
   edit "ddns6"
      set phase1name "ddns6"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
           chacha20poly1305
      set src-addr-type subnet6
      set dst-addr-type subnet6
      set src-subnet6 2003:1:1:1::/64
   next
end
```

2. FQDN resolves the IPv6 address

```
# diagnose test application dnsproxy 7
vfid=0, name=rgwa61.vpnlab.org, ttl=3600:3547:1747
2003:33:1:1::22 (ttl=3600)
```

3. FortiGate uses FQDN to connect to the IPv6 device

```
# diagnose vpn tunnel list name ddns6
list ipsec tunnel by names in vd 0

------------------------------------------------

name=ddns6 ver=2 serial=2 2003:33:1:1::1:0->2003:33:1:1::22:0 dst_mtu=1500
bound_if=32 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/520 options[0208]=npu frag-rfc
     run_state=0 accept_traffic=1 overlay_id=0
proxyid_num=1 child_num=0 refcnt=10 ilast=9 olast=9 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=72340
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=ddns6 proto=0 sa=1 ref=2 serial=1
   src: 0:2003:1:1:1::/64:0
   dst: 0:::/0:0
   SA: ref=3 options=10226 type=00 soft=0 mtu=1422 expire=42680/0B replaywin=2048
      seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0 hash_search_len=1
   life: type=01 bytes=0/0 timeout=42901/43200
   dec: spi=ac7a5718 esp=aes key=16 9976b66280cc49f500d8edca093e03fb
      ah=sha1 key=20 4d94d76fc18df5a180c52e0a6cd5f430fde48fe8
   enc: spi=7ab888ec esp=aes key=16 841a95d3ee5ea5108a2ba269b74998d1
      ah=sha1 key=20 ed0b52d27776e30149ee36af4fd4626681c2a3a1
   dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
   npu_flag=00 npu_rgwy=2003:33:1:1::22 npu_lgwy=2003:33:1:1::1 npu_selid=0 dec_npuid=0 enc_
        npuid=0
run_tally=1
```

4. The tunnel can still connect to the FQDN address when the IPv6 address changes

```
# diagnose debug application ike -1
# diagnose debug enable
ike 0:ddns6: set oper down
ike 0:ddns6: carrier down
ike shrank heap by 159744 bytes
ike 0: cache rebuild start
ike 0:ddns6: sending DNS request for remote peer rgwa61.vpnlab.org
ike 0: send IPv6 DNS query : rgwa61.vpnlab.org
ike 0: cache rebuild done
ike 0:ddns6: remote IPv6 DDNS gateway is empty, retry to resolve it
ike 0: DNS response received for remote gateway rgwa61.vpnlab.org
ike 0: DNS rgwa61.vpnlab.org -> 2003:33:1:1::33
ike 2:test:46932: could not send IKE Packet(P1_RETRANSMIT):50.1.1.1:500->50.1.1.2:500,
     len=716: error 101:Network is unreachable
ike 0:ddns6: remote IPv6 DDNS gateway is empty, retry to resolve it
ike 0:ddns6: 'rgwa61.vpnlab.org' resolved to 2003:33:1:1::33
ike 0: cache rebuild start
ike 0:ddns6: local:2003:33:1:1::1, remote:2003:33:1:1::33
ike 0:ddns6: cached as static-ddns.
ike 0: cache rebuild done
ike 0:ddns6: auto-negotiate connection
ike 0:ddns6: created connection: 0x155aa510 32 2003:33:1:1::1->2003:33:1:1::33:500.

.....................................................................................................

ike 0:ddns6:46933:ddn6:47779: add IPsec SA: SPIs=ac7a5719/7ab888ed
ike 0:ddns6:46933:ddn6:47779: IPsec SA dec spi ac7a5719 key
     16:0F27F1D1D02496F90D15A30E2C032678 auth 20:46564E0E86A054374B31E58F95E4458340121BCE
```

```
ike 0:ddns6:46933:ddn6:47779: IPsec SA enc spi 7ab888ed key
     16:926B12908EE670E1A5DDA6AD8E96607B auth 20:42BF438DC90867B837B0490EAB08E329AB62CBE3
ike 0:ddns6:46933:ddn6:47779: added IPsec SA: SPIs=ac7a5719/7ab888ed
ike 0:ddns6:46933:ddn6:47779: sending SNMP tunnel UP trap
ike 0:ddns6: carrier up
```

Redirect to WAD after handshake completion

In a proxy-based policy, the TCP connection is proxied by the FortiGate. A TCP 3-way handshake can be established with the client even though the server did not complete the handshake.

This option uses IPS to handle the initial TCP 3-way handshake. It rebuilds the sockets and redirects the session back to proxy only when the handshake with the server is established.

To enable proxy after a TCP handshake in an SSL/SSH profile:

```
config firewall ssl-ssh-profile
    edit "test"
        config https
            set ports 443
            set status certificate-inspection
            set proxy-after-tcp-handshake enable
        end
        .....
   next
end
```

To enable proxy after a TCP handshake in protocol options:

```
config firewall profile-protocol-options
    edit "test"
        config http
            set ports 80
            set proxy-after-tcp-handshake enable
            unset options
            unset post-lang
        end
        ....
   next
end
```

No session timeout

To allow clients to permanently connect with legacy medical applications and systems that do not have keepalive or auto-reconnect features, the session timeout can be set to never for firewall services, policies, and VDOMs.

The options to disable session timeout are hidden in the CLI.

To set the session TTL value of a custom service to never:

```
config firewall service custom
    edit "tcp_23"
        set tcp-portrange 23
        set session-ttl never
    next
end
```

To set the session TTL value of a policy to never:

```
config firewall policy
    edit 201
        set srcintf "wan1"
        set dstintf "wan2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "TCP_8080"
        set logtraffic disable
        set session-ttl never
        set nat enable
    next
end
```

To set the session TTL value of a VDOM to never:

```
config system session-ttl
    set default never
    config port
        edit 1
            set protocol 6
            set timeout never
            set start-port 8080
            set end-port 8080
        next
    end
end
```

To view a session list with the timeout set to never:

```
# diagnose sys session list

session info: proto=6 proto_state=01 duration=9 expire=never timeout=never flags=00000000 sock-
flag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty f00
statistic(bytes/packets/allow_err): org=2290/42/1 reply=2895/34/1 tuples=2
tx speed(Bps/kbps): 238/1 rx speed(Bps/kbps): 301/2
orgin->sink: org pre->post, reply pre->post dev=18->17/17->18 gwy=172.16.200.55/10.1.100.41
hook=post dir=org act=snat 10.1.100.41:34256->172.16.200.55:23(172.16.200.10:34256)
```

```
hook=pre dir=reply act=dnat 172.16.200.55:23->172.16.200.10:34256(10.1.100.41:34256)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=9 auth_info=0 chk_client_info=0 vd=1
serial=00000b27 tos=ff/ff app_list=0 app=0 url_cat=0
rpdb_link_id = 00000000 ngfwid=n/a
dd_type=0 dd_mode=0
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
total session 1
```

UUID field added to all policy types

The UUID field has been added to all policy types, including multicast, local-in (IPv4 and IPv6), and central SNAT policies. UUIDs are automatically generated by FortiOS when the policy is created and can be viewed in the CLI using the `show` command.

A comments field has also been added for multicast policies.

To view the UUID for a multicast policy:

1. Create a policy:

```
config firewall multicast-policy
    edit 1
        set comments "multicast-policy-1"
        set logtraffic enable
        set srcintf "wan1"
        set dstintf "wan2"
        set srcaddr "all"
        set dstaddr "230-0-0-1" "test-multicast-addr-1"
        set snat enable
        set snat-ip 10.1.100.188
        set dnat 229.1.2.19
        set auto-asic-offload disable
    next
end
```

2. Use the `show` command to see the UUID:

```
# show firewall multicast-policy
config firewall multicast-policy
    edit 1
        set uuid d0f74f64-fc41-51e9-2dfc-729f027e9979
        set comments "multicast-policy-1"
        set logtraffic enable
        set srcintf "wan1"
        set dstintf "wan2"
        set srcaddr "all"
        set dstaddr "230-0-0-1" "test-multicast-addr-1"
        set snat enable
        set snat-ip 10.1.100.188
        set dnat 229.1.2.19
        set auto-asic-offload disable
```

```
    next
end
```

To view the UUID for an IPv4 or IPv6 local-in policy:

1. Create a policy:

```
config firewall local-in-policy
    edit 1
        set intf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set service "PING"
        set schedule "always"
        set comments "test-1"
    next
end
```

2. Use the `show` command to see the UUID:

```
# show firewall local-in-policy
config firewall local-in-policy
    edit 1
        set uuid 1aeb7d98-0016-51ea-7913-b6d62f4409cd
        set intf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set service "PING"
        set schedule "always"
        set comments "test-1"
    next
end
```

To view the UUID for a central SNAT policy:

1. Create a policy:

```
config firewall central-snat-map
    edit 1
        set srcintf "wan2"
        set dstintf "wan1"
        set orig-addr "all"
        set dst-addr "all"
        set orig-port 11111
        set nat-ippool "Overload-ippool-1"
        set nat-port 22222
    next
end
```

2. Use the `show` command to see the UUID:

```
# show firewall central-snat-map
config firewall central-snat-map
    edit 1
        set uuid d0f87af6-fc41-51e9-ef72-32f8655f8008
        set srcintf "wan2"
```

```
            set dstintf "wan1"
            set orig-addr "all"
            set dst-addr "all"
            set orig-port 11111
            set nat-ippool "Overload-ippool-1"
            set nat-port 22222
        next
    end
```

Use CP9/SoC3 entropy source

FortiGate models that use FortiASIC CP9 and SoC3 chips periodically reseed a PRNG (Pseudo-Random Number Generator) in normal (non-CC/FIPS) mode.

Authentication support for upstream proxy in transparent proxy mode

A downstream proxy FortiGate that needs to be authenticated by the upstream web proxy can use the basic authentication method to send its username and password, in the base64 format, to the upstream web proxy for authentication. If the authentication succeeds, web traffic that is forwarded from the downstream proxy FortiGate to the upstream proxy can be accepted and forwarded to its destinations.

In this example, a school has a FortiGate acting as a downstream proxy that is configured with firewall policies for each user group: students, and staff. In each policy, a forwarding server is configured to forward the web traffic to the upstream web proxy.

The username and password that the upstream web proxy uses to authenticate the downstream proxy are configured on the forwarding server, and are sent to the upstream web proxy with the forwarded HTTP requests.

	Username	Password
student.proxy.local:8080	students	ABC123
staff.proxy.local:8081	staff	123456

On the downstream FortiGate, configure forwarding servers with the usernames and passwords for authentication on the upstream web proxy, then apply those servers to firewall policies for transparent proxy. For explicit web proxy, the forwarding servers can be applied to proxy policies.

When the transparent proxy is configured, clients can access websites without configuring a web proxy in their browser. The downstream proxy sends the username and password to the upstream proxy with forwarded HTTP requests to be authenticated.

To configure the forwarding server on the downstream FortiGate:

```
config web-proxy forward-server
    edit "Student_Upstream_WebProxy"
        set addr-type fqdn
```

```
        set fqdn "student.proxy.local"
        set port 8080
        set username "student"
        set password ABC123
    next
    edit "Staff_Upstream_WebProxy"
        set addr-type fqdn
        set fqdn "staff.proxy.local"
        set port 8081
        set username "staff"
        set password 123456
    next
end
```

To configure firewall policies for transparent proxy:

```
config firewall policy
    edit 1
        set srcintf "Vlan_Student"
        set dstintf "port9"
        set srcaddr "Student_Subnet"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
        set webproxy-forward-server "Student_Upstream_WebProxy"
        set nat enable
    next
    edit 2
        set srcintf "Vlan_Staff"
        set dstintf "port9"
        set srcaddr "Staff_Subnet"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
        set webproxy-forward-server "Staff_Upstream_WebProxy"
        set nat enable
    next
end
```

SNMP bridge MIB module support



This feature is only available on FortiGate Rugged 30D, which supports 802.1p.

SNMP bridge MIB module support is available on FortiGates with 802.1p to monitor STP activity.

The following OIDs have been added:

Object name	OID
dot1dBridge.dot1dBase.dot1dBaseBridgeAddress	1.3.6.1.2.1.17.1.1
dot1dBridge.dot1dBase.dot1dBaseNumPorts	1.3.6.1.2.1.17.1.2
dot1dBridge.dot1dBase.Type	1.3.6.1.2.1.17.1.3
dot1dBridge.dot1dBase.dot1dBasePortEntry.dot1dBasePortIfIndex	1.3.6.1.2.1.17.1.4.1.2
dot1dBridge.dot1dBase.dot1dBasePortEntry.dot1dBasePortCircuit	1.3.6.1.2.1.17.1.4.1.3
dot1dBridge.dot1dBase.dot1dBasePortEntry.dot1dBasePortDelayExceededDiscards	1.3.6.1.2.1.17.1.4.1.5
dot1dBridge.dot1dBase.dot1dBasePortEntry.dot1dBasePortMtuExceededDiscards	1.3.6.1.2.1.17.1.4.1.5
dot1dBridge.dot1dStp.dot1dStpProtocolSpecification	1.3.6.1.2.1.17.2.1
dot1dBridge.dot1dStp.dot1dStpPriority	1.3.6.1.2.1.17.2.2
dot1dBridge.dot1dStp.dot1dStpDesignatedRoot	1.3.6.1.2.1.17.2.5
dot1dBridge.dot1dStp.dot1dStpRootCost	1.3.6.1.2.1.17.2.6
dot1dBridge.dot1dStp.dot1dStpRootPort	1.3.6.1.2.1.17.2.7
dot1dBridge.dot1dStp.dot1dStpMaxAge	1.3.6.1.2.1.17.2.8
dot1dBridge.dot1dStp.dot1dStpHelloTime	1.3.6.1.2.1.17.2.9
dot1dBridge.dot1dStp.dot1dStpForwardDelay	1.3.6.1.2.1.17.2.11
dot1dBridge.dot1dStp.dot1dStpBridgeMaxAge	1.3.6.1.2.1.17.2.12
dot1dBridge.dot1dStp.dot1dStpBridgeHelloTime	1.3.6.1.2.1.17.2.13
dot1dBridge.dot1dStp.dot1dStpBridgeForwardDelay	1.3.6.1.2.1.17.2.14
dot1dBridge.dot1dStp.dot1dStpPortEntry.dot1dStpPortPriority	1.3.6.1.2.1.17.2.15.1.2
dot1dBridge.dot1dStp.dot1dStpPortEntry.dot1dStpPortState	1.3.6.1.2.1.17.2.15.1.3
dot1dBridge.dot1dStp.dot1dStpPortEntry.dot1dStpPortEnable	1.3.6.1.2.1.17.2.15.1.4
dot1dBridge.dot1dStp.dot1dStpPortEntry.dot1dStpPortPathCost	1.3.6.1.2.1.17.2.15.1.5

To configure an SNMP bridge MIB module:

1. On the FortiGate, configure SNMP:

```
config system snmp sysinfo
    set status enable
    set description "BRIDGE_MIB"
    set contact-info "Strike Freedom"
    set location "QA LAB"
end
```

