

FORTINET®



FortiOS™ Handbook - Firewall

VERSION 6.0.2



FORTINET DOCUMENT LIBRARY

<https://docs.fortinet.com>

FORTINET VIDEO GUIDE

<https://video.fortinet.com>

FORTINET KNOWLEDGE BASE

<http://kb.fortinet.com>

FORTINET BLOG

<https://blog.fortinet.com>

CUSTOMER SERVICE & SUPPORT

<https://support.fortinet.com>

FORTINET COOKBOOK

<http://cookbook.fortinet.com>

FORTINET NSE INSTITUTE (TRAINING)

<https://training.fortinet.com/>

FORTIGUARD CENTER

<https://fortiguard.com>

FORTICAST

<http://forticast.fortinet.com>

END USER LICENSE AGREEMENT AND PRIVACY POLICY

<https://www.fortinet.com/doc/legal/EULA.pdf>

<https://www.fortinet.com/corporate/about-us/privacy.html>

FEEDBACK

Email: techdocs@fortinet.com



July 26, 2018

FortiOS™ Handbook - Firewall

01-602-481066-20180728

TABLE OF CONTENTS

Change log	15
Introduction	16
How this guide is organized	16
Fundamentals	16
Firewall optimization	18
How does a FortiGate protect your network?	18
What's new for Firewall in 6.0.1	19
What's new for Firewall in 6.0	19
Firewall concepts	21
What is a firewall?	21
Network layer or packet filter firewalls	22
Application layer firewalls	22
Proxy servers	23
UTM/ NGFW	23
FortiGate modes	24
NAT mode	25
Transparent mode	25
How packets are handled by FortiOS	25
Interfaces and zones	26
Interfaces	27
Zones	27
Virtual wire pair	27
Access control lists	28
Incoming interfaces	28
Addresses	28
Services	28
Firewall policies	29
Firewall policy parameters	30
Application groups for NGFW policies	31
What is not expressly allowed is denied	35
Policy order	36
Policy identification	38
UUID support	38
NTurbo support CAPWAP traffic	39

Learning mode for policies.....	39
Policy modes.....	41
NGFW policy mode and NAT.....	42
Application control in NGFW policy mode.....	42
Web filtering in NGFW mode.....	43
Other NGFW policy mode options.....	44
Security profiles.....	44
AntiVirus.....	45
Web Filter.....	45
Application Control.....	46
Intrusion Prevention (IPS).....	46
Anti-Spam.....	46
Data Leak Prevention (DLP).....	46
VoIP.....	47
ICAP.....	47
Security profile groups.....	47
Proxy option components.....	50
The use of different proxy profiles and profile options.....	50
SSL/SSH inspection.....	52
SSH MITM deep inspection.....	53
Mirroring SSL inspected traffic.....	57
Encryption strength for proxied SSH sessions.....	57
RPC over HTTP.....	58
Configuration in protocol options.....	58
Configuration in SSL/SSH inspection.....	58
NAT.....	58
The origins of NAT.....	59
Dynamic NAT.....	59
Static NAT.....	60
Benefits of NAT.....	61
NAT in transparent mode.....	62
Central NAT table.....	63
NAT64 and NAT46.....	63
NAT66.....	64
How FortiOS differentiates sessions when NATing.....	65
IP pools.....	71
Source IP address and IP pool address matching when using a range.....	72
ARP replies.....	73
IP pools and zones.....	74
Fixed port.....	74
Match-VIP.....	74
Services and TCP ports.....	74

Protocol types.....	75
VPN policies.....	100
IPsec policies.....	100
DSRI.....	100
CLI syntax for changing the status of the DSRI setting.....	101
Interface policies.....	101
DoS protection.....	102
One-Arm IDS.....	106
IPv6 IPS.....	107
Traffic destined to the FortiGate unit.....	107
Dropped, flooded, broadcast, multicast and L2 packets.....	107
GUI and CLI.....	107
Local-In policies.....	108
Security policy 0.....	109
DNS traffic in NGFW policy-mode.....	109
Deny policies.....	110
Accept policies.....	110
Fixed port.....	110
Endpoint security.....	111
Traffic logging.....	111
IPv6.....	113
IPv6 addressing.....	113
Possible addresses:.....	113
IPv6 address syntax.....	113
IPv6 packet structure.....	114
Jumbograms and jumbo payloads.....	114
Fragmentation and reassembly.....	114
Benefits of IPv6.....	114
IPv6 in FortiOS.....	115
IPv6 features.....	116
IPv6 policies.....	116
IPv6 network address translation.....	124
IPv6 tunneling.....	129
Tunneling IPv6 through IPsec VPN.....	131
SIP over IPv6.....	131
New Fortinet FortiGate IPv6 MIB fields.....	132
IPv6 per-IP traffic shaper.....	133
DHCPv6.....	134
IPv6 forwarding.....	135
Obtaining IPv6 addresses from an IPv6 DHCP server.....	135
Authentication support.....	135
FSSO.....	136

IPv6 Neighbor Discovery Proxy.....	136
IPv6 configuration.....	137
IPv6 address groups.....	137
IPv6 address ranges.....	138
IPv6 firewall addresses.....	138
IPv6 SSH.....	140
ICMPv6.....	141
IPv6 IPsec VPN.....	142
TCP MSS values.....	155
BGP and IPv6.....	155
RIPng — RIP and IPv6.....	156
IPv6 RSSO support.....	162
IPv6 IPS.....	162
Blocking IPv6 packets by extension headers.....	162
IPv6 denial of service policies.....	163
Configure hosts in an SNMP v1/2c community to send queries or receive traps.....	163
IPv6 PIM sparse mode multicast routing.....	163
IPv6 Neighbor Discovery Proxy.....	164
Network defense.....	166
Monitoring.....	166
Blocking external probes.....	166
Address sweeps.....	167
Port scans.....	167
Probes using IP traffic options.....	167
Evasion techniques.....	169
Defending against DoS attacks.....	171
The “three-way handshake”.....	171
SYN flood.....	172
SYN spoofing.....	172
DDoS SYN flood.....	173
Configuring the SYN threshold to prevent SYN floods.....	174
SYN proxy.....	174
Other flood types.....	174
DoS policies.....	174
Inside FortiOS: Denial of Service (DoS) protection.....	176
About DoS and DDoS attacks.....	176
FortiOS DoS and DDoS protection.....	176
FortiOS DDoS prevention.....	177
Configuration options.....	178
Standard configuration.....	178
Out of band configuration (sniffer mode).....	178
DoS policies.....	179