```
config system snmp community
    edit 1
        set name "REGR-SWITCH"
        config hosts
            edit 1
                set ip 172.16.200.55 255.255.255.255
            next
            edit 2
                set ip 172.18.60.149 255.255.255.255
            next
        end
        set events cpu-high mem-low log-full intf-ip vpn-tun-up vpn-tun-down ha-switch ha-
hb-failure ips-anomaly av-oversize av-fragmented fm-conf-change ha-member-up ha-member-down
av-conserve av-bypass av-oversize-blocked ips-pkg-update ips-fail-open faz-disconnect
    next
end
```

2. On the SNMP server, run `snmpwalk` on the OID from the newly added bridge MIB.
   The OID is for the bridge hello time. The SNMP server is able to query the bridge hello time from the FortiGate:

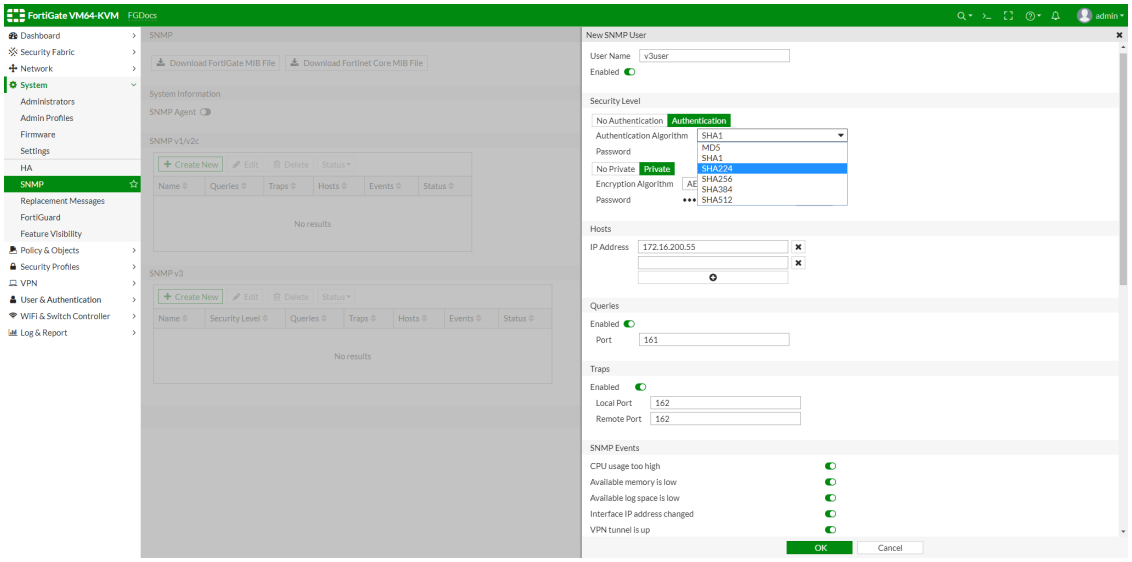
```
root@ControlPC:~# snmpwalk -v1 -c REGR-SWITCH 172.16.200.2 1.3.6.1.2.1.17.2.13
BRIDGE-MIB::dot1dStpBridgeHelloTime.0 = INTEGER: 200 centi-seconds
```

Support SHA-2 for SNMPv3

SNMPv3 supports HMAC-SHA-2 authentication protocols based on the following SHA-2 hash functions: SHA-224, SHA-256, SHA-384, and SHA-512.

To configure an SNMPv3 user in the GUI:

1. Go to *System* > *SNMP*.
2. In the *SNMPv3* section, click *Create New*. The *New SNMP User* pane opens.
3. In the *Security Level* section, click *Authentication* and for *Authentication Algorithm*, select a SHA-2 authentication protocol.



4. Configure the other settings as needed.
5. Click *OK*.

To configure an SNMPv3 user in the CLI:

```
config system snmp user
    edit "v3user"
        set security-level auth-priv
        set auth-proto {md5 | sha | sha224 | sha256 | sha384 | sha512}
        set auth-pwd xxxxxxxx
        set priv-pwd xxxxxxxx
    next
end
```

Set minimum RIP update timer to one second

The RIP update timer can be set to a minimum value of 1 second. The previous minimum timer value was 5 seconds.

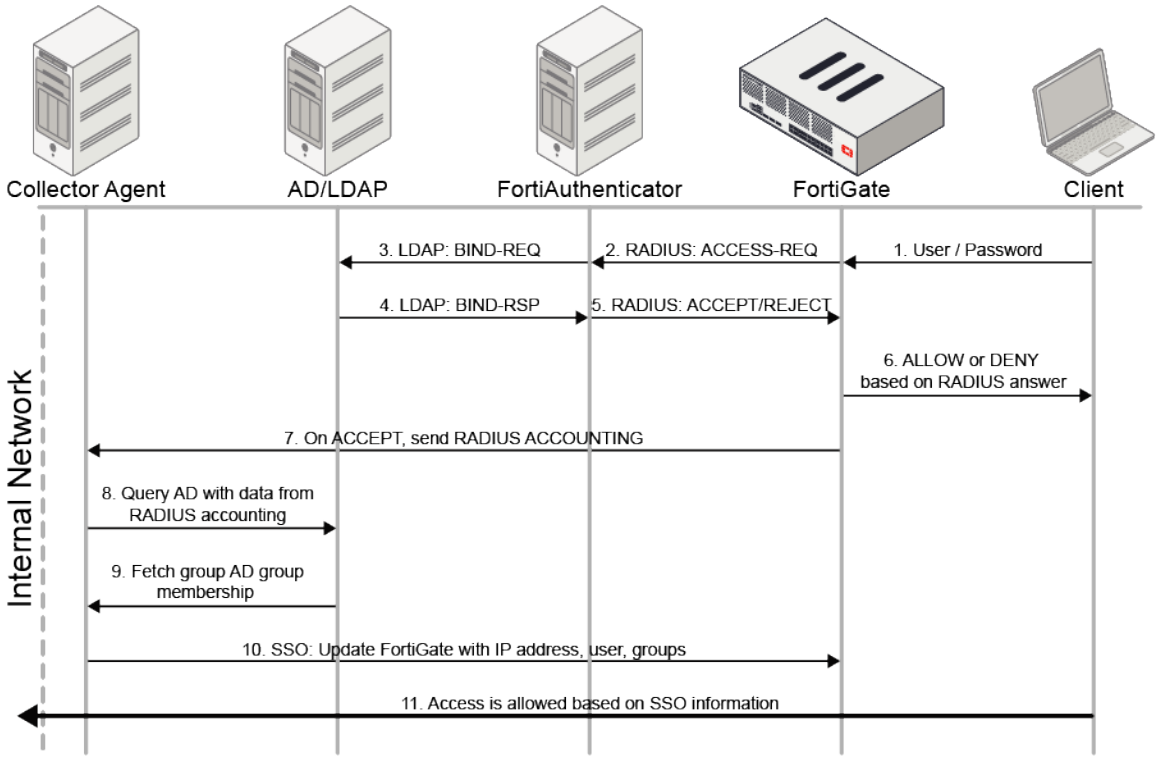
To set the RIP timer value to one second:

```
config router rip
    set update-timer 1
end
```

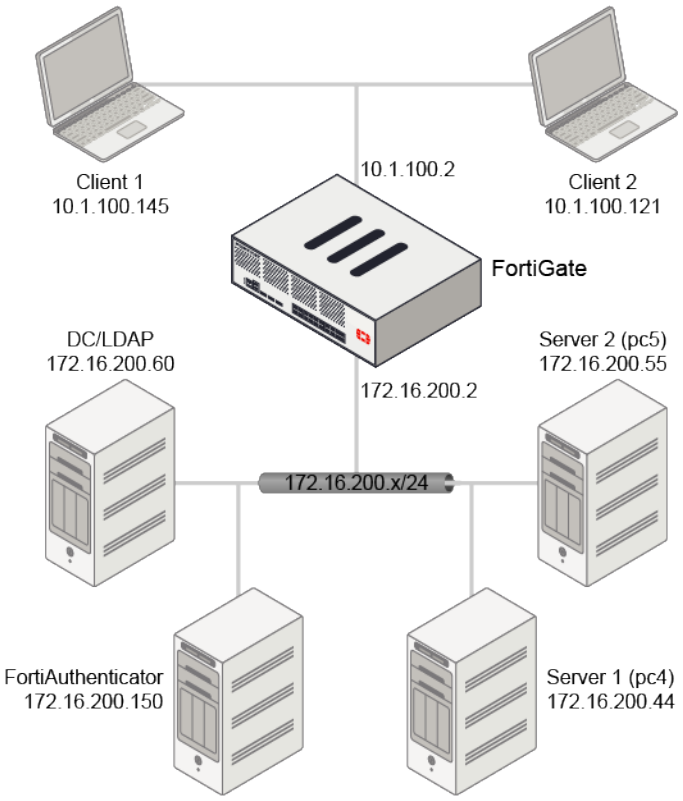
Dynamic address support for SSL VPN policies

Dynamic SSO user groups can be used in place of address objects when configuring SSL VPN policies. This allows dynamic IP addresses to be used in SSL VPN policies. A remote user group can be used for authentication while an FSSO group is separately used for authorization. Using a dummy policy for remote user authentication and a policy for FSSO group authorization, FSSO can be used with SSL VPN tunnels

This image shows the authentication and authorization flow:



In this example, FortiAuthenticator is used as a RADIUS server. It uses a remote AD/LDAP server for authentication, then returns the authentication results to the FortiGate. This allows the client to have a dynamic IP address after successful authentication.

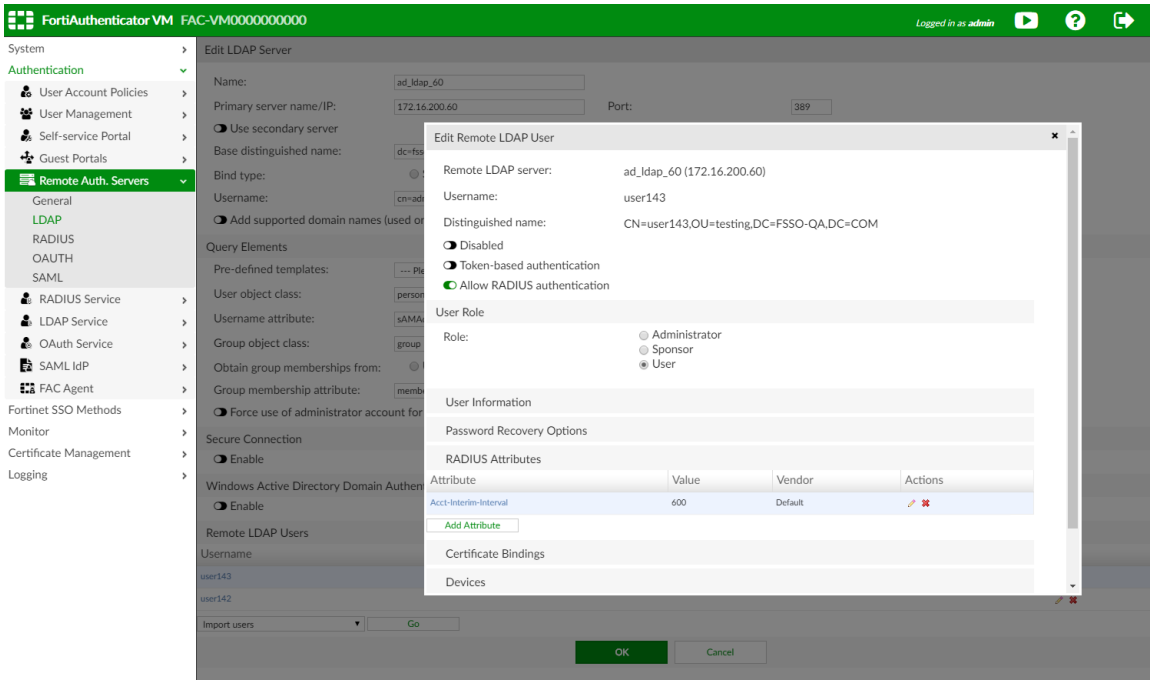


First, on the LDAP server, create two users each in their own group, *user142* in group *pc_group1*, and *user143* in group *pc_group2*.

Configure the FortiAuthenticator

To add a remote LDAP server and users on the FortiAuthenticator:

1. Go to *Authentication* > *Remote Auth. Servers* > *LDAP*.
2. Click *Create New*.
3. Set the following:
   - *Name*: *ad_ldap_60*
   - *Primary server name/IP*: *172.16.200.60*
   - *Base distinguished name*: *dc=fsso-qa,dc=com*
   - *Bind type: Regular*
   - *Username*: *cn=administrator,cn=User*
   - *Password*: <enter a password>
4. Click *OK*.
5. Edit the new LDAP server.
6. Import the remote LDAP users.
7. Edit each user to confirm that they have the RADIUS attribute *Acct-Interim-Interval*. This attribute is used by FortiGate to send interim update account messages to the RADIUS server.



To create a RADIUS client for FortiGate as a remote authentication server:

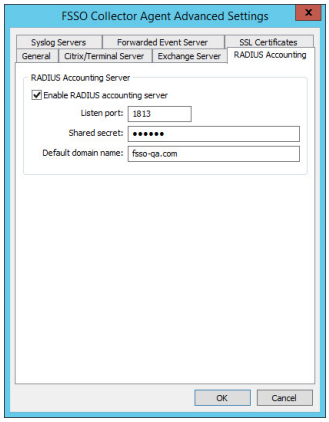
1. Go to *Authentication* > *RADIUS Service* > *Clients*.
2. Click *Create New*.

3. Set the following:
   - *Name*: *fsso_ldap*
   - *Client address*: *Range 172.16.200.1~172.16.200.10*
   - *Secret*: <enter a password>
4. In the *Realms* table, set the realm to the LDAP server that was just added: *ad_ldap_60*.
5. Click *OK*.
   FortiAuthenticator can now be used as a RADIUS server, and the authentication credentials all come from the DC/LDAP server.

Fortinet Single Sign-On Collector Agent

To configure the Fortinet Single Sign-On Collector Agent:

1. Select *Require authenticated connection from FortiGate* and enter a *Password*.
2. Click *Advanced Settings*.
3. Select the *RADIUS Accounting* tab.
4. Select *Enable RADIUS accounting server* and set the *Shared secret*.



5. Click *OK*, then click *Save&close*.
   The collector agent can now accept accounting requests from FortiGate, and retrieve the IP addresses and usernames of SSL VPN client from the FortiGate with accounting request messages.

Configure the FortiGate

To configure the FortiGate in the CLI:

1. Create a Fortinet Single Sign-On Agent fabric connector:

```
config user fsso
    edit "AD_CollectAgent"
        set server "172.16.200.60"
        set password 123456
    next
end
```

2. Add the RADIUS server:

```
config user radius
    edit "rad150"
        set server "172.16.200.150"
        set secret 123456
        set acct-interim-interval 600
        config accounting-server
            edit 1
                set status enable
                set server "172.16.200.60"
                set secret 123456
            next
        end
    next
end
```

3. Create a user group for the RADIUS server:

```
config user group
    edit "rad_group"
        set member "rad150"
    next
end
```

4. Create user groups for each of the FSSO groups:

```
config user group
    edit "fsso_group1"
        set group-type fsso-service
        set member "CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM"
    next
    edit "fsso_group2"
        set group-type fsso-service
        set member "CN=PC_GROUP2,OU=TESTING,DC=FSSO-QA,DC=COM"
    next
end
```

5. Create an SSL VPN portal and assign the RADIUS user group to it:

```
config vpn ssl web portal
    edit "testportal"
        set tunnel-mode enable
        set ipv6-tunnel-mode enable
        set web-mode enable
        ...
    next
end
config vpn ssl settings
    ...
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "rad_group"
            set portal "testportal"
        next
    end
end
```

6. Create firewall addresses:

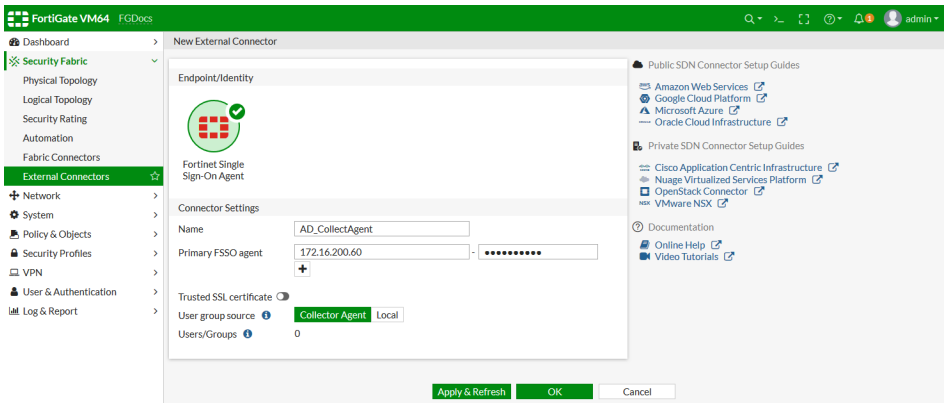
```
config firewall address
    edit "none"
        set subnet 0.0.0.0 255.255.255.255
    next
    edit "pc4"
        set subnet 172.16.200.44 255.255.255.255
    next
    edit "pc5"
        set subnet 172.16.200.55 255.255.255.255
    next
end
```

7. Create one dummy policy for authentication only, and two normal policies for authorization:

```
config firewall policy
    edit 1
        set name "sslvpn_authentication"
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "none"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set groups "rad_group"
        set nat enable
    next
    edit 3
        set name "sslvpn_authorization1"
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "pc4"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set groups "fsso_group1"
        set nat enable
    next
    edit 4
        set name "sslvpn_authorization2"
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "pc5"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set groups "fsso_group2"
        set nat enable
    next
end
```

To create an FSSO agent fabric connector in the GUI:

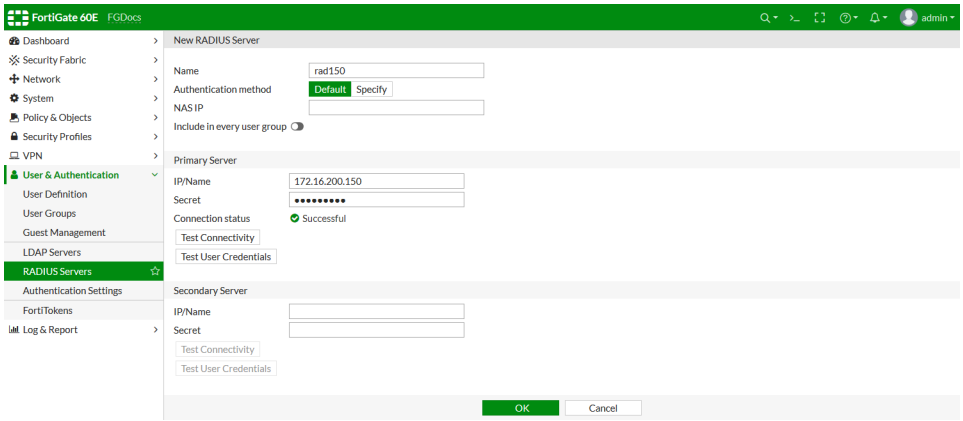
1. Go to *Security Fabric* > *External Connectors*.
2. Click *Create New*.
3. Click *Fortinet Single Sign-On Agent*.
4. Enter the name and *Primary FSSO agent* information.



5. Click *Apply & Refresh*.
   The FSSO groups are retrieved from the collector agent.

To add the RADIUS server in the GUI:

1. Go to *User & Authentication* > *RADIUS Servers*.
2. Click *Create New*.
3. Enter a name for the server.
4. Enter the *IP/Name* and *Secret* for the primary server.
5. Click *Test Connectivity* to ensure that there is a successful connection.



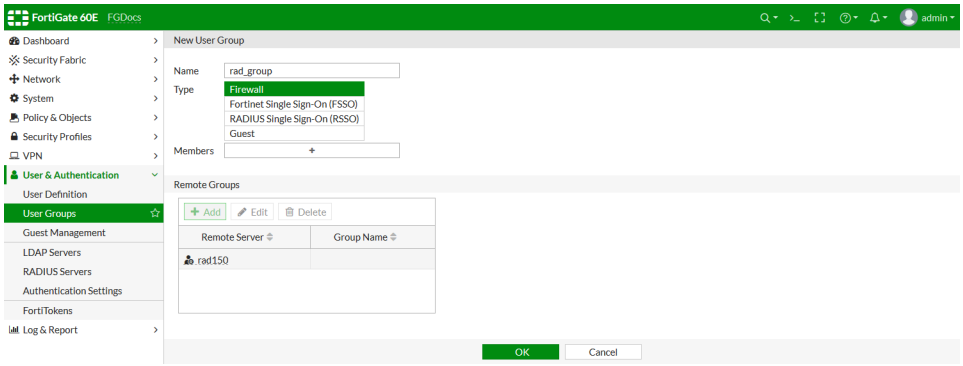
6. Click *OK*.
7. Configure an accounting server with the following CLI command:

```
config user radius
    edit rad150
        set acct-interim-interval 600
        config accounting-server
            edit 1
```

```
                set status enable
                set server 172.16.200.60
                set secret *********
            next
        end
    next
end
```

To create a user group for the RADIUS server in the GUI:

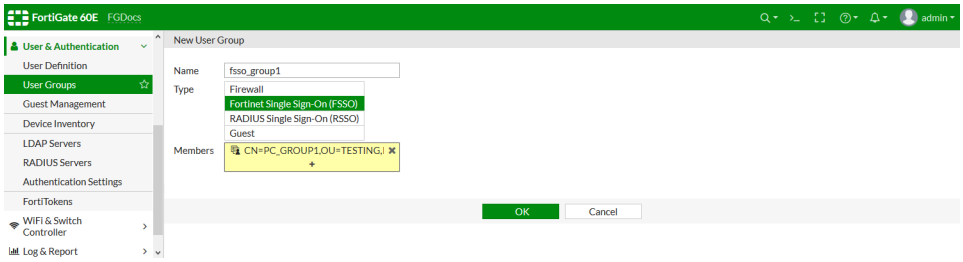
1. Go to *User & Authentication* > *User Groups*.
2. Click *Create New*.
3. Enter a name for the group and set the *Type* to *Firewall*.
4. Add the RADIUS server as a remote group.



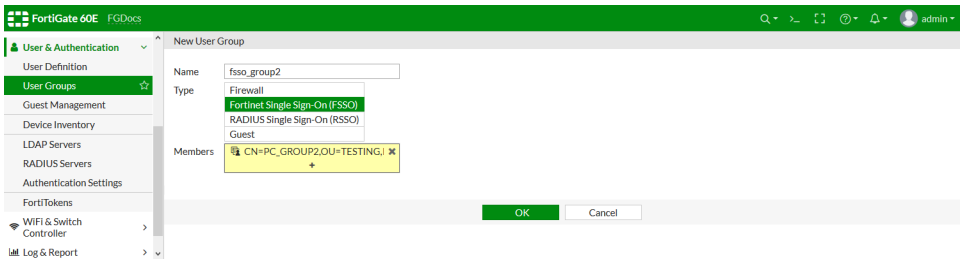
5. Click *OK*.

To create user groups for each of the FSSO groups in the GUI:

1. Go to *User & Authentication* > *User Groups*.
2. Click *Create New*.
3. Enter a name for the group and set the *Type* to *Fortinet Single Sign-On (FSSO)*.
4. Add PC_GROUP1 as a member:
   `CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM`



5. Click *OK*.
6. Add a second user group with PC_GROUP2 as a member:
   `CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM`



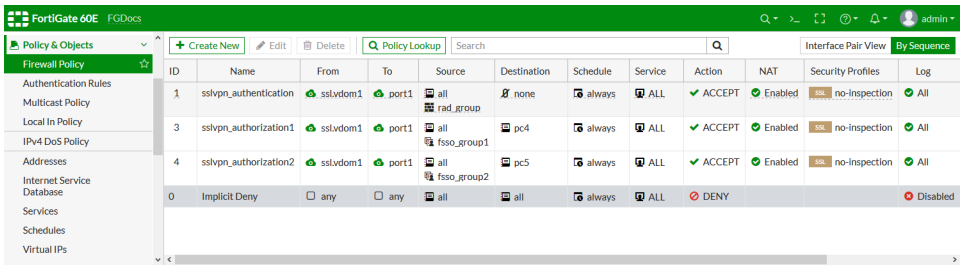
7. Click *OK*.

To create an SSL VPN portal and assign the RADIUS user group to it in the GUI:

1. Go to *VPN* > *SSL-VPN Portals*.
2. Click *Create New*.
3. Configure the portal, then click *OK*.
4. Go to *VPN* > *SSL-VPN Settings*.
5. Configure the required settings.
6. Create an *Authentication/Portal Mapping* table entry:
   a. Click *Create New*.
   b. Set *User/Groups* to *rad_group*.
   c. Set *Portal* to *testportal*.
   d. Click *OK*.
7. Click *OK*.

To create policies for authentication and authorization in the GUI:

1. Go to *Policy & Object* > *Firewall Policy*.
2. Configure a dummy policy for authentication. Set the destination to *none* so that traffic is not allowed through the FortiGate, and add *rad_group* as a source.
3. Configure two authorization policies, with the FSSO groups as sources.



Confirmation

On *Client 1*, log in to FortiClient using *user142*. Traffic can go to *pc4* (172.16.200.44), but cannot go to *pc5* (172.16.200.55).

On *Client 2*, log in to FortiClient using *user143*. Traffic can go to *pc5* (172.16.200.55), but cannot go to *pc4* (172.16.200.44).

On the FortiGate, check the authenticated users list and the SSL VPN status:

```
# diagnose firewall auth list

10.212.134.200, USER142
        type: fsso, id: 0, duration: 173, idled: 173
        server: AD_CollectAgent
        packets: in 0 out 0, bytes: in 0 out 0
        user_id: 16777229
        group_id: 3 33554434
        group_name: fsso_group1 CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM

10.212.134.200, user142
        type: fw, id: 0, duration: 174, idled: 174
        expire: 259026, allow-idle: 259200
        flag(80): sslvpn
        server: rad150
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 4
        group_name: rad_group

10.212.134.201, USER143
        type: fsso, id: 0, duration: 78, idled: 78
        server: AD_CollectAgent
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 1 33554435
        group_name: fsso_group2 CN=PC_GROUP2,OU=TESTING,DC=FSSO-QA,DC=COM

10.212.134.201, user143
        type: fw, id: 0, duration: 79, idled: 79
        expire: 259121, allow-idle: 259200
        flag(80): sslvpn
        server: rad150
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 4
        group_name: rad_group

----- 4 listed, 0 filtered ------
```

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index   User    Auth Type      Timeout         From     HTTP in/out    HTTPS in/out
 0       user142         2(1)            600     10.1.100.145   0/0     0/0
 1       user143         2(1)            592     10.1.100.254   0/0     0/0

SSL VPN sessions:
 Index   User    Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       user142         10.1.100.145    104     32190/16480     10.212.134.200
 1       user143         10.1.100.254    11      4007/4966       10.212.134.201
```

SNAT support for policies with virtual wire pairs

Source NAT (SNAT) support has been added for IPv4 and IPv6 policies with virtual wire pair (VWP) interfaces.

To configure a policy using SNAT and a virtual wire pair:

1. Create the virtual wire pair interface:

```
config system virtual-wire-pair
    edit "test-vw-1"
        set member "port1" "port4"
    next
end
```

2. Create the IP pool. The IP pool must have a different subnet than the VWP peers:

```
config firewall ippool
    edit "vwp-pool-1"
        set startip 172.16.222.99
        set endip 172.16.222.100
    next
end
```

3. Configure the policy:

```
config firewall policy
    edit 88
        set srcintf "port4"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set ippool enable
        set poolname "vwp-pool-1"
        set nat enable
    next
end
```

4. Verify the IP pool functions as expected and traffic passes through:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
23.438095 port4 in 172.16.200.11 -> 172.16.200.156: icmp: echo request
23.438126 port1 out 172.16.222.100 -> 172.16.200.156: icmp: echo request
23.438492 port1 in 172.16.200.156 -> 172.16.222.100: icmp: echo reply
23.438501 port4 out 172.16.200.156 -> 172.16.200.11: icmp: echo reply
24.439305 port4 in 172.16.200.11 -> 172.16.200.156: icmp: echo request
24.439319 port1 out 172.16.222.100 -> 172.16.200.156: icmp: echo request
24.439684 port1 in 172.16.200.156 -> 172.16.222.100: icmp: echo reply
24.439692 port4 out 172.16.200.156 -> 172.16.200.11: icmp: echo reply

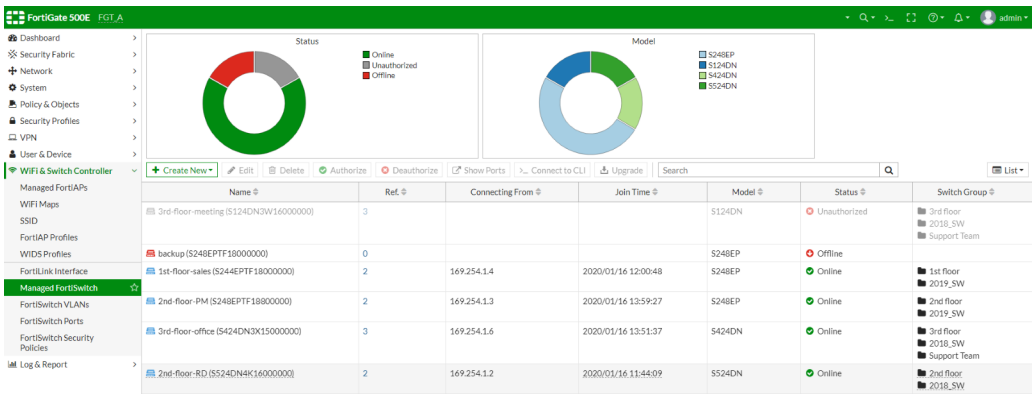
8 packets received by filter
0 packets dropped by kernel
```

GUI support for FortiLink groups

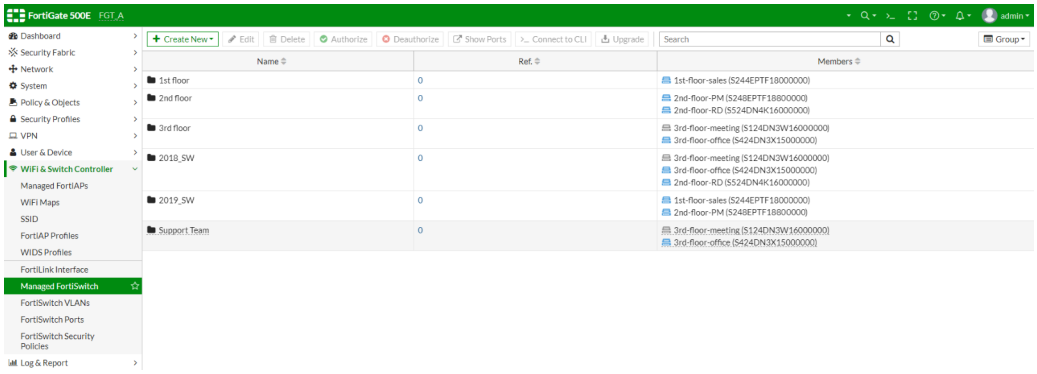
The *Managed FortiSwitch* page includes two new display options: *List* view and *Group* view.

To view the display options:

1. Go to *WiFi & Switch Controller* > *Managed FortiSwitch*.
2. In the toolbar menu, use the dropdown list to switch between views. The previous *Managed FortiSwitch Topology* is under *Topology* view.
   List view:



   Group view:



Increase in maximum number of VIP real servers

Except for desktop models, all other platforms' table size of VIP real servers have increased as follows:

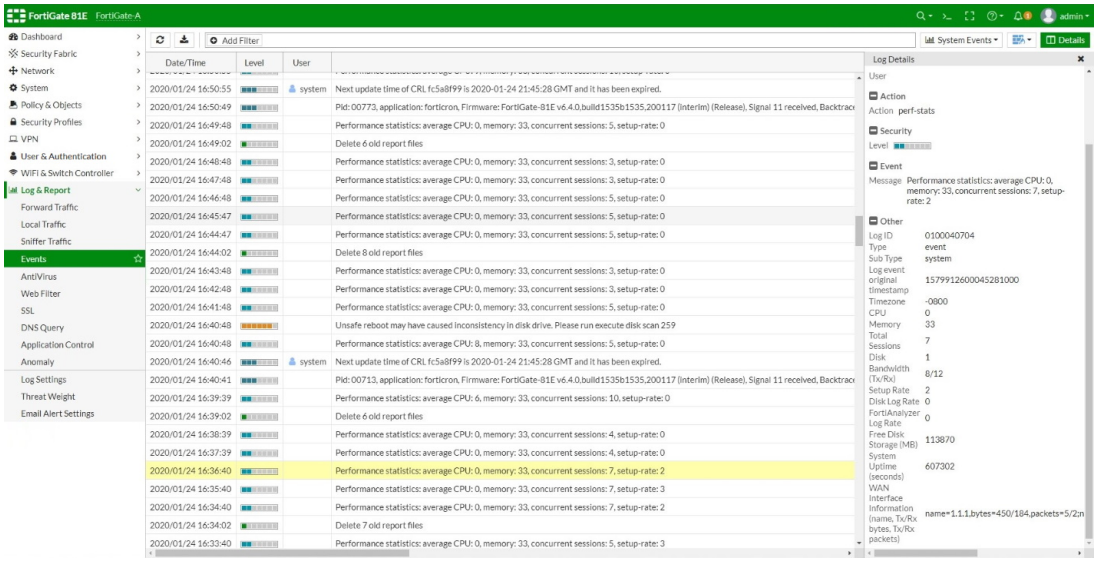
- 1U platforms increased from 8 to 16
- 2U platforms increased from 32 to 64
- High-end platforms increased from 32 to 256

WAN interface bandwidth log

In the system performance statistics event log, `waninfo` (logID 40704) collects WAN interface information for analyzing purpose by FortiAnalyzer. The log supports up to three interfaces assigned a WAN role and the interfaces are displayed in alphabetical order.

To view the WAN interface bandwidth log in the GUI:

1. Go to *Log & Report* > *Events*.
2. On the toolbar menu, select the *System Events* subtype from the dropdown.
3. Select a *Performance statistics* log.
4. Click *Details* and scroll to view the *WAN Interface Information* (log ID 40704).



To view the WAN interface bandwidth log in the CLI:

```
# execute log filter device fortianalyzer
# execute log filter category event
# execute log filter action "perf-stats"
# execute log display
```

Sample logs

When no WAN interface role is configured:

```
1: date=2020-01-24 time=11:17:57 logid="0100040704" type="event" subtype="system" level-
l="notice" vd="vdom1" eventtime=1579893477525796593 tz="-0800" logdesc="System performance
statistics" action="perf-stats" cpu=0 mem=19 totalsession=26 disk=1 bandwidth="46/127"
setuprate=0 disklograte=0 fazlograte=0 freediskstorage=28349 sysuptime=4869 waninfo="N/A" msg-
g="Performance statistics: average CPU: 0, memory: 19, concurrent sessions: 26, setup-rate: 0"
```

After three WAN interface roles are configured:

```
1: date=2020-01-24 time=11:26:58 logid="0100040704" type="event" subtype="system" level-
l="notice" vd="vdom1" eventtime=1579894018320178732 tz="-0800" logdesc="System performance
```

```
statistics" action="perf-stats" cpu=0 mem=19 totalsession=38 disk=1 bandwidth="40/95"
setuprate=0 disklograte=0 fazlograte=0 freediskstorage=28349 sysuptime=5410 wan-
info-
="nam-
e=dmz-
,bytes-
=6519/294381,pac-
ket-
s=50/2407-
7;name=ha1,bytes=474/0,packets=5/0;name=wan1,bytes=92312156/46493734,packets=811589/362592;"
msg="Performance statistics: average CPU: 0, memory:  19, concurrent sessions:  38, setup-
rate: 0"
```

After four WAN interface roles are configured:

```
1: date=2020-01-24 time=15:25:45 logid="0100040704" type="event" subtype="system" level-
l="notice" vd="root" eventtime=1579908345124515733 tz="-0800" logdesc="System performance stat-
istics" action="perf-stats" cpu=1 mem=34 totalsession=14 disk=1 bandwidth="120/93" setuprate=0
disklograte=0 fazlograte=0 freediskstorage=113870 sysuptime=603047 wan-
info-
="nam-
e=~@.TEST........-
,bytes-
=16878/1057173,pac-
ket-
s=282/3200-
;name-
e=port2,bytes=178589214/140899648,packets=657920/769158;name=wan1,bytes=90/184,packets=1/2;"
msg="Performance statistics: average CPU: 1, memory:  34, concurrent sessions:  14, setup-
rate: 0"
```

Source interface setting for NetFlow data

NetFlow data can be routed over the HA management interface when the `ha-direct` option is enabled. The secondary unit does not send out any flow data whether it is running in A-A or A-P.

To route NetFlow data over the HA management interface:

1. On the master unit (FortiGate A), configure the HA and mgmt1 interface settings:

```
(global) # config system ha
    set group-name "test-ha"
    set mode a-p
    set password ENC
    set hbdev "port6" 50
    set hb-interval 4
    set hb-lost-threshold 10
    set session-pickup enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "mgmt1"
        next
```

```
    end
    set override enable
    set priority 200
    set ha-direct enable
end
```

```
(global) # config system interface
    edit "mgmt1"
        set ip 10.6.30.111 255.255.255.0
        set allowaccess ping https ssh http telnet fgfm
        set type physical
        set dedicated-to management
        set role lan
        set snmp-index 1
    next
end
```

2. On the secondary unit (FortiGate B), configure the HA and mgmt1 interface settings:

```
(global) # config system ha
    set group-name "test-ha"
    set mode a-p
    set password ENC
    set hbdev "port6" 50
    set hb-interval 4
    set hb-lost-threshold 10
    set session-pickup enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "mgmt1"
        next
    end
    set override enable
    set priority 100
    set ha-direct enable
end
```

```
(global) # config system interface
    edit "mgmt1"
        set ip 10.6.30.112 255.255.255.0
        set allowaccess ping https ssh http telnet fgfm
        set type physical
        set dedicated-to management
        set role lan
        set snmp-index 1
    next
end
```

3. On the master unit (FortiGate A), configure the NetFlow setting:

```
(global) # config system netflow
    set collector-ip 10.6.30.59
end
```

When the `ha-direct` option is enabled in `config system ha`, FortiOS is no longer allowed to `set source-ip` in `config system netflow`.

4. Verify that NetFlow uses the mgmt1 IP:

```
(global) # diagnose test application sflowd 3
===== Netflow Vdom Configuration =====
Global collector:10.6.30.59:[2055] source ip: 0.0.0.0 active-timeout(minutes):1 inactive-
timeout(seconds):15
____ vdom: vdom1, index=3, is master, collector: disabled (use global config)  HA_direct
   |_ coll_ip:10.6.30.59[2055],src_ip:10.6.30.111,seq_num:6,pkts/time to next template:
14/35
   |_ exported: Bytes:0, Packets:0, Sessions:0 Flows:0
   |____ interface:port1 sample_direction:both device_index:9 snmp_index:3
```

5. Verify that the NetFlow packets are being sent by the mgmt1 IP:

```
(vdom1) # diagnose sniffer packet any 'udp and port 2055' 4
interfaces=[any]
filters=[udp and port 2055]
8.397265 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 60
23.392175 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 188
23.392189 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 60
^C
3 packets received by filter
0 packets dropped by kernel
```

6. On the secondary device (FortiGate B), change the priority so that it becomes the master:

```
(global) # config system ha
    set priority 250
end
```

7. Verify the NetFlow status on FortiGate A, which is using the new master's mgmt1 IP:

```
(global) # diagnose test application sflowd 3
===== Netflow Vdom Configuration =====
Global collector:10.6.30.59:[2055] source ip: 0.0.0.0 active-timeout(minutes):1 inactive-
timeout(seconds):15
____ vdom: vdom1, index=3, is master, collector: disabled (use global config)  HA_direct
   |_ coll_ip:10.6.30.59[2055],src_ip:10.6.30.112,seq_num:8,pkts/time to next template:
16/55
   |_ exported: Bytes:0, Packets:0, Sessions:0 Flows:0
   |____ interface:port1 sample_direction:both device_index:9 snmp_index:3
```

8. Verify that the NetFlow packets use the new source IP on FortiGate B:

```
(vdom1) # diagnose sniffer packet any 'udp and port 2055' 4
interfaces=[any]
filters=[udp and port 2055]
7.579574 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 60
22.581830 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 60
29.038336 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 1140
^C
3 packets received by filter
0 packets dropped by kernel
```

FortiSwitch link status visibility improvements

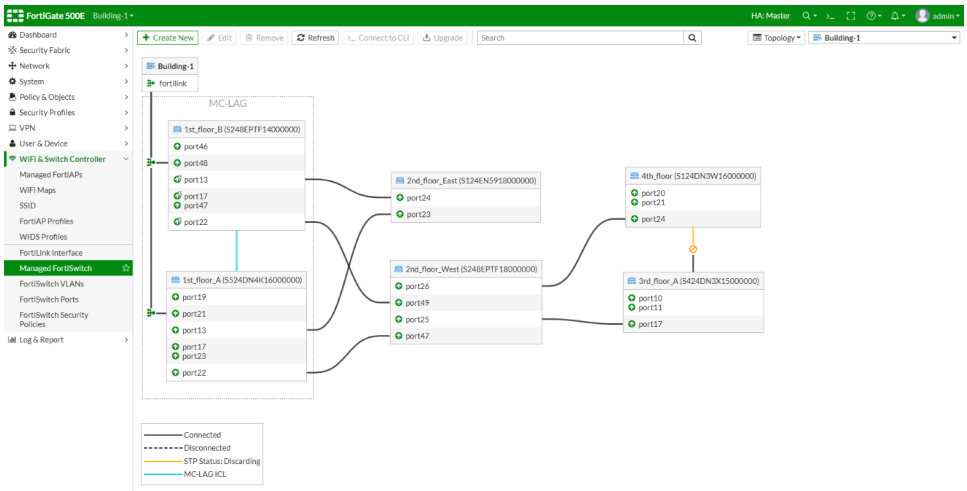
The *Managed FortiSwitch* page topology view has been improved to illustrate the FortiSwitch link status. The following changes have been made in the GUI:

- The FortiSwitch faceplate was replaced with a box that displays ports used for FortiLink management.
- Hovering over switch ports and the links between switches displays a tooltip, which shows the port on both sides of a link.
- The MC-LAG ICL and STP discarding statuses are color coded.
- The MC-LAG cluster is enclosed in a box with an *MC-LAG* label.
- A dropdown list is available to switch between *Managed ForitSwitch* pages of downstream Security Fabric members.
- FortiSwitch names and serial numbers can be used as parameters in the *Search* function.

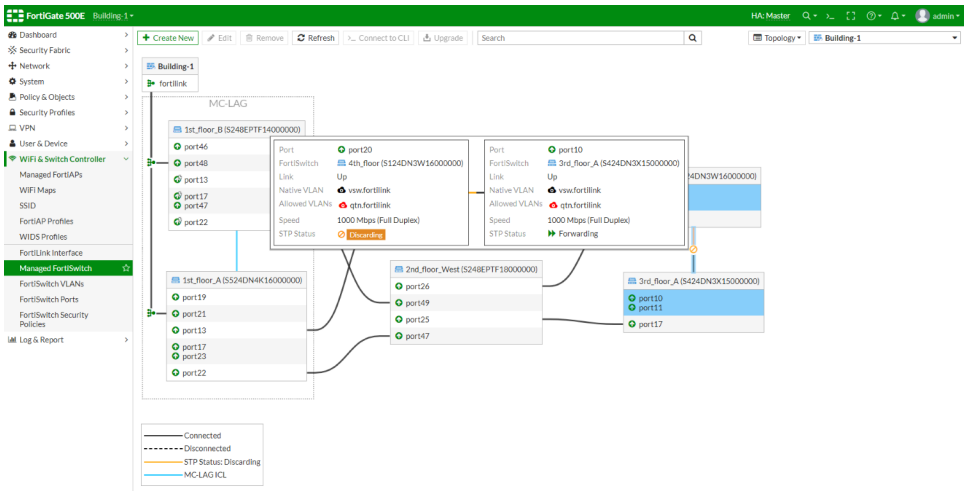
In the following example, FG-500E (Building-1) is the Fabric root device and FG-90E (Building-2) is the downstream device. There are six FortiSwitches managed by FG-500E and two FortiSwitches managed by FG-90E.

To view the FortiSwitch link status in the GUI:

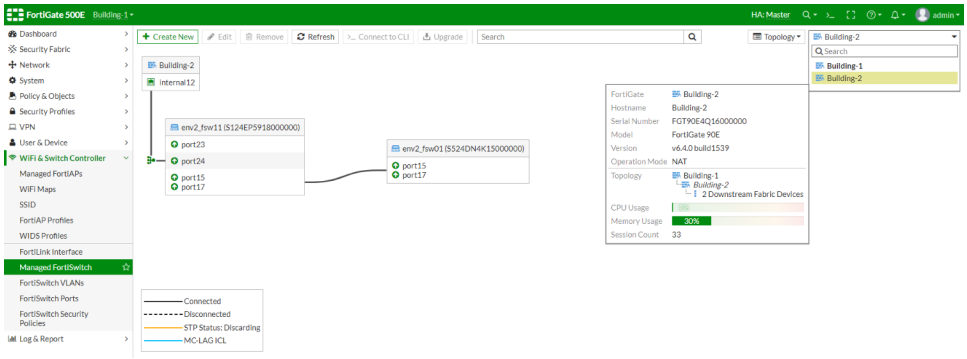
1. On the root device, go to *WiFi & Switch Controller* > *Managed FortiSwitch*.
2. In the toolbar menu, select *Topology* from the dropdown list.
   The MC-LAG cluster is enclosed in a box, and the MC-LAG ICL and STP discarding statuses are color coded:



3. Hover over a link to view the tooltip.



4. In the toolbar menu, use the dropdown list to view the *Managed ForitSwitch* page of the downstream Security Fabric member.

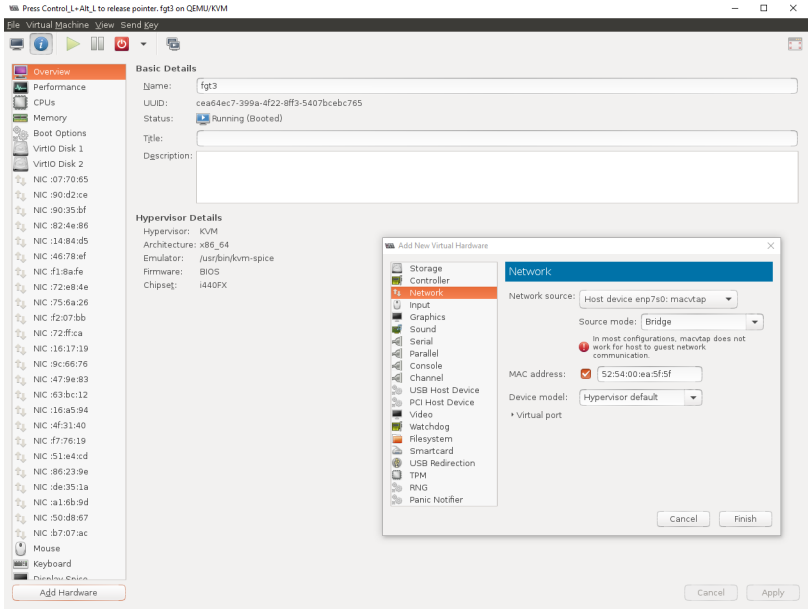


Support up to 24 interfaces on FortiGate VM

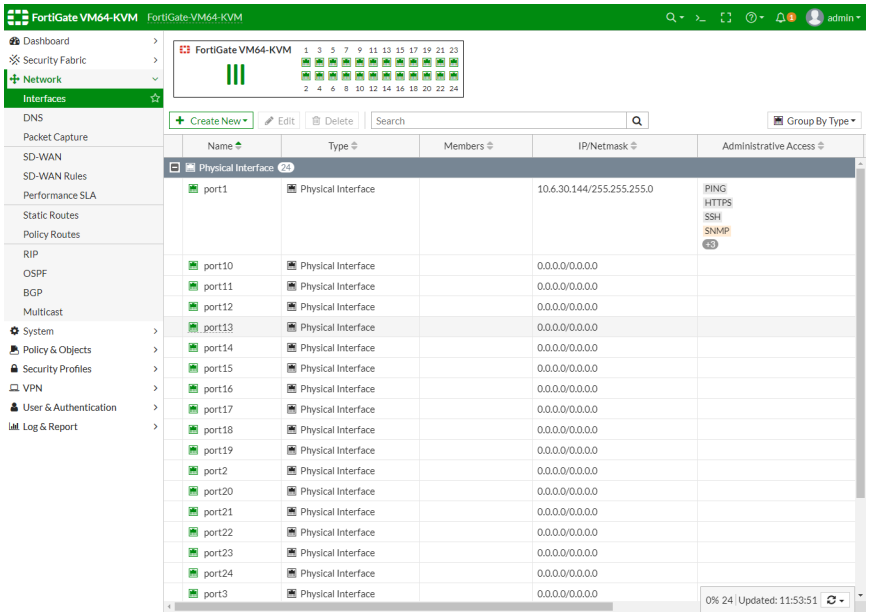
FortiGate VM now supports 24 interfaces or ports.

To use all 24 interfaces:

1. In the hypervisor, such as KVM, create 24 interfaces.



2. On the FortiGate, go to *Network* > *Interfaces* to see all of the available interfaces.



3. In the CLI, enter the following command to see the interfaces:

```
# show system interface
config system interface
    edit "port1"
        set vdom "root"
        set ip 10.6.30.144 255.255.255.0
        set allowaccess ping https ssh snmp http telnet fgfm
        set type physical
        set snmp-index 1
```

```
        next
        edit "port2"
            set vdom "root"
            set type physical
            set snmp-index 3
        next

        ...

        edit "port23"
            set vdom "root"
            set type physical
            set snmp-index 24
        next
        edit "port24"
            set vdom "root"
            set type physical
            set snmp-index 25
        next
        edit "ssl.root"
            set vdom "root"
            set type tunnel
            set alias "SSL VPN interface"
            set snmp-index 2
        next
    end
```

Maintain radio SSID WLAN IDs

WLAN IDs remain the same after a daemon restart or a controller reboot. BSSIDs also remain the same, which keeps the WiFi service stable. This is confirmed by read-only commands in the downloaded backup FortiOS configuration file and the `iwconfig` output from FortiAP.

Sample FortiOS configuration

```
config wireless-controller vap
    edit "wifi-m-1"
        set mesh-backhaul enable
        set ssid "FOS-QA-LFU-FWF61E-M-1"
        set broadcast-ssid disable
        set passphrase qa12345678
        set schedule "always"
    next
    edit "wifi-b-1"
        set ssid "FOS-QA-LFU-FWF61E-B-2"
        set passphrase qa12345678
        set local-bridging enable
        set schedule "always"
    next
    edit "wifi-b-2"
        set ssid "FOS-QA-LFU-FWF61E-B-2"
        set passphrase qa12345678
```

```
        set local-bridging enable
        set schedule "always"
    next
    edit "wifi-b-3"
        set ssid "FOS-QA-LFU-FWF61E-B-3"
        set passphrase qa12345678
        set local-bridging enable
        set schedule "always"
    next
    edit "wifi-b-4"
        set ssid "FOS-QA-LFU-FWF61E-B-4"
        set passphrase qa12345678
        set local-bridging enable
        set schedule "always"
    next
    edit "wifi-b-5"
        set ssid "FOS-QA-LFU-FWF61E-B-5"
        set passphrase qa12345678
        set local-bridging enable
        set schedule "always"
    next
    edit "wifi-b-6"
        set ssid "FOS-QA-LFU-FWF61E-B-6"
        set passphrase qa12345678
        set local-bridging enable
        set schedule "always"
    next
end
```

Backup configuration file output

To verify that the SSIDs remain the same:

```
 config wireless-controller wtp-profile
    edit "FAP423E-default"
        config platform
            set type 423E
        end
        set handoff-sta-thresh 30
        set allowaccess https ssh snmp
        config radio-1
            set band 802.11n,g-only
            set channel-utilization disable
            set vap-all none
        end
        config radio-2
            set band 802.11ac
            set channel-utilization disable
            set darrp enable
            set vap-all none
            set vaps "wifi-b-1" "wifi-b-2" "wifi-b-3" "wifi-b-4" "wifi-b-5" "wifi-b-6" "wifi-
m-1"
            set vap1 "wifi-b-1"
            set vap2 "wifi-b-2"
            set vap3 "wifi-b-3"
            set vap4 "wifi-b-4"
```

```
            set vap5 "wifi-b-5"
            set vap6 "wifi-b-6"
            set vap7 "wifi-m-1"
        end
        set ext-info-enable disable
    next
end
cconfig wireless-controller wtp
    edit "FP423E3X16000320"
        set admin enable
        set wtp-profile "FAP423E-default"
        config radio-1
            set override-vaps enable
            set vap-all none
            set vaps "wifi-b-1" "wifi-b-2" "wifi-b-3" "wifi-b-4" "wifi-b-5" "wifi-b-6" "wifi-
m-1"
            set vap1 "wifi-b-1"
            set vap2 "wifi-b-2"
            set vap3 "wifi-b-3"
            set vap4 "wifi-b-4"
            set vap5 "wifi-b-5"
            set vap6 "wifi-b-6"
            set vap7 "wifi-m-1"
        end
        config radio-2
        end
    next
end
```

FortiAP iwconfig output

To confirm that the BSSIDs remain the same after a reboot:

1. Verify the `iwconfig` output before the reboot:

```
wlan10    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-1"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:38
            < non-relative output omitted >
          Wlan ID:  0    MAC Mode:local    Tun Mode:802.3
wlan11    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-2"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:39
            < non-relative output omitted >
          Wlan ID:  1    MAC Mode:local    Tun Mode:802.3
wlan12    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-3"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3A
            < non-relative output omitted >
          Wlan ID:  2    MAC Mode:local    Tun Mode:802.3
wlan13    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-4"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3B
            < non-relative output omitted >
          Wlan ID:  3    MAC Mode:local    Tun Mode:802.3
wlan14    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-5"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3C
            < non-relative output omitted >
          Wlan ID:  4    MAC Mode:local    Tun Mode:802.3
```

```
wlan15    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-6"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3D
            < non-relative output omitted >
          Wlan ID:  5    MAC Mode:local    Tun Mode:802.3
wlan16    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-M-1"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3E
            < non-relative output omitted >
          Wlan ID:  6    MAC Mode:local    Tun Mode:802.3
wlbgs1    IEEE 802.11ac  ESSID:""
          Mode:Managed  Frequency:5.72 GHz  Access Point: Not-Associated
            < non-relative output omitted >
          Wlan ID:  7    MAC Mode:fat    Tun Mode:local
```

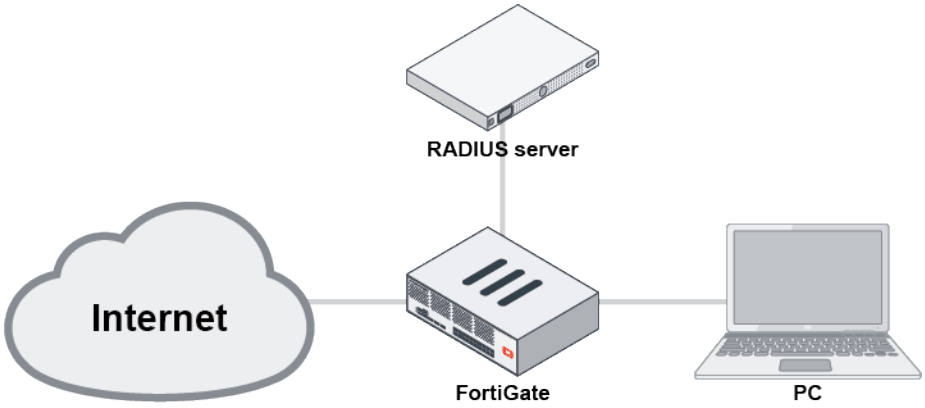
2. Verify the `iwconfig` output after the reboot:

```
wlan10    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-1"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:38
            < non-relative output omitted >
          Wlan ID:  0    MAC Mode:local    Tun Mode:802.3
wlan11    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-2"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:39
            < non-relative output omitted >
          Wlan ID:  1    MAC Mode:local    Tun Mode:802.3
wlan12    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-3"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3A
            < non-relative output omitted >
          Wlan ID:  2    MAC Mode:local    Tun Mode:802.3
wlan13    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-4"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3B
            < non-relative output omitted >
          Wlan ID:  3    MAC Mode:local    Tun Mode:802.3
wlan14    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-5"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3C
            < non-relative output omitted >
          Wlan ID:  4    MAC Mode:local    Tun Mode:802.3
wlan15    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-B-6"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3D
            < non-relative output omitted >
          Wlan ID:  5    MAC Mode:local    Tun Mode:802.3
wlan16    IEEE 802.11ac  ESSID:"FOS-QA-LFU-FWF61E-M-1"
          Mode:Master  Frequency:5.72 GHz  Access Point: 90:6C:AC:DC:62:3E
            < non-relative output omitted >
          Wlan ID:  6    MAC Mode:local    Tun Mode:802.3
wlbgs1    IEEE 802.11ac  ESSID:""
          Mode:Managed  Frequency:5.72 GHz  Access Point: Not-Associated
            < non-relative output omitted >
          Wlan ID:  7    MAC Mode:fat    Tun Mode:local
```

Support for Okta RADIUS attributes filter-Id and class

This feature adds support for RADIUS user group membership information that is returned in the filter-Id (11) and class (25) attributes in RADIUS Access-Accept messages. The group membership information can be used for group

matching in FortiGate user groups in firewall policies and for FortiGate wildcard administrators with remote RADIUS authentication.



In this example, a FortiAuthenticator is used as the RADIUS server. A local RADIUS user on the FortiAuthenticator is configure with two groups in the filter-Id attribute: *okta-group1* and *okta-group2*.

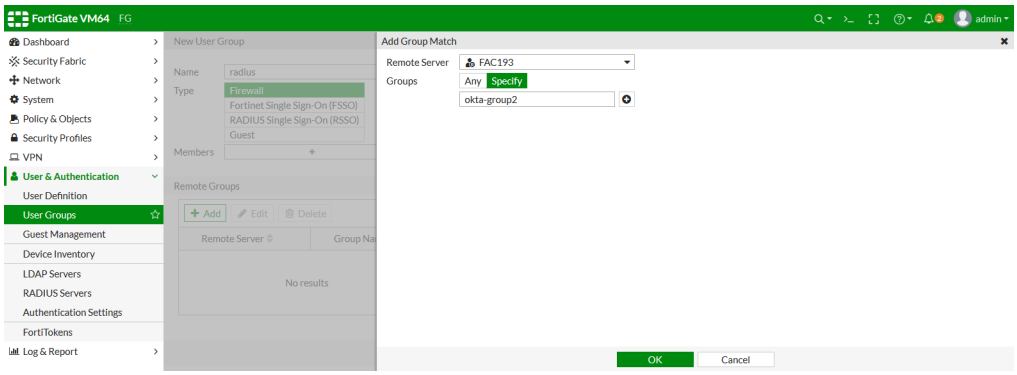
To create the RADIUS user and set the attribute type to override group information:

```
config user radius
    edit "FAC193"
        set server "10.1.100.189"
        set secret **********
        set group-override-attr-type filter-Id
    next
end
```

FortiOS will only use the configured filter-Id attribute, even if the RADIUS server sends group names in both class and filter-id attributes. To return group membership information from the class attribute instead, set `group-override-attr-type` to `class`.

To configure group match in the user group:

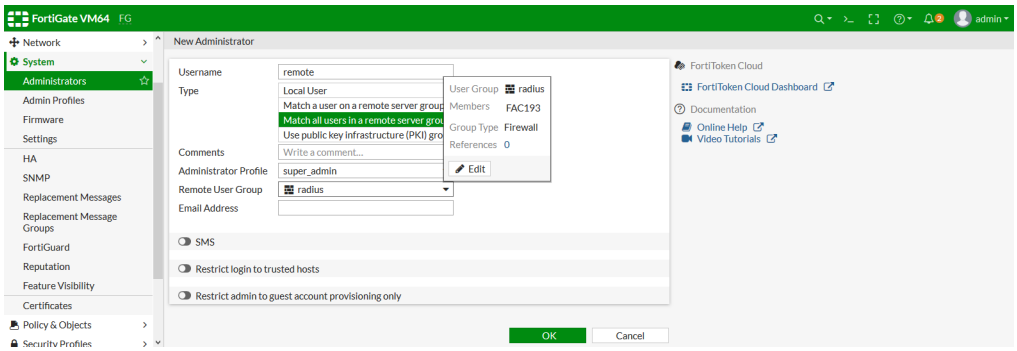
1. Go to *User & Authentication* > *User Groups*.
2. Click *Create New*.
3. Enter a name for the group, and set *Type* to *Firewall*.
4. In the *Remote Groups* table, click *Add*.
5. Set *Remote Server* to the just created RADIUS server, *FAC193*.
6. Set *Groups* to *Specify*, and enter the group name, `okta-group2`. The string must match the group name configured on the RADIUS server for the filter-Id attribute.



7. Click *OK*.
   The remote server is added to the *Remote Groups* table.
8. Click *OK*.
9. Add the new user group to a firewall policy and generate traffic on the client PC that requires firewall authentication, such as connecting to an external web server.
10. After authentication, on the FortiGate, verify that traffic is authorized in the traffic log:
    a. Go to *Log & Report* > *Forward Traffic*.
    b. Verify that the traffic was authorized.

To use the remote user group with group match in a system wildcard administrator configuration:

1. Go to *System* > *Administrators*.
2. Edit an existing administrator, or create a new one.
3. Set *Type* to *Match all users in a remote server group*.
4. Set *Remote User Group* to the remote server.

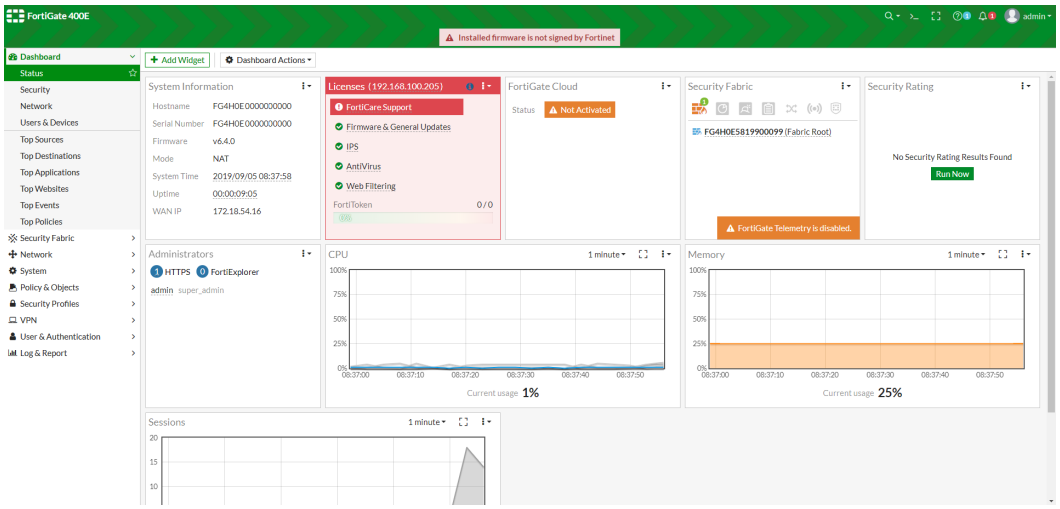


5. Configure the remaining settings as required.
6. Click *OK*.
7. Log in to the FortiGate using the remote user credentials on the RADIUS server.
   If the correct group name is returned in the filter-Id attribute, administrative access is allowed.

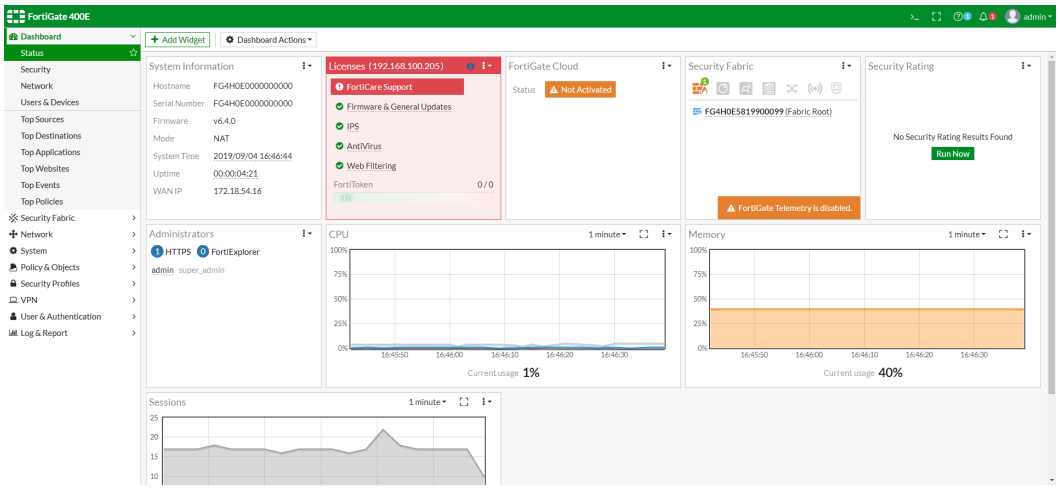
FortiOS image signing and verification

Official FortiOS firmware images are signed by the Fortinet CA. The BIOS checks the validity of an image when it is uploaded to the device. If the image is not signed by the Fortinet CA, a warning message is shown in the GUI.

Unsigned image:



Signed image:



This feature is implemented on the following devices:

FortiGate 40F	FortiWiFi 41F	FortiGate 100F	FortiGate 3400E
FortiGate 40F-3G4G	FortiWiFi 41F-3G4G	FortiGate 101F	FortiGate 3401E
FortiGate 41F	FortiGate 60F	FortiGate 400E	FortiGate 3600E
FortiGate 41F-3G4G	FortiGate 61F	FortiGate 401E	FortiGate 3601E
FortiWiFi 40F	FortiWiFi 60F	FortiGate 600E	
FortiWiFi 40F-3G4G	FortiWiFi 61F	FortiGate 601E	

ICAP response filtering

ICAP HTTP responses can be forwarded or bypassed based on the HTTP header value and status code.

When configuring the ICAP profile, if `response` is enabled, the `respmod-default-action` option can be configured:

- If `respmod-default-action` is set to forward, FortiGate will treat every HTTP response, and send ICAP requests to the ICAP server.
- If `respmod-default-action` is set to bypass, FortiGate will only send ICAP requests if the HTTP response matches the defined rules, and the rule's action is set to forward.

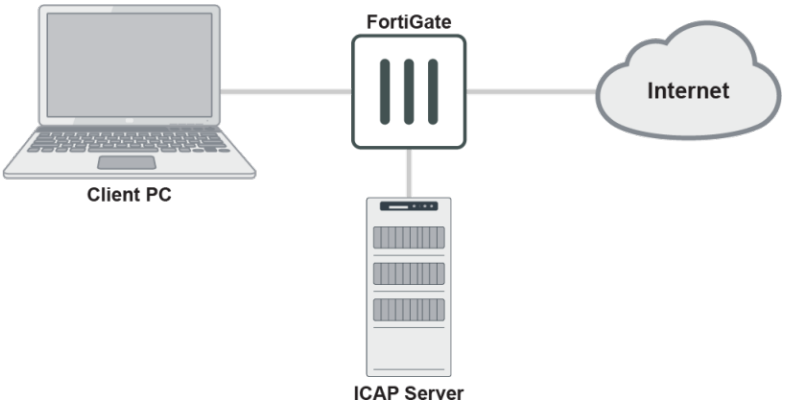
When configuring a response rule:

- The `http-resp-status-code` option is configured to specific HTTP response codes. If the HTTP response has any one of the configured values, then the rule takes effect.
- Multiple header value matching groups can be configured. If the header value matches one of the groups, then the rule takes effect.
- If both status codes and header values are specified in a rule, the response must match at least one of each.

The UTM ICAP log category is used for logging actions when FortiGate encounters errors with the ICAP server, such as no service, unreachable, error response code, or timeout. If an error occurs, a traffic log and an associated UTM ICAP log will be created.

Example

The FortiGate acts as a gateway for the client PC, and connects to a reachable ICAP server. The ICAP server can be in NAT, transparent, or proxy mode.



In this example, client request HTTP responses will be forwarded to the ICAP server from all hosts if they have an HTTP status code of 200, 301, or 302, and have `content-type: image/jpeg` in the their header.

To configure an ICAP profile with HTTP response rules:

```
config icap profile
    edit "icap_profile2"
        set request disable
        set response enable
        set streaming-content-bypass disable
```

```
        set preview disable
        set response-server "icap_server1"
        set response-failure error
        set response-path ''
        set methods delete get head options post put trace other
        set response-req-hdr disable
        set respmod-default-action bypass
        config respmod-forward-rules
            edit "rule2"
                set host "all"
                set action forward
                set http-resp-status-code 200 301 302
                config header-group
                    edit 2
                        set header-name "content-type"
                        set header "image/jpeg"
                    next
                end
            next
        end
    next
end
```

To view the logs if an error occurs:

1. View the traffic log:

```
# execute log filter category 0
# execute log display
1 logs found.
1 logs returned.

 1: date=2019-10-25 time=17:43:47 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1572050627037314464 tz="-0700" srcip=10.1.100.145
srcport=47968 srcintf="port1" srcintfrole="undefined" dstip=172.16.200.46 dstport=80
dstintf="port2" dstintfrole="undefined" poluuid="a4d5324e-f6c3-51e9-ce2d-f360994fb547"
sessionid=43549 proto=6 action="close" policyid=1 policytype="policy" service="HTTP"
dstcountry="Reserved" srccountry="Reserved" trandisp="snat" transip=172.16.200.1
transport=47968 duration=1 sentbyte=485 rcvdbyte=398 sentpkt=6 rcvdpkt=5 appcat="unscanned"
wanin=478 wanout=165 lanin=165 lanout=165 utmaction="block" counticap=1 crscore=5
craction=262144 crlevel="low" utmref=65532-0
```

2. View the UTM ICAP log:

```
# execute log filter category 20
# execute log display
1 logs found.
1 logs returned.

 1: date=2019-10-25 time=17:43:46 logid="2000060000" type="utm" subtype="icap"
eventtype="icap" level="warning" vd="vdom1" eventtime=1572050626010097145 tz="-0700"
msg="Request blocked due to ICAP server error" service="HTTP" srcip=10.1.100.145
dstip=172.16.200.46 srcport=47968 dstport=80 srcintf="port1" srcintfrole="undefined"
dstintf="port2" dstintfrole="undefined" policyid=1 sessionid=43549 proto=6 action="blocked"
profile="icap_profile1" url="/icap_test/"
```

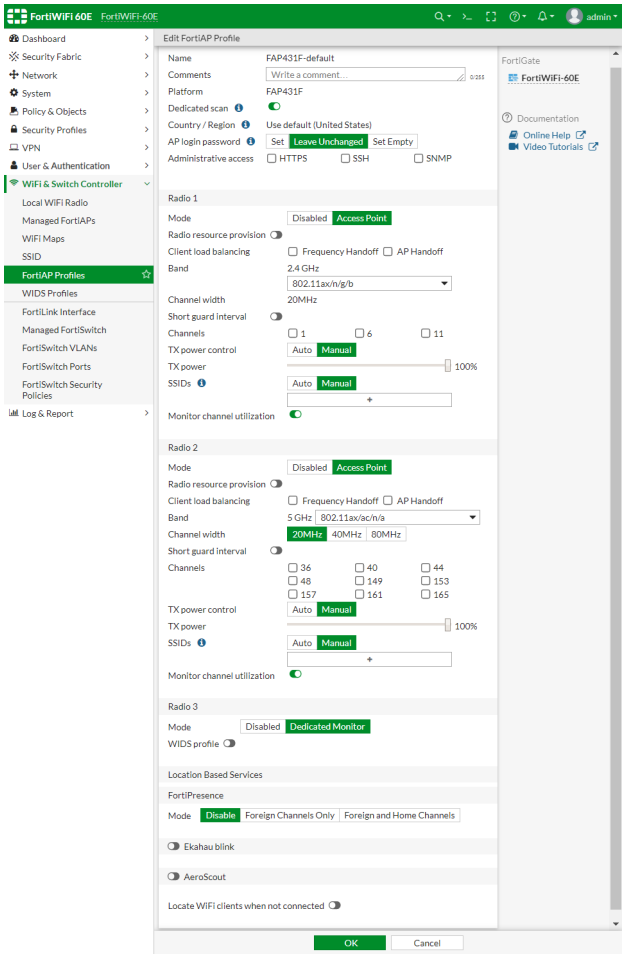
The logs show that, in this case, the ICAP services stopped before the access. When the client tried to access HTTP and ICAP took effect, the FortiGate sent the ICAP request to the ICAP server and received an error. The client sees a *502 Bad Gateway* message, and FortiGate writes the two logs. In the GUI, the logged traffic is displayed as *Result: Deny: UTM Blocked*.

Support for FAP431F and FAP433F

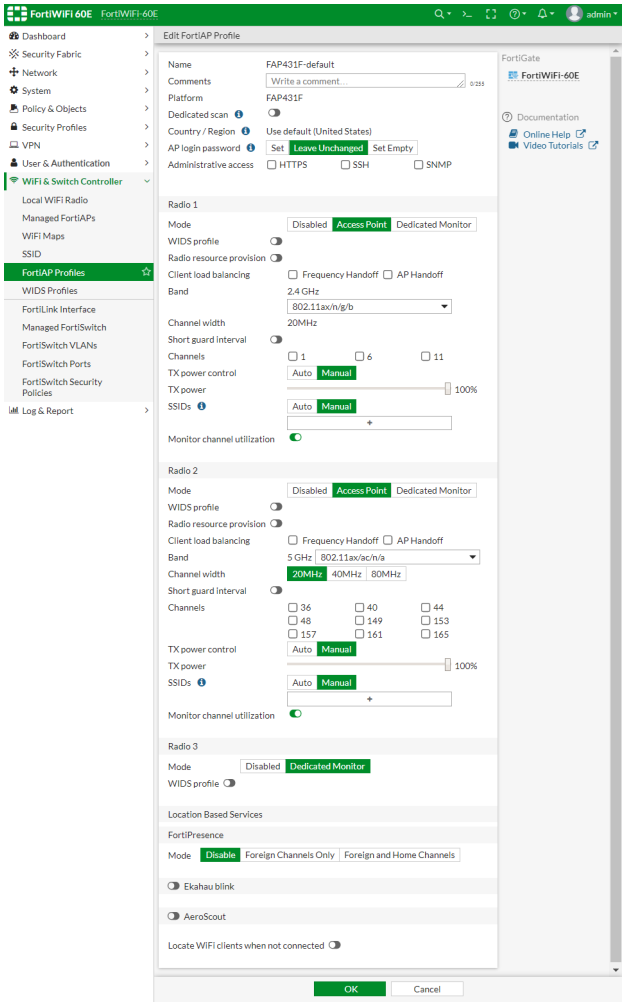
FortiOS 6.4 supports FortiAP NPI models FAP431F and FAP433F. You can use the GUI or CLI to create and edit WTP profiles for platform types 431F and 433F.

To view the default configuration with the GUI:

1. Go to *Wifi & Switch Controller* > *FortiAP Profiles*, and click *Create New*. The *New FortiAP Profile* window opens.
2. From the *Platform* dropdown, select *FAP431F* or *FAP433F*. The default profile opens.
   - *Dedicated scan* is enabled by default.
   - Radio 1 and Radio 2 display *Disabled* and *Access Point* mode.
   - Radio 3 displays *Disabled* and *Dedicated Monitor* mode.



3. Disable *Dedicated Scan*.
   When *Dedicated Scan* is disabled:
   - Radio 1 and Radio 2 display *Disabled*, *Access Point*, and *Dedicated Monitor* mode.
   - Radio 3 displays *Disabled* and *Dedicated Monitor* mode.



Sample CLI configurations

FAP43xF models support 802.11ax on both 2.4G and 5G radios

```
FortiWiFi-60E # config wireless-controller wtp-profile
FortiWiFi-60E (wtp-profile) # ed FAP431F-default
FortiWiFi-60E (FAP431F-default) # conf radio-1
FortiWiFi-60E (radio-1) # set band ?
802.11b              802.11b.
802.11g              802.11g/b.
802.11n              802.11n/g/b at 2.4GHz.
802.11ax             802.11ax/n/g/b at 2.4GHz.
802.11n,g-only       802.11n/g at 2.4GHz.
802.11g-only         802.11g.
802.11n-only         802.11n at 2.4GHz.
802.11ax,n-only      802.11ax/n at 2.4GHz.
802.11ax,n,g-only    802.11ax/n/g at 2.4GHz.
```

```
802.11ax-only        802.11ax at 2.4GHz.

FortiWiFi-60E (radio-1) # en
FortiWiFi-60E (FAP431F-default) # conf radio-2
FortiWiFi-60E (radio-2) # set band ?
802.11a               802.11a.
802.11n-5G            802.11n/a at 5GHz.
802.11ac              802.11ac/n/a.
802.11ax-5G           802.11ax/ac/n/a at 5GHz.
802.11n-5G-only       802.11n at 5GHz.
802.11ac,n-only       802.11ac/n.
802.11ac-only         802.11ac.
802.11ax,ac-only      802.11ax/ac at 5GHz.
802.11ax,ac,n-only    802.11ax/ac/n at 5GHz.
802.11ax-5G-only      802.11ax at 5GHz.

FortiWiFi-60E (radio-2) # en
FortiWiFi-60E (FAP431F-default) #
```

ddscan is enabled by default when creating a new profile for FAP43xF models

```
FortiWiFi-60E # config wireless-controller wtp-profile
FortiWiFi-60E (wtp-profile) # ed 431F
new entry '431F' added
FortiWiFi-60E (431F) # conf platform
FortiWiFi-60E (platform) # set type 431F
FortiWiFi-60E (platform) # sh
config platform
   set type 431F
   set ddscan enable
end
FortiWiFi-60E (platform) # en
FortiWiFi-60E (431F) # sh
config wireless-controller wtp-profile
   edit "431F"
      config platform
         set type 431F
         set ddscan enable
      end
      set handoff-sta-thresh 55
      config radio-1
         set band 802.11ax
      end
      config radio-2
         set band 802.11ax-5G
      end
      config radio-3
         set mode monitor
      end
   next
end
FortiWiFi-60E (431F) # en
```

Radio 3 of FAP43xF models will only support monitor, sniffer, and disable mode when ddscan is enabled or disabled

When ddscan is enabled:

```
FortiWiFi-60E # config wireless-controller wtp-profile
FortiWiFi-60E (wtp-profile) # ed 431F
FortiWiFi-60E (431F) # conf platform
FortiWiFi-60E (platform) # set ddscan enable
FortiWiFi-60E (platform) # end
FortiWiFi-60E (431F) # conf radio-3
FortiWiFi-60E (radio-3) # set mode
disabled    Radio 3 is disabled.
monitor     Radio 3 operates as a dedicated monitor. As a monitor, the radio scans for other
      WiFi access points and adds them to the Rogue AP monitor list.
sniffer     Radio 3 operates as a sniffer capturing WiFi frames on air.

FortiWiFi-60E (radio-3) # set mode monitor
FortiWiFi-60E (radio-3) # end
FortiWiFi-60E (431F) # end
```

When ddscan is disabled:

```
FortiWiFi-60E # config wireless-controller wtp-profile
FortiWiFi-60E (wtp-profile) # ed 431F
FortiWiFi-60E (431F) # conf platform
FortiWiFi-60E (platform) # set ddscan disable
FortiWiFi-60E (platform) # end
FortiWiFi-60E (431F) # conf radio-3
FortiWiFi-60E (radio-3) # set mode
disabled    Radio 3 is disabled.
monitor     Radio 3 operates as a dedicated monitor. As a monitor, the radio scans for other
      WiFi access points and adds them to the Rogue AP monitor list.
sniffer     Radio 3 operates as a sniffer capturing WiFi frames on air.

FortiWiFi-60E (radio-3) # set mode monitor
FortiWiFi-60E (radio-3) # end
FortiWiFi-60E (431F) # end
```

Enhanced autoscale clusters for FortiGate VM

This improvement supports the visibility of autoscale VM clusters on FortiManager, and its ability to read cluster information from new slave members.

When a FortiGate VM slave is added to a cluster, the new slave member can query the cluster about its autoscale environment. FortiManager can then run this query on the new slave member to update its autoscale record.

To view cluster information from a slave member:

```
# diagnose sys ha checksum autoscale-cluster
```

Cluster information sample

Sample cloud topology:

```
FGT_BYOL; master; 10.0.0.6; FGVM04TM00000066
FGT_BYOL; slave; 10.0.0.7; FGVM00000000056
FGT_PAYG; slave; 10.0.0.4; FGTAZ000000000CD
FGT_PAYG; slave; 10.0.0.5; FGTAZ00000000003D
```

From the slave, you can see cluster checksums and the master device.