Hardware acceleration.....	179
The FortiGuard Center.....	179
Policy configuration.....	181
Viewing firewall policies.....	181
Menu items.....	182
Menu items not shared by all policies.....	182
The Table of Policies.....	182
Policy names.....	183
Configuring the Name field.....	183
Disabling policy name requirement.....	183
IPv4 policy.....	184
To configure a IPv4 policy in the GUI.....	184
IPv6 policy.....	189
To configure a IPv6 policy in the GUI.....	189
NAT64 policy.....	191
To configure a NAT64 policy in the GUI.....	191
NAT46 policy.....	193
To configure a NAT46 policy in the GUI.....	193
Central SNAT.....	194
To configure a Central SNAT entry in the GUI.....	195
To configure Central SNAT in the CLI.....	196
IPv4 access control list.....	197
To configure a IPv4 access control list entry in the GUI.....	197
To configure a IPv4 access control list entry in the CLI.....	198
IPv6 access control list.....	198
To configure a IPv6 access control list entry in the GUI.....	198
To configure a IPv6 access control list entry in the CLI.....	199
IPv4 DoS policy.....	199
To configure a IPv4 DoS policy in the GUI.....	199
Example.....	200
IPv6 DoS policy.....	204
To configure a IPv6 DoS policy in the GUI.....	204
Multicast policy.....	205
SSL mirroring for policies.....	206
Object configuration.....	208
Addresses.....	208
Interfaces.....	209
IPv4 addresses.....	210
FQDN addresses.....	211
Verification.....	212
Geography based addresses.....	213
IP range addresses.....	215

IP / netmask addresses.....	216
Wildcard addressing.....	218
Wildcard FQDN.....	221
IPv6 addresses.....	223
Subnet addresses.....	223
IPv6 FQDN firewall addresses.....	224
Firewall IPv6 address templates.....	224
Multicast addresses.....	226
Multicast IP range.....	226
Broadcast subnet.....	228
Multicast IP addresses.....	228
Proxy addresses.....	229
Internet services.....	232
Address groups.....	233
Virtual IPs.....	234
Creating a virtual IP.....	236
Dynamic VIP according to DNS translation.....	241
Virtual IP groups.....	242
Configuring IP pools.....	242
Creating a IPv4 pool.....	243
Creating a IPv6 pool.....	247
Creating NAT46 IP pool and multiple (secondary) NAT64 prefixes.....	247
Services.....	248
Categories.....	249
Configuring a new service.....	250
Specific addresses in TCP/UDP/SCTP.....	255
Service groups.....	255
Firewall schedules.....	256
One-time schedule object.....	256
Recurring schedule object.....	258
Schedule groups.....	260
Schedule expiration.....	260
Secure web gateway, WAN optimization, web caching and WCCP.....	262
Before you begin.....	262
FortiGate models that support WAN optimization.....	263
Distributing WAN optimization, explicit proxy, and web caching to multiple CPU cores.....	263
Dispatching traffic to WAD worker based on source affinity.....	263
Toggling disk usage for logging or wan-opt.....	264
Example topologies relevant to WAN optimization.....	266
Basic WAN optimization topology.....	266
Out-of-path WAN optimization topology.....	267
Topology for multiple networks.....	268

WAN optimization with web caching.....	269
Explicit web proxy topologies.....	269
Explicit FTP proxy topologies.....	270
Web caching topologies.....	270
WCCP topologies.....	271
Inside FortiOS: WAN optimization.....	272
Centralize without compromising your WAN performance.....	272
FortiOS WAN optimization.....	272
Protocol optimization.....	272
Web caching.....	273
Byte caching.....	273
Dynamic data chunking.....	274
Data deduplication.....	274
Server monitoring and management.....	274
SSL acceleration.....	274
VPN replacement.....	274
Road warriors and home workers.....	275
WAN optimization concepts.....	276
Client/server architecture.....	276
WAN optimization peers.....	277
Protocol optimization.....	277
Protocol optimization and MAPI.....	278
Byte caching.....	278
Dynamic data chunking for byte caching.....	279
WAN optimization transparent mode.....	279
Configuring transparent mode.....	279
Operating modes and VDOMs.....	280
WAN optimization tunnels.....	280
Tunnel sharing.....	281
WAN optimization and user and device identity policies, load balancing and traffic shaping.....	282
Traffic shaping.....	282
WAN optimization and HA.....	282
WAN optimization, web caching and memory usage.....	283
WAN optimization configuration.....	284
Manual (peer-to-peer) and active-passive WAN optimization.....	284
Manual (peer to peer) configurations.....	284
Active-passive configurations.....	285
WAN optimization profiles.....	286
Processing non-HTTP sessions accepted by a WAN optimization profile with HTTP optimization.....	288
Processing unknown HTTP sessions.....	288
Monitoring WAN optimization performance.....	289

Traffic summary.....	290
Bandwidth optimization.....	290
WAN optimization configuration summary.....	290
Client-side configuration summary.....	290
server-side configuration summary.....	292
WANopt storage.....	293
WANopt cache service.....	294
Video caching.....	295
Best practices.....	297
Example basic manual (peer-to-peer) WAN optimization configuration.....	297
Network topology and assumptions.....	298
General configuration steps.....	298
Configuring basic peer-to-peer WAN optimization - web-based manager.....	298
Configuring basic peer-to-peer WAN optimization - CLI.....	301
Testing and troubleshooting the configuration.....	302
Example active-passive WAN optimization.....	304
Network topology and assumptions.....	304
General configuration steps.....	305
Configuring basic active-passive WAN optimization - web-based manager.....	305
Configuring basic active-passive WAN optimization - CLI.....	308
Testing and troubleshooting the configuration.....	310
Example adding secure tunneling to an active-passive WAN optimization configuration.....	311
Network topology and assumptions.....	312
General configuration steps.....	312
Configuring WAN optimization with secure tunneling - web-based manager.....	313
Configuring WAN optimization with secure tunneling - CLI.....	316
Peers and authentication groups.....	319
Basic WAN optimization peer requirements.....	319
Accepting any peers.....	319
How FortiGate units process tunnel requests for peer authentication.....	319
Configuring peers.....	320
Configuring authentication groups.....	321
Secure tunneling.....	323
Monitoring WAN optimization peer performance.....	324
Web cache concepts.....	325
Turning on web caching for HTTP and HTTPS traffic.....	325
Turning on web caching for HTTPS traffic.....	326
Full mode SSL server configuration.....	327
Half mode SSL server configuration.....	328
Changing the ports on which to look for HTTP and HTTPS traffic to cache.....	329
Web caching and HA.....	329
Web caching and memory usage.....	329

Changing web cache settings.....	329
Always revalidate.....	330
Max cache object size.....	330
Negative response duration.....	330
Fresh factor.....	330
Max TTL.....	330
Min TTL.....	331
Default TTL.....	331
Proxy FQDN.....	331
Max HTTP request length.....	331
Max HTTP message length.....	331
Ignore.....	331
Cache expired objects.....	332
Revalidated pragma-no-cache.....	332
Web cache configuration.....	333
Forwarding URLs to forwarding servers and exempting web sites from web caching.....	333
Forwarding URLs and URL patterns to forwarding servers.....	333
Exempting web sites from web caching.....	333
Monitoring web caching performance.....	335
Example web caching of HTTP and HTTPS Internet content for users on an internal network.....	335
Network topology and assumptions.....	335
Example reverse proxy web caching and SSL offloading for an Internet web server using a static one-to-one virtual IP.....	338
Network topology and assumptions.....	338
General configuration steps.....	339
Configuration steps - web-based manager.....	340
Configuration steps - CLI.....	341
Using a FortiCache as a cache service.....	342
WCCP concepts.....	344
WCCP Cisco to FortiGate client using L2-forwarding tunneling.....	344
WCCP configuration.....	345
WCCP configuration overview.....	345
WCCP service groups, service numbers, service IDs and well known services.....	345
Example WCCP server and client configuration for caching HTTP sessions (service ID = 0).....	346
Example WCCP server and client configuration for caching HTTPS sessions.....	347
Example WCCP server and client configuration for caching HTTP and HTTPS sessions.....	347
Other WCCP service group options.....	348
Example caching HTTP sessions on port 80 using WCCP.....	349
Configuring the WCCP server (WCCP_srv).....	349

Configuring the WCCP client (WCCP_client).....	351
Example caching HTTP sessions on port 80 and HTTPS sessions on port 443 using WCCP.....	351
Configuring the WCCP server (WCCP_srv).....	352
Configuring the WCCP client (WCCP_client).....	353
WCCP packet flow.....	354
Configuring the forward and return methods and adding authentication.....	354
WCCP messages.....	355
Troubleshooting WCCP.....	355
Real time debugging.....	355
Application debugging.....	355
Web proxy concepts.....	357
Proxy policy.....	357
Proxy authentication.....	357
Matching.....	357
Processing policies for authentication.....	358
CLI syntax.....	358
Configuring authentication in transparent proxy.....	360
Proxy addresses.....	361
Proxy address group.....	361
Web proxy firewall services and service groups.....	361
Learn client IP.....	362
Example.....	363
Web proxy configuration.....	365
General web proxy configuration steps.....	365
Logging options in web proxy profiles.....	369
Policy matching based on referrer headers and query strings.....	370
Multiple web proxy PAC files in one VDOM.....	372
Explicit proxy concepts.....	374
The FortiGate explicit web proxy.....	374
Other explicit web proxy options.....	375
HTTP port, HTTPS port, FTP port, PAC port.....	376
Proxy FQDN.....	376
Max HTTP request length.....	376
Max HTTP message length.....	376
Multiple incoming ports and port ranges.....	376
Internet services.....	377
IP pools.....	377
Proxy chaining (web proxy forwarding servers).....	377
Adding a web proxy forwarding server.....	377
Web proxy forwarding server monitoring and health checking.....	378
Grouping forwarding servers and load balancing traffic to them.....	379

Adding proxy chaining to an explicit web proxy policy.....	380
Security profiles, threat weight, device identification, and the explicit web proxy.....	381
Explicit web proxy sessions and user limits.....	381
Explicit proxy configuration.....	384
Configuring an external IP address for the IPv4 explicit web proxy.....	384
Configuring an external IP address for the IPv6 explicit web proxy.....	384
Restricting the IP address of the IPv4 explicit web proxy.....	384
Restricting the outgoing source IP address of the IPv4 explicit web proxy.....	384
Restricting the IP address of the explicit IPv6 web proxy.....	385
Restricting the outgoing source IP address of the IPv6 explicit web proxy.....	385
Explicit proxy firewall address types.....	385
Proxy auto-config (PAC) configuration.....	386
PAC file content.....	386
Unknown HTTP version.....	387
Authentication realm.....	387
Implementing botnet features.....	387
Adding disclaimer messages to explicit proxy policies.....	387
Changing HTTP headers.....	388
Preventing the explicit web proxy from changing source addresses.....	388
Example users on an internal network browsing the Internet through the explicit web proxy with web caching, RADIUS authentication, web filtering, and virus scanning.....	389
General configuration steps.....	389
Configuring the explicit web proxy - web-based manager.....	390
Configuring the explicit web proxy - CLI.....	391
Testing and troubleshooting the configuration.....	393
Kerberos and NTLM authentication.....	393
Kerberos authentication for explicit proxy users.....	394
Enhancements to Kerberos explicit and transparent web proxy.....	394
Transparent web-proxy Kerberos authentication.....	400
Transparent proxy concepts.....	404
More about the transparent proxy.....	404
Flat policies.....	404
Authentication.....	404
New proxy type.....	405
IP pools support.....	405
SOCKSv5.....	405
Forwarding.....	405
Support for explicit proxy address objects & groups into IPv4 firewall policies.....	406
Support application service in the proxy based on HTTP requests.....	406
Transparent proxy configuration.....	407
CLI changes due to addition of transparent proxy.....	409
FTP proxy concepts.....	411

The FortiGate explicit FTP proxy.....	411
How to use the explicit FTP proxy to connect to an FTP server.....	412
Security profiles, threat weight, device identification, and the explicit FTP proxy.....	414
Explicit FTP proxy options and SSL/SSH inspection.....	415
Explicit FTP proxy sessions and antivirus.....	415
Explicit FTP proxy sessions and user limits.....	415
FTP proxy configuration.....	416
General explicit FTP proxy configuration steps.....	416
Restricting the IP address of the explicit FTP proxy.....	418
Restricting the outgoing source IP address of the explicit FTP proxy.....	419
Example users on an internal network connecting to FTP servers on the Internet through the explicit FTP with RADIUS authentication and virus scanning.....	419
General configuration steps.....	419
Configuring the explicit FTP proxy - web-based manager.....	420
Configuring the explicit FTP proxy - CLI.....	421
Testing and troubleshooting the configuration.....	423
Diagnose commands for WAN optimization.....	424
get test {wad wccpd} <test_level>.....	424
Examples.....	424
diagnose wad.....	425
Example diagnose wad tunnel list.....	426
Example diagnose wad webcache list.....	427
diagnose wad csvc.....	428
diagnose wad worker.....	429
diagnose wacs.....	429
diagnose wadbd.....	430
diagnose debug application {wad wccpd} [<debug_level>].....	430
diagnose test application wad 2200.....	431

Change log

Date	Change description
July 26, 2018	FortiOS 6.0.2 document release. Minor updates added. See Wildcard addressing on page 218
June 5, 2018	FortiOS 6.0.1 document release. See What's new for Firewall in 6.0.1 on page 19 .
March 29, 2018	FortiOS 6.0 document release. See "What's new for Firewall in 6.0" on page 19 .

Introduction

Welcome and thank you for selecting Fortinet products for your network protection. This document is intended to provide the concepts and techniques that will be needed to configure the FortiGate firewall on your FortiGate unit.

Before you start administrating your FortiGate device, certain assumptions have been made in the writing of this manual:

- You have administrative access to the Web based GUI or to the Command Line Interface.
- The FortiGate unit is integrated into your network.
- The operation mode (NAT or Transparent) has been configured.
- Network Interfaces have been configured.
- DNS settings have been configured.
- The system time settings have been configured.
- Firmware is up to date.
- FortiGuard Service licenses are current and the device is able to connect to the FortiGuard Servers.
- If you are using FortiCloud, it is properly configured.

How this guide is organized

This guide contains a number of different topic that, at its simplest, can be grouped into fundamental firewall topics such as policies, objects and network defense and topics that have to do with the optimization of the firewall such as WAN optimization, proxies and caching.

Fundamentals

"Firewall concepts" on page 21 explains the ideas behind the components, techniques and processes that are involved in setting up and running a firewall in general and the FortiGate firewall in particular. The premise here is that regardless of how experienced someone is with firewalls as they go through the process of configuring a firewall that is new to them they are likely to come across a term or setting that they may not be familiar with even if it is only in the context of the setting they are working in at the moment. FortiGate firewall are quite comprehensive and can be very granular in the functions that they perform, so it makes sense to have a consistent frame of reference for the ideas that we will be working with.

Some examples of the concepts that will be addressed here are:

- "What is a Firewall?"
- "NAT"
- "IPv6"

"Firewall objects" describes the following firewall objects:

- Addressing
- Services
- Firewall Policies

"Network defense" on page 166 describes various methods of defending your Network using the abilities of the FortiGate Firewall.

"Object configuration" on page 208 is similar to a cookbook in that it will refer to a number of common tasks that you will likely perform to get the full functionality out of your FortiGate firewall. Because of the way that firewall are designed, performing many of the tasks requires that firewall components be set up in a number of different sections of the interface and be configured to work together to achieve the desired result. This section will bring those components all together as a straight forward series of instructions.

FortiGate firewall components

The FortiGate firewall is made up of a number of different components that are used to build an impressive list of features that have flexibility of scope and granularity of control that provide protection that is beyond that provided by the basic firewalls of the past.

Some of the components that FortiOS uses to build features are:

- Interfaces
- VLANs
- Soft Switches
- Zones
- Predefined Addresses
- IP address based
- FQDN based
- Geography based
- Access Schedules
- Authentication
- Local User based
- Authentication Server based (Active Directory, Radius, LDAP)
- Device Based
- Configureable Services
- IPv4 and IPv6 protocol support

The features of FortiOS include but are not limited to:

- Security profiles, sometimes referred to as Unified Threat Management (UTM) or Next Generation Firewall (NGFW)
- Predefined firewall addresses (this includes IPv4 and IPv6, IP pools,.. wildcard addresses and netmasks, and geography-based addresses)
- Monitoring traffic
- Traffic shaping and per-IP traffic shaping (advanced)
- Firewall schedules
- Services (such as AOL, DHCP and FTP)
- Logging traffic
- Quality of Service (QoS)
- Identity-based policies
- Endpoint security

Firewall optimization

There are a few different methodologies of optimization and most of these methodologies has been divided into:

- Concepts section - This will have the basic ideas behind the how and why of the topic. Because the number of topics is larger, the ideas are not as pervasive and the content is not so extensive as in the Fundamentals section, some of the topics will include instructions on the configuration of that individual topic in order to keep the information for granular topics together.
- Configuration section- Just like the Configuration section of the Fundamentals, this will be a cookbook style of documentation showing how to configure something that achieves a specific functionality from the FortiGate.

The optimization topics include:

- Secure web gateway, WAN optimization, web caching and WCCP
- Example topologies relevant to WAN optimization
- Inside FortiOS: WAN optimization
- WAN optimization concepts
- WAN optimization configuration
- Peers and authentication groups
- Web cache concepts
- Web cache configuration
- WCCP concepts
- WCCP configuration
- Web proxy concepts
- Web proxy configuration
- Explicit proxy concepts
- Explicit proxy configuration
- Transparent proxy concepts
- Transparent proxy configuration
- FTP proxy concepts
- FTP proxy configuration
- Diagnose commands for WAN optimization

How does a FortiGate protect your network?

The FortiGate firewall protects your network by taking the various components and using them together to build a kind of wall or access control point so that anyone that is not supposed to be on your network is prevented from accessing your network in anyway other than those approved by you. It also protects your network from itself by keeping things that shouldn't happen from happening and optimizing the flow of traffic so that the network is protected from traffic congestion that would otherwise impede traffic flow.

Most people have at one time or another played with the children's toy system that is made up of interlocking blocks. The blocks come in different shapes and sizes so that you can build structures to suit your needs and in your way. The components of the FortiGate firewall are similar. You are not forced to use all of the blocks all of the time. You mix and match them to get the results that you are looking for. You can build a very basic structure

that's only function is to direct traffic in and out to the correct subnets or you can build a fortress that only allows specific traffic to specific hosts from specific hosts at specific times of day and that is only if they provide the credentials that have been pre-approved and all of the traffic is encrypted so that even when the traffic is out on the Internet it is private from the world. Just like the interlocking blocks, what you build is up to you, but chances are if you put them together the right way there isn't much that can't be built.

Here is one example of how the components could be put together to support the requirements of a network infrastructure design.

- Off the Internal interface you could have separate VLANs. One for each for the departments of Sales, Marketing and Engineering so that the traffic from the users on one VLAN does not intrude upon the hosts of the other VLANs and the department are isolated from one another for security reasons.
- To ease in the administration each of the VLAN sub-interfaces is made a member of a zone so that security policies that apply to all of the hosts on all of the VLANs can be applied to all of them at once.
- Using the addresses component each of the IP address ranges could be assigned a user friendly name so that they could be referred to individually and then for policies that would refer to them all as a whole the individual ranges to be made members of an address group.
- Firewall schedules could be created to address the differing needs of each of the groups so that Sales and Marketing could be allowed access to the Internet during regular business hours and the Engineering department could be allowed access during the lunch break.
- By setting up the outgoing policies to use FortiGuard Web-filtering the employees could be prevented from visiting inappropriate sites and thus enforcing the policies of the HR department.
- A couple of virtual IP addresses with port forwarding could be configured to allow users on the Internet to access a web server on the DMZ subnet using the company's only Public IP address without affecting the traffic that goes to the company's mail server that is hosted on a complete different computer.
- Even though the Web server on the same DMZ has an FTP service to allow for the uploading of web pages to the web server from the Marketing and Engineer teams, by placing a DENY policy on any FTP traffic from the Internet malicious users are prevented from abusing the FTP service.
- By monitoring the traffic as it goes through the policies you can verify that the policies are in working order.
- By using a combination of ALLOW and DENY policies and placing them in the correct order you could arrange for an outside contractor to be allowed to update the web site as well

These set of configurations is not extensive but it does give an idea of how different components can be mixed and matched to build a configuration that meets an organization's needs but at the same time protect it from security risks.

What's new for Firewall in 6.0.1

The following list contains new firewall features added in FortiOS 6.0. Click on a link to navigate to that section for further information.

- [IPv6 Neighbor Discovery Proxy on page 136](#)

What's new for Firewall in 6.0

The following list contains new firewall features added in FortiOS 6.0. Click on a link to navigate to that section for further information.

- SSH MITM deep inspection on page 53
- Creating NAT46 IP pool and multiple (secondary) NAT64 prefixes on page 247
- Application groups for NGFW policies on page 31
- Wildcard FQDNs for SSL deep inspection exemptions on page 222
- IPv6 FQDN firewall addresses on page 224
- Firewall IPv6 address templates on page 224
- Port block allocation timeout on page 245
- WAN Optimization and web cache improvements. Changes can be found at:
 - [WANopt storage](#) on page 293
 - [WANopt cache service](#) on page 294
 - [Video caching](#) on page 295
 - [diagnose wad csvc](#) on page 428
 - [diagnose wad worker](#) on page 429
- SSL mirroring for policies on page 206
- ISDB and IRDB in firewall policies on page 188
- IPv6 support for GRE tunnels on page 131
- Dispatching traffic to WAD worker based on source affinity on page 263
- Explicit proxy authentication timeout on page 1
- Multiple web proxy PAC files in one VDOM on page 372
- Logging options in web proxy profiles on page 369
- Encryption strength for proxied SSH sessions on page 57
- DNS traffic in NGFW policy-mode on page 109
- Using a FortiCache as a cache service on page 342
- IPv6 SSH on page 140
- WCCP Cisco to FortiGate client using L2-forwarding tunneling on page 344

Firewall concepts

There are a number of foundational concepts that are necessary to have a grasp of before delving into the details of how the FortiGate firewall works. Some of these concepts are consistent throughout the firewall industry and some of them are specific to more advanced firewalls such as the FortiGate. Having a solid grasp of these ideas and terms can give you a better idea of what your FortiGate firewall is capable of and how it will be able to fit within your networks architecture.

This chapter describes the following firewall concepts:

- [What is a firewall?](#)
- [FortiGate modes](#)
- [How packets are handled by FortiOS](#)
- [Interfaces and zones](#)
- [Access control lists](#)
- [IPv6](#)
- [NAT](#)
- ["IP pools" on page 71](#)
- [Services and TCP ports](#)
- ["Firewall policies" on page 29](#)
- ["SSL/SSH inspection" on page 52](#)
- ["VPN policies" on page 100](#)
- ["DSRI" on page 100](#)
- ["Interface policies" on page 101](#)
- ["Local-In policies" on page 108](#)
- ["Security policy 0" on page 109](#)
- ["Deny policies" on page 110](#)
- ["Accept policies" on page 110](#)
- [IPv6 Policies](#)
- ["Fixed port" on page 110](#)
- ["Endpoint security" on page 111](#)
- ["Traffic logging" on page 111](#)

What is a firewall?

The term firewall originally referred to a wall intended to confine a fire or potential fire within a building. Later uses refer to similar structures, such as the metal sheet separating the engine compartment of a vehicle or aircraft from the passenger compartment.

A firewall can either be software-based or hardware-based and is used to help keep a network secure. Its primary objective is to control the incoming and outgoing network traffic by analyzing the data packets and determining whether it should be allowed through or not, based on a predetermined rule set. A network's firewall builds a

bridge between an internal network that is assumed to be secure and trusted, and another network, usually an external (inter)network, such as the Internet, that is not assumed to be secure and trusted.

Network layer or packet filter firewalls

Stateless firewalls

Stateless firewalls are the oldest form of these firewalls. They are faster and simple in design requiring less memory because they process each packet individually and don't require the resources necessary to hold onto packets like stateful firewalls. Stateful firewalls inspect each packet individually and check to see if it matches a predetermined set of rules. According to the matching rule the packet is either be allowed, dropped or rejected. In the case of a rejection an error message is sent to the source of the traffic. Each packet is inspected in isolation and information is only gathered from the packet itself. Simply put, if the packets were not specifically allowed according to the list of rules held by the firewall they were not getting through.

Stateful Firewalls

Stateful firewalls retain packets in memory so that they can maintain context about active sessions and make judgments about the state of an incoming packet's connection. This enables Stateful firewalls to determine if a packet is the start of a new connection, a part of an existing connection, or not part of any connection. If a packet is part of an existing connection based on comparison with the firewall's state table, it will be allowed to pass without further processing. If a packet does not match an existing connection, it will be evaluated according to the rules set for new connections. Predetermined rules are used in the same way as a stateless firewall but they can now work with the additional criteria of the state of the connection to the firewall.

Best Practices Tip for improving performance:

Blocking the packets in a denied session can take more cpu processing resources than passing the traffic through. By putting denied sessions in the session table, they can be kept track of in the same way that allowed session are so that the FortiGate unit does not have to redetermine whether or not to deny all of the packets of a session individually. If the session is denied all packets of that session are also denied.



In order to configure this you will need to use 2 CLI commands

```
config system setting
    set ses-denied-traffic enable
    set block-session-timer <integer 1 - 300> (this determines in
seconds how long, in seconds, the session is kept in the table)
end
```

Application layer firewalls

Application layer filtering is yet another approach and as the name implies it works primarily on the Application Layer of the OSI Model.

Application Layer Firewalls actually, for lack of a better term, understand certain applications and protocols. Examples would be FTP, DNS and HTTP. This form of filtration is able to check to see if the packets are actually behaving incorrectly or if the packets have been incorrectly formatted for the protocol that is indicated. This

process also allows for the use of deep packet inspection and the sharing of functionality with Intrusion Prevention Systems (IPS).

Application-layer firewalls work on the application level of the TCP/IP stack (i.e., all browser traffic, or all telnet or ftp traffic), and may intercept all packets traveling to or from an application. They block other packets (usually dropping them without acknowledgment to the sender). Application firewalls work much like a packet filter but application filters apply filtering rules (allow/block) on a per process basis instead of filtering connections on a per port basis.

On inspecting all packets for improper content, firewalls can restrict or prevent outright the spread of networked computer worms and trojans. The additional inspection criteria can add extra latency to the forwarding of packets to their destination.

Proxy servers

A proxy server is an appliance or application that acts as an intermediary for communicating between computers. A computer has a request for information. The packets are sent to the designated resource but before they can get there they are blocked by the proxy server saying that it will take the request and pass it on. The Proxy Server processes the request and if it is valid it passes onto the designated computer. The designated computer gets the packet and processes the request, sending the answer back to the proxy server. The proxy server sends the information back to the originating computer. It's all a little like a situation with two people who refuse to talk directly with each other using someone else to take messages back and forth.

From a security stand point a Proxy Server can serve a few purposes:

- Protects the anonymity of the originating computer
- The two computers never deal directly with each other
- Packets that are not configured to be forwarded are dropped before reaching the destination computer.
- If malicious code is sent it will affect the Proxy server without affecting the originating or sending computer.

Proxies can perform a number of roles including:

- Content Filtering
- Caching
- DNS proxy
- Bypassing Filters and Censorship
- Logging and eavesdropping
- Gateways to private networks
- Accessing service anonymously

UTM/ NGFW

Unified Threat Management and Next Generation Firewall are terms originally coined by market research firms and refer to the concept of a comprehensive security solution provided in a single package. It is basically combining of what used to be accomplished by a number of different security technologies all under a single umbrella or in this case, a single device. On the FortiGate firewall this is achieved by the use of Security Profiles and optimized hardware.

In effect it is going from a previous style of firewall that included among its features:

- Gateway Network Firewall
- Routing

- VPN

To a more complete system that includes:

- Gateway Network Firewall
- Routing
- VPN
- Traffic Optimization
- Proxy Services
- Content Filtering
- Application Control
- Intrusion Prevention
- Denial of Service Attack Protection
- Anti-virus
- Anti-spam
- Data Leak Prevention
- Endpoint Control of Security Applications
- Load Balancing
- WiFi Access Management
- Authentication Integration into Gateway Security
- Logging
- Reporting

Advantages of using security profiles

- Avoidance of multiple installations.
- Hardware requirements are fewer.
- Fewer hardware maintenance requirements.
- Less space required.
- Compatibility - multiple installations of products increase the probability of incompatibility between systems.
- Easier support and management.
- There is only one product to learn therefore a reduced requirement of technical knowledge.
- Only a single vendor so there are fewer support contracts and Service Level Agreements.
- Easier to incorporate into existing security architecture.
- Plug and play architecture.
- Web based GUI for administration.

FortiGate modes

The FortiGate unit has a choice of modes that it can be used in, either NAT mode or transparent mode. The FortiGate unit is able to operate as a firewall in both modes, but some of its features are limited in transparent mode. It is always best to choose which mode you are going to be using at the beginning of the set up. Once you start configuring the device, if you want to change the mode you are going to lose all configuration settings in the change process.

NAT mode

NAT mode is the most commonly used mode by a significant margin and is thus the default setting on the device. As the name implies the function of NAT is commonly used in this mode and is easily configured but there is no requirement to use NAT. The FortiGate unit performs network address translation before IP packets are sent to the destination network.

These are some of the characteristics of NAT mode:

- Typically used when the FortiGate unit is a gateway between private and public networks.
- Can act as a router between multiple networks within a network infrastructure.
- When used, the FortiGate unit is visible to the networks that is connected to.
- Each logical interface is on a distinct subnet.
- Each Interface needs to be assigned a valid IP address for the subnet that it is connected to it.

Transparent mode

Transparent mode is so named because the device is effectively transparent in that it does not appear on the network in the way that other network devices show as a nodes in the path of network traffic. Transparent mode is typically used to apply the FortiOS features such as Security Profiles etc. on a private network where the FortiGate unit will be behind an existing firewall or router.

These are some of the characteristics of transparent mode:

- The FortiGate unit is invisible to the network.
- All of its interfaces are on the same subnet and share the same IP address.
- The FortiGate unit uses a Management IP address for the purposes of Administration.
- Still able to use NAT to a degree, but the configuration is less straightforward

In transparent mode, you can also perform NAT by creating a security policy or policies that translates the source addresses of packets passing through the FortiGate unit as well as virtual IP addresses and/or IP pools.

How packets are handled by FortiOS

To give you idea of what happens to a packet as it makes its way through the FortiGate unit here is a brief overview. This particular trip of the packet is starting on the Internet side of the FortiGate firewall and ends with the packet exiting to the Internal network. An outbound trip would be similar. At any point in the path if the packet is going through what would be considered a filtering process and if fails the filter check the packet is dropped and does not continue any further down the path.

This information is covered in more detail in other in the Troubleshooting chapter of the FortiOS Handbook in the Life of a Packet section.

The incoming packet arrives at the external interface. This process of entering the device is referred to as **ingress**.

Step #1 - Ingress

1. Denial of Service Sensor
2. IP integrity header checking

3. IPsec connection check
4. Destination NAT
5. Routing

Step #2 - Stateful inspection engine

1. Session Helpers
2. Management Traffic
3. SSL VPN
4. User Authentication
5. Traffic Shaping
6. Session Tracking
7. Policy lookup

Step #3 - Security profiles scanning process

1. Flow-based Inspection Engine
2. IPS
3. Application Control
4. Data Leak Prevention
5. Email Filter
6. Web Filter
7. Anti-virus
8. Proxy-based Inspection Engine
9. VoIP Inspection
10. Data Leak Prevention
11. Email Filter
12. Web Filter
13. Anti-virus
14. ICAP

Step #4 - Egress

1. IPsec
2. Source NAT
3. Routing

Interfaces and zones

A Firewall is a gateway device that may be the nexus point for more than 2 networks. The interface that the traffic is coming in on and should be going out on is a fundamental concern for the purposes of routing as well as security. Routing, policies and addresses are all associated with interfaces. The interface is essentially the connection point of a subnet to the FortiGate unit and once connected can be connected to other subnets.

The following types of interfaces are found on a FortiGate:

- Interface , this can refer to a physical or virtual interface
- Zone
- Virtual Wired Pair

Interfaces

Physical interfaces or not the only ones that need to be considered. There are also virtual interfaces that can be applied to security policies. VLANs are one such virtual interface. Interfaces if certain VPN tunnels are another.

Policies are the foundation of the traffic control in a firewall and the Interfaces and addressing is the foundation that policies are based upon. Using the identity of the interface that the traffic connects to the FortiGate unit tells the firewall the initial direction of the traffic. The direction of the traffic is one of the determining factors in deciding how the traffic should be dealt with. You can tell that interfaces are a fundamental part of the policies because, by default, this is the criteria that the policies are sorted by.

Zones

Zones are a mechanism that was created to help in the administration of the firewalls. If you have a FortiGate unit with a large number of ports and a large number of nodes in your network the chances are high that there is going to be some duplication of policies. Zones provide the option of logically grouping multiple virtual and physical FortiGate firewall interfaces. The zones can then be used to apply security policies to control the incoming and outgoing traffic on those interfaces. This helps to keep the administration of the firewall simple and maintain consistency.

For example you may have several floors of people and each of the port interfaces could go to a separate floor where it connects to a switch controlling a different subnet. The people may be on different subnets but in terms of security they have the same requirements. If there were 4 floors and 4 interfaces a separate policy would have to be written for each floor to be allowed out onto the Internet off the WAN1 interface. This is not too bad if that is all that is being done, but now start adding the use of more complicated policy scenarios with Security Profiles, then throw in a number of Identity based issues and then add the complication that people in that organization tend to move around in that building between floors with their notebook computers.

Each time a policy is created for each of those floors there is a chance of an inconsistency cropping up. Rather than make up an additional duplicate set of policies for each floor, a zone can be created that combines multiple interfaces. And then a single policy can be created that uses that zone as one side of the traffic connection.



You cannot add physical interfaces associated with VLANs to zones.

Virtual wire pair

The simplified explanation is that two interfaces are set up so that whatever traffic goes through one of the pair is replicated on the other. They are most commonly used when scanning is needed on an interface without interfering with the traffic. On interface "A", everything goes through unaffected. The replicated traffic on interface "B" is sent to an analyzer of some kind and the traffic can be thoroughly scanned without worry of impacting performance.

When two physical interfaces are setup as a Virtual Wire Pair, they will have no IP addressing and are treated similar to a transparent mode VDOM. All packets accepted by one of the interfaces in a virtual wire pair can only

exit the FortiGate through the other interface in the virtual wire pair and only if allowed by a virtual wire pair firewall policy. Packets arriving on other interfaces cannot be routed to the interfaces in a virtual wire pair. A FortiGate can have multiple virtual wire pairs.

You cannot add VLANs to virtual wire pairs. However, you can enable wildcard VLANs for a virtual wire pair. This means that all VLAN-tagged traffic can pass through the virtual wire pair if allowed by virtual wire pair firewall policies.

Access control lists

Access control lists (ACLs) in the FortiOS firmware could be considered a granular or more specifically targeted blacklist. These ACLs drop IPv4 or IPv6 packets at the physical network interface before the packets are analyzed by the CPU. On a busy appliance this can really help the performance.

The ACL feature is available on FortiGates with NP6-accelerated interfaces. ACL checking is one of the first things that happens to the packet and checking is done by the NP6 processor. The result is very efficient protection that does not use CPU or memory resources.

Incoming interfaces

The configuration of the Access Control List allow you to specify which interface the ACL will be applied to. There is a hardware limitation that needs to be taken into account. The ACL is a Layer 2 function and is offloaded to the ISF hardware, therefore no CPU resources are used in the processing of the ACL. It is handled by the inside switch chip which can do hardware acceleration, increasing the performance of the FortiGate. The drawback is that the ACL function is only supported on switch fabric driven interfaces. It also cannot be applied to hardware switch interfaces or their members. Ports such as WAN1 or WAN2 that are found on some models that use network cards that connect to the CPU through a PCIe bus will not support ACL.

Addresses

Because the address portion of an entry is based on a FortiGate address object, it can be any of the address types used by the FortiGate, including address ranges. There is further granularity by specifying both the source and destination addresses. The traffic is blocked not on an either or basis of these addresses but the combination of the two, so that they both have to be correct for the traffic to be denied. Of course, if you want to block all of the traffic from a specific address all you have to do is make the destination address "all".

Because the blocking takes place at the interface based on the information in the packet header and before any processing such as NAT can take place, a slightly different approach may be required. For instance, if you are trying to protect a VIP which has an external address of x.x.x.x and is forwarded to an internal address of y.y.y.y, the destination address that should be used is x.x.x.x, because that is the address that will be in the packet's header when it hits the incoming interface.

Services

Further granulation of the filter by which the traffic will be denied is done by specifying which service the traffic will use.

Firewall policies

The firewall policy is the axis around which most of the other features of the FortiGate firewall revolve. A large portion of the settings in the firewall at some point will end up relating to or being associated with the firewall policies and the traffic that they govern. Any traffic going through a FortiGate unit has to be associated with a policy. These policies are essentially discrete compartmentalized sets of instructions that control the traffic flow going through the firewall. These instructions control where the traffic goes, how it's processed, if it's processed and even whether or not it's allowed to pass through the FortiGate.

When the firewall receives a connection packet, it analyzes the packet's source address, destination address, and service (by port number). It also registers the incoming interface, the outgoing interface it will need to use and the time of day. Using this information the FortiGate firewall attempts to locate a security policy that matches the packet. If it finds a policy that matches the parameters it then looks at the action for that policy. If it is **ACCEPT** the traffic is allowed to proceed to the next step. If the Action is **DENY** or a match cannot be found the traffic is not allowed to proceed.

The 2 basic actions at the initial connection are either **ACCEPT** or **DENY**:

- If the **Action** is **ACCEPT**, the policy action permits communication sessions. There may be other packet processing instructions, such as requiring authentication to use the policy or restrictions on the source and destination of the traffic.
- If the **Action** is **DENY**, the policy action blocks communication sessions, and you can optionally log the denied traffic. If no security policy matches the traffic, the packets are dropped. A **DENY** security policy is needed when it is required to log the denied traffic, also called "violation traffic".

There are two other Actions that can be associated with the policy:

- **LEARN** - This is a specialized variation on the **ACCEPT** action. That is set up to allow traffic but to keep traffic logs so that the administrator can go through them to learn what kind of traffic has to be dealt with.
- **IPsec** - This is an **ACCEPT** action that is specifically for IPsec VPNs.

There can also be a number of instructions associated with a FortiGate firewall in addition to the **ACCEPT** or **DENY** actions, some of which are optional. Instructions on how to process the traffic can also include such things as:

- Logging Traffic
- Authentication
- Network Address Translation