```
# diagnose sys ha checksum autoscale-cluster
================== FGTAZ000000000CD ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGVM04TM00000066 ==================
is_autoscale_master()=1
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGVM00000000056 ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGTAZ00000000003D ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
```

To get ha sync information from the slave:

```
# get test hasync 50
```

HA sync information:

```
autoscale_count=69. current_jiffies=41235125
   10.0.0.6, timeo=31430, serial_no=FGVM04TM19001766
   10.0.0.7, timeo=31430, serial_no=FGVM04TM19008156
   10.0.0.5, timeo=31430, serial_no=FGTAZR7UZRKKNR3D
connections = 0
```

SNMP traps and query for monitoring DHCP pool

The SNMP DHCP event contains three traps and one query.

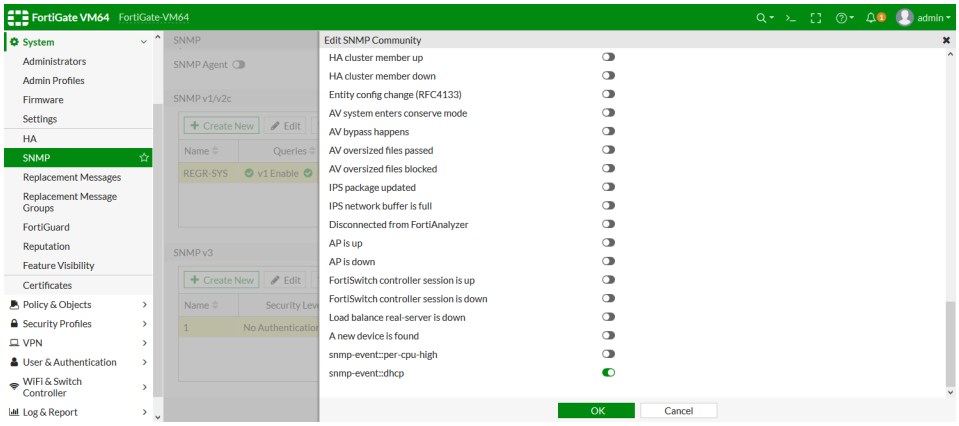
Traps are sent when:

- DHCP server IP pool usage reaches 90%
- DHCP server detect an IP address that is already in use
- DHCP client receives DHCP NAK

SNMP queries are accepted for DHCP lease usage information (OID = 1.3.6.1.4.1.12356.101.23). The query result is based on the leased out percentage.

To enable the SNMP DHCP event in the GUI:

1. Go to *System* > *SNMP*.
2. Click *Create New* in either the *SNMP v1/v2c* table or *SNMP v3* table, or edit an existing community or user.
3. Configure the settings as required.
4. In the *SNMP Events* list, enable *snmp-event::dhcp*.



5. Click *OK*.

To enable the SNMP DHCP event in the CLI:

```
config system snmp community
    edit 1
        set name "REGR-SYS"
        config hosts
            edit 1
                set ip 10.1.100.11 255.255.255.255
            next
            edit 2
```

```
                set ip 172.16.200.55 255.255.255.255
            next
        end
        set events dhcp
    next
end

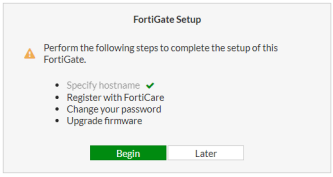
config system snmp user
    edit "1"
        set notify-hosts 172.10.1.0 172.20.1.0
        set events dhcp
        set security-level auth-priv
        set auth-proto sha384
        set auth-pwd ********************
        set priv-proto aes256
        set priv-pwd **********************
    next
end
```

Firmware upgrade notifications

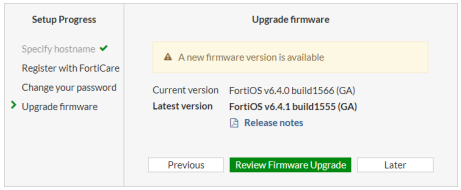
FortiGates with a firmware upgrade license and that are connected to FortiGuard display upgrade notifications in the setup window, the banner, and the FortiGate menu. You can use the CLI console to enable or disable the notifications.

To view the firmware upgrade notifications in the GUI:

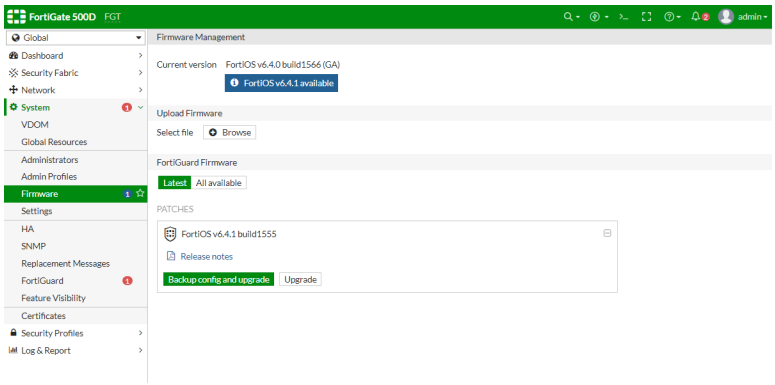
1. When you log in to FortiGate, the *FortiGate Setup* window includes an *Upgrade firmware* step. Click *Begin*.



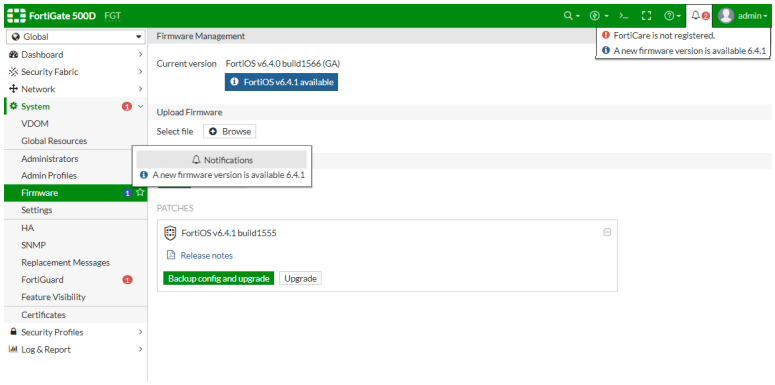
2. Follow the steps in the *Setup Progress*, then click *Review Firmware Upgrade*.



   You are taken to the *Firmware* page.



3. Notifications also appear next to the *Firmware* page in the menu, and below the *Notification* icon in the banner.



To enable or disable the firmware nonfiction in the CLI:

```
config system global
   set gui-firmware-upgrade-setup-warning {enable | disable>}
end
```

Firmware notifications are enabled by default.

Identify the XAUI link used for a specific traffic stream

The `diagnose npu np6 xaui-hash` command takes a 6-tuple input of the traffic stream to identify the NP6 XAUI link that the traffic passes through.

This command is only available on the 38xxD, 39xxD, 34xxE, 36xxE, and 5001E series devices.

Syntax

```
diagnose npu np6 xaui-hash <interface> <proto> <src_ip> <dst_ip> <src_port> <dst_
port>
```

Variable	Description
`<interface>`	The network interface that the packets are coming from.

Variable	Description
<proto>	The proto number, 6 for TCP or 17 for UDP.
<src_ip>	The source IP address.
<dst_ip>	The destination IP address.
<src_port>	The source port.
<dst_port>	The destination port.

Examples

```
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 80
 NP6_ID: 0, XAUI_LINK: 2
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 200
NP6_ID: 6, XAUI_LINK: 2
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 20
NP6_ID: 1, XAUI_LINK: 2
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 23
NP6_ID: 1, XAUI_LINK: 1
```

The `NP6_ID` is the NP index of the model that is being used. It can be found with the `diagnose npu np6 port-list` command.

DHCP client options

When an interface is in DHCP addressing mode, DHCP client options can be configured in the CLI. For example, a vendor class identifier (usually DCHP client option 60) can be specified so that a request can be matched by a specific DHCP offer.

Multiple options can be configured, but any options not recognized by the DHCP server are discarded.

To configure client option 60 - vendor class identifier:

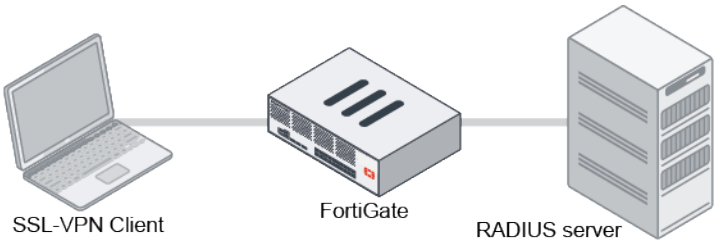
```
config system interface
    edit port1
        set vdom vdom1
        set mode dhcp
        config client-options
            edit 1
                set code 60
                set type hex
                set value aabbccdd
            next
        end
        set type physical
        set snmp-index 4
    next
end
```

Variable	Description
`code <integer>`	DHCP client option code (0 - 255, default = 0). See Dynamic Host Configuration Protocol (DHCP) and Bootstrap Protocol (BOOTP) Parameters for a list of possible options.
`type {hex | string | ip | fqdn}`	DHCP client option type (default = hex).
`value <string>`	DHCP client option value.
`ip <ip>`	DHCP client option IP address. This option is only available when `type` is `ip`.

NAS-IP support per SSL-VPN realm

For RADIUS authentication and authorization, the RADIUS client (the FortiGate) passes the username, password, and NAS-IP to the RADIUS server in its access request. The RADIUS server authenticates and authorizes based on this information. Each RADIUS server can be configured with multiple NAS-IPs for authenticating different groups and NAS clients.

On the FortiGate, configuring the NAS-IP in the realm settings overrides the RADIUS server setting, allowing multiple NAS-IPs to be mapped to the same RADIUS server.



In this example, the user wants to present one FortiGate VDOM with different NAS-IPs to a single RADIUS server based on specific rules.

To configure the SSL-VPN to use the NAS-IP in the realm settings:

1. Configure a RADIUS user and add then to a group:

```
config user radius
    edit "fac150"
        set server "172.16.200.150"
        set secret ********
        set nas-ip 172.16.200.2
        config accounting-server
            edit 1
                set status enable
                set server "172.16.200.150"
                set secret ********
            next
        end
    next
end
```

```
config user group
    edit "radgrp"
        set member "fac150"
    next
end
```

2. Configure a realm for the user with a different NAS-IP:

```
config vpn ssl web realm
    edit "realm1"
        set login-page '.......'
        set radius-server "fac150"
        set nas-ip 10.1.100.2
    next
end
```

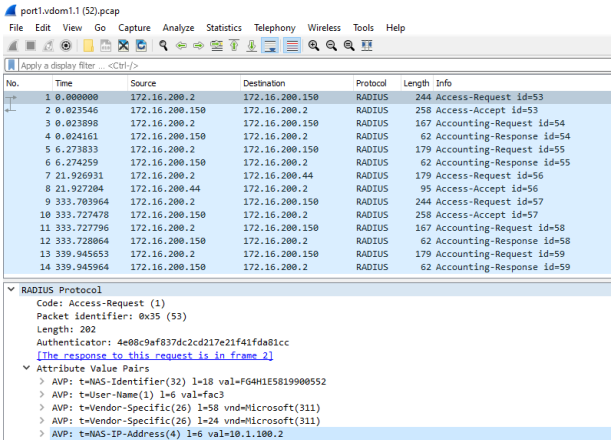
3. Configure SSL-VPN with an authentication rule that includes the user group and the realm:

```
config vpn ssl settings
    ...
    config authentication-rule
        edit 1
            set groupd "radgrp"
            set portal "testportal1"
            set realm "realm1"
        next
    end
end
```

4. Create a firewall policy:

```
config firewall policy
    edit 1
        set name "sslvpn1"
        ...
        set srcintf "ssl.vdom1"
        set groups "radgrp"
    next
end
```

Because the RADIUS server and NAS-IP are specified in realm1, its NAS-IP is used for authentication.

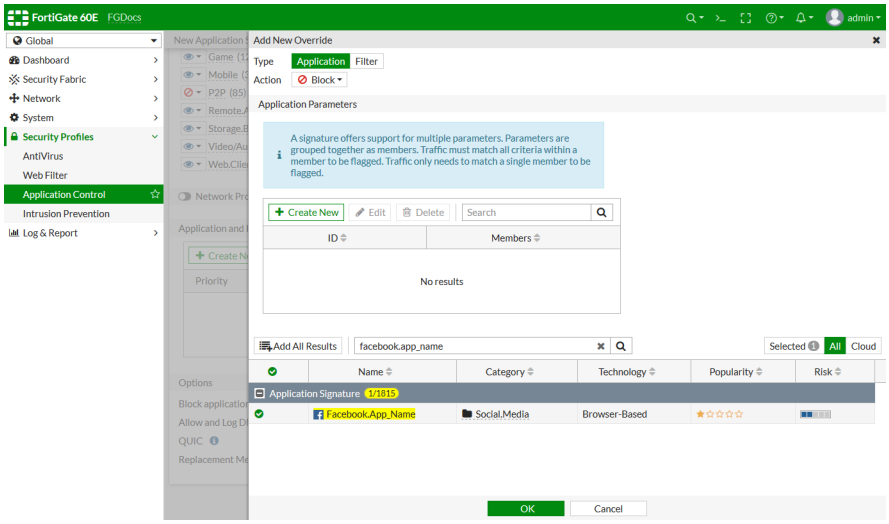


Matching multiple parameters on application control signatures

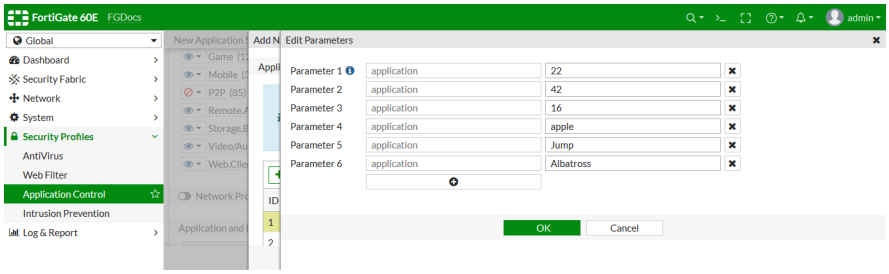
Application control signatures that support parameters (such as SCADA protocols) can have multiple parameters grouped together and matched at the same time. To match a member, traffic must match all of the parameters. To match a signature, at least one member must be matched.

To configure an application sensor with multiple parameters in the GUI:

1. Go to *Security Profiles* > *Application Control*.
2. Click *Create New* to create a new application sensor, or edit an existing sensor.
3. In the *Application and Filter Overrides* table, click *Create New*.
4. Add an application signature that has parameters, such as *Facebook.App_Name*.

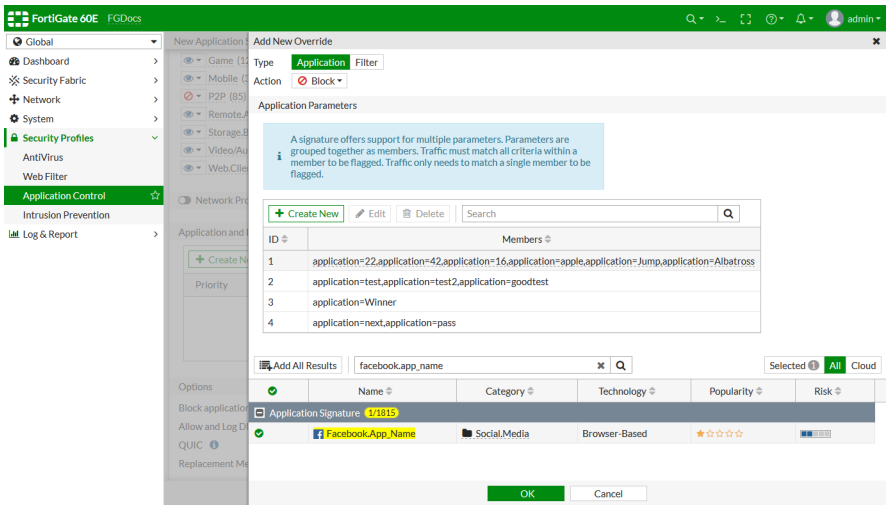


5. Click *Create New* to add parameters. Multiple parameters can be added to a member.



6. Click *OK*.

7. Add more members as needed.



8. Click *OK*.

To configure an application sensor with multiple parameters in the CLI:

```
config application list
    edit "g-test"
        set other-application-log enable
        config entries
            edit 1
                set application 23813
                config parameters
                    edit 1
                        config members
                            edit 1
                                set name "application"
                                set value "22"
                            next
                            ...
                            edit 6
                                set name "application"
                                set value "Albatross"
                            next
                        end
                    next
                    edit 2
                        config members
                            edit 1
                                set name "application"
                                set value "test"
                            next
                            ...
                        end
                    next
                    edit 3
                        config members
                            edit 1
                                set name "application"
                                set value "Winner"
```

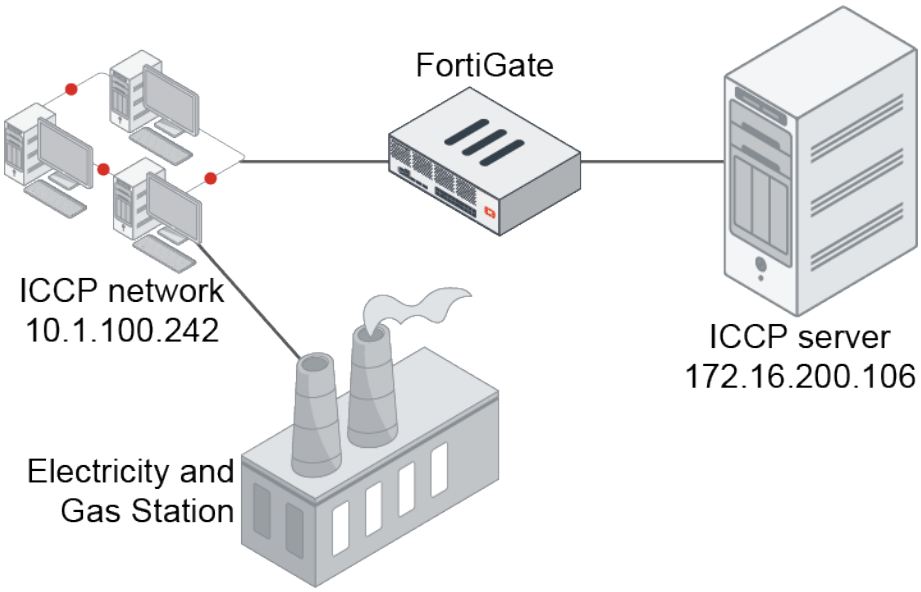
```
                        next
                    end
                next
                edit 4
                    config members
                        edit 1
                            set name "application"
                            set value "next"
                        next
                        edit 2
                            set name "application"
                            set value "pass"
                        next
                    end
                next
            end
        next
        edit 2
            set category 2 6
        next
    end
next
end
```

Detecting IEC 61850 MMS protocol in IPS

IEC 61850 is a SCADA protocol whose services are mapped to a number of protocols, including MMS services. MMS/ICCP detection is supported in IPS. The purpose of the MMS dissectors is to identify every IEC 61850 service to distinguish different MMS/ICCP messages. IPS engine 6.0.12 and later support MMS dissectors.

The following scenarios are also supported:

- Multiple MMS PDUs are transferred in one TCP payload, and the IPS engine identifies individuals.
- An MMS message is split over multiple TCP segments, where MMS runs over COTP segments.
- ICCP/TASE.2 that also uses MMS transport (ISO transport over TCP for ICCP) is detected.



Industrial signatures must be enabled in the global IPS settings to receive MMS/ICCP signatures. By default, industrial signatures are excluded.

```
config ips global
    set exclude-signatures none
end
```

Below are some industrial signatures for MMS/ICCP messages that can be detected by the IPS engine. This is not an exhaustive list.

- MMS_GetNameList.Request
- MMS_GetNamedVariableListAttributes.Request
- MMS_GetVariableAccessAttributes.Request
- MMS_Identify.Request
- MMS_Initiate.Request
- MMS_Read.Request
- MMS_Reset.Request
- ICCP_Transfer.Reporting
- ICCP_Create.Dataset
- ICCP_Abort
- ICCP_Start.Transfer.DSTransferSet
- ICCP_Get.Dataset.Element.Values
- ICCP_Get.Next.DSTransfer.Set.Value
- ICCP_Delete.Dataset
- ICCP_Start.Transfer.IMTransferSet

Diagnose command

The COTP dissector adds support for identifying every MMS PDU, and let the IPS engine separate them, like the Modbus and IEC-104 services for example.

```
# diagnose ips debug enable all
# diagnose debug enable
[284@78]ips_l7_dsct_processor: serial=8142 create: cotp
[284@78]ips_l7_dsct_processor: serial=8142 create: iec104
[284@78]ips_l7_dsct_processor: serial=8142 create: modbus
```

Log samples

MMS dissectors can be triggered, and MMS/ICCP signatures can be monitored and logged.

Log samples:

```
date=2020-03-26 time=15:51:10 logid="1059028704" type="utm" subtype="app-ctrl" event-
type="signature" level="information" vd="vd1" eventtime=1585263070836106492 tz="-0700" appid-
d=43699 srcip=10.1.100.242 dstip=172.16.200.106 srcport=50963 dstport=102 srcintf="port13"
srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6 service="tcp/26112"
direction="outgoing" policyid=1 sessionid=2711 applist="test" action="pass" appcat-
t="Industrial" app="MMS_Read.Request" incidentserialno=376610508 msg="Industrial: MMS_
Read.Request," apprisk="elevated"
```

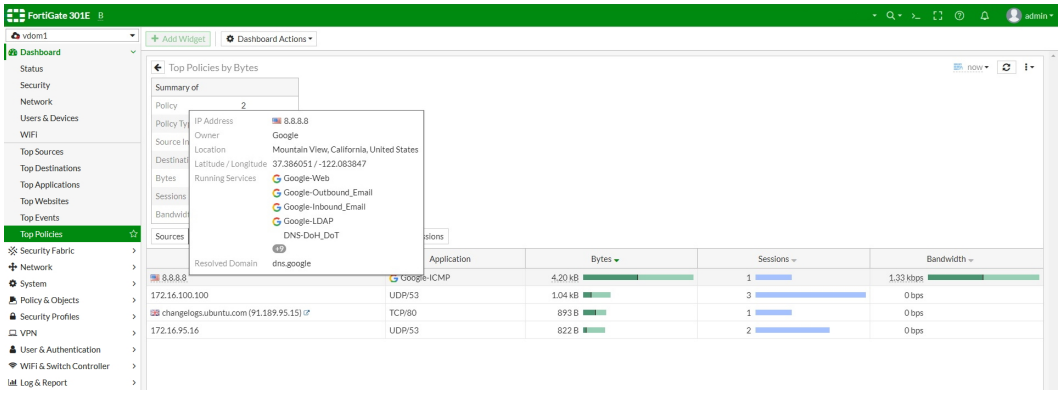
```
date=2020-03-26 time=16:15:45 logid="1059028704" type="utm" subtype="app-ctrl" event-
type="signature" level="information" vd="vd1" eventtime=1585091746264983273 tz="-0700" appid-
d=44684 srcip=10.1.100.242 dstip=172.16.200.106 srcport=41665 dstport=102 srcintf="port13"
srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6 service="tcp/26112"
direction="incoming" policyid=1 sessionid=194463 applist="test" action="pass" appcat-
t="Industrial" app="ICCP_Transfer.Reporting" incidentserialno=762763993 msg="Industrial: ICCP_
Transfer.Reporting," apprisk="elevated"
```

IP address tooltips

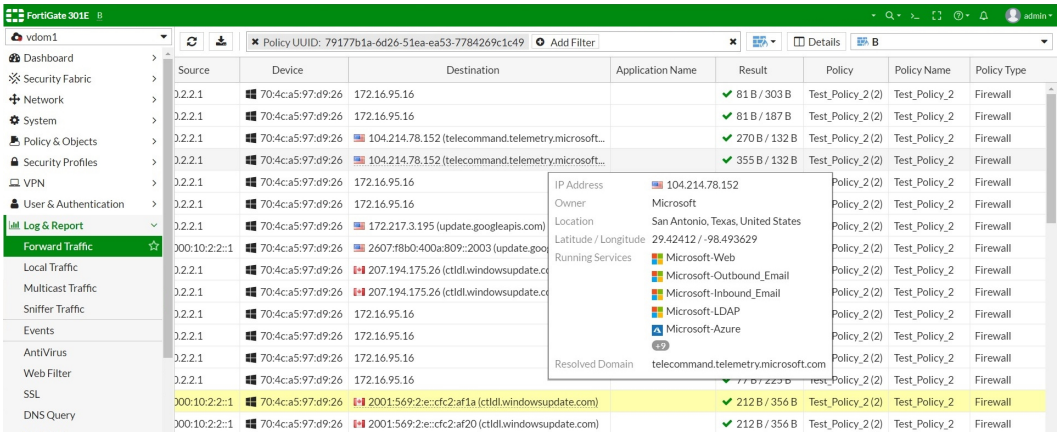
Hovering over an IP address on different GUI pages (for example, *Dashboard* > *Top Policies*, *Log & Report* > *Forward Traffic*, *Security Fabric* > *External Connectors*) displays a tooltip that contains additional information about the IP such as its country, location, owner, resolved domains, and internet services.

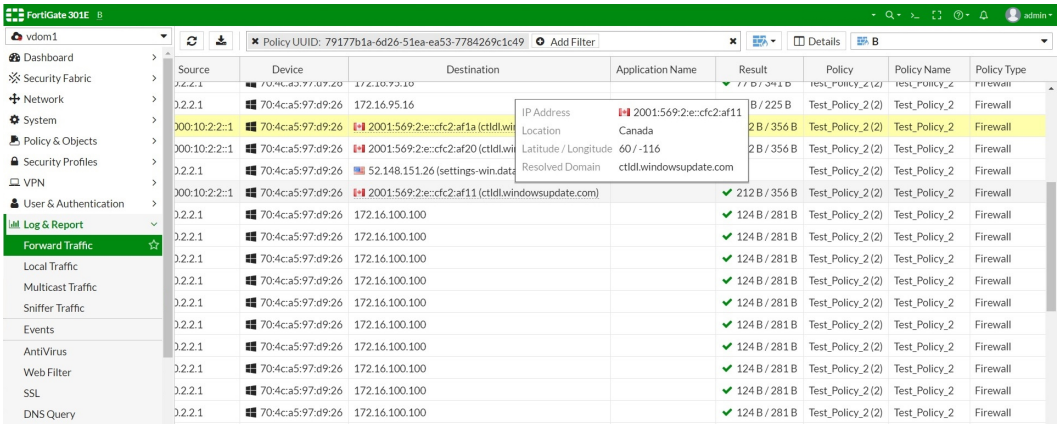
Tooltip examples

Dashboard > *Top Policies* page:

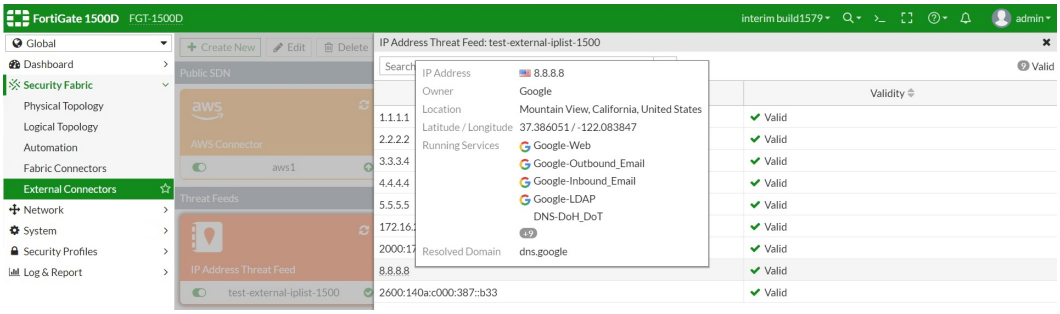


Log & Report > *Forward Traffic* page:





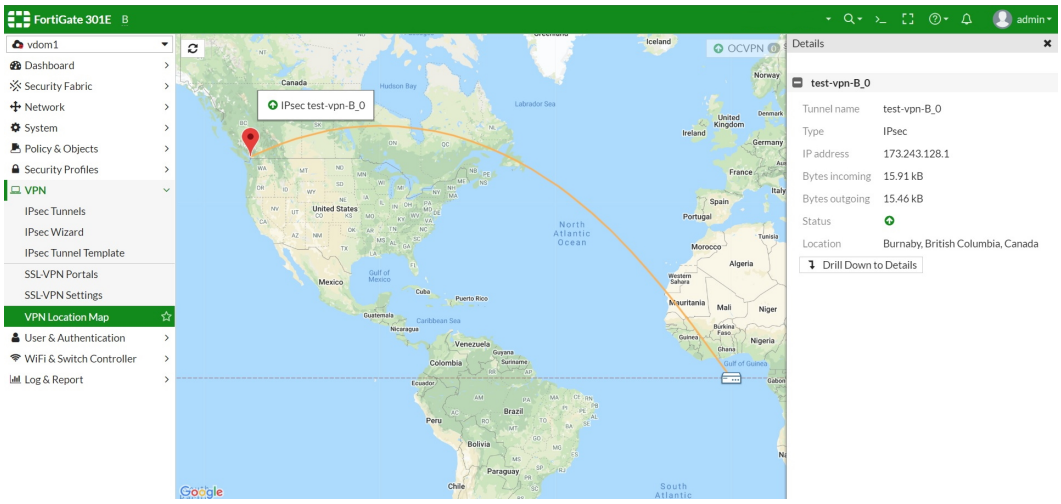
Security Fabric > *External Connectors* page in the *Threat Feeds* section. Hover over a card and click *View Entries*:



Network > *DNS* page:



VPN > *VPN Location Map* page:



Interface-based traffic shaping

Interface-based traffic shaping is supported on some devices.

An administrator configures the WAN interface's maximum outbound bandwidth and, based on that, creates a traffic shaping profile with a percentage based shaper. This allows for proper QoS and traffic shaping. VLAN interfaces are not supported.

This feature is supported on FortiGate 600E, 500E, 300E, 100F, 60F, and 40F models.

To configure interface-based traffic shaping:

1. Enable NPU offloading when doing interface-based traffic shaping according to the egress-shaping-profile:

```
config system npu
    set intf-shaping-offload enable
end
```

2. Configure shaping profiles:

```
config firewall shaping-profile
    edit "sdwan"
        set default-class-id 4
        config shaping-entries
            edit 1
                set class-id 4
                set guaranteed-bandwidth-percentage 3
                set maximum-bandwidth-percentage 5
            next
            edit 2
                set class-id 3
                set priority medium
                set guaranteed-bandwidth-percentage 50
                set maximum-bandwidth-percentage 100
            next
            edit 3
```

```
                set class-id 2
                set priority low
                set guaranteed-bandwidth-percentage 1
                set maximum-bandwidth-percentage 5
            next
        end
    next
end
```

The class number is limited to 16.

3. Configure a traffic shaper and shaping policy:

```
config firewall shaper traffic-shaper
    edit "Transactional"
        set priority medium
    next
end
```

```
config firewall shaping-policy
    edit 1
        set service "ALL"
        set dstintf "any"
        set traffic-shaper "Transactional"
        set class-id 3
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

4. Apply the egress shaping profile on the interface:

```
config system interface
    edit "port2"
        set vdom "root"
        set ip 10.1.100.23 255.255.255.0
        set allowaccess ping
        set type physical
        set outbandwidth 500
        set egress-shaping-profile "sdwan"
        set snmp-index 4
    next
end
```

5. Configure a firewall policy:

```
config firewall policy
    edit 3
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
end
```





Copyright© 2020 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, FortiCare® and FortiGuard®, and certain other marks are registered trademarks of Fortinet, Inc., in the U.S. and other jurisdictions, and other Fortinet names herein may also be registered and/or common law trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance and other metrics contained herein were attained in internal lab tests under ideal conditions, and actual performance and other results may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's General Counsel, with a purchaser that expressly warrants that the identified product will perform according to certain expressly-identified performance metrics and, in such event, only the specific performance metrics expressly identified in such binding written contract shall be binding on Fortinet. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. In no event does Fortinet make any commitment related to future deliverables, features or development, and circumstances may change such that any forward-looking statements herein are not accurate. Fortinet disclaims in full any covenants, representations, and guarantees pursuant hereto, whether express or implied. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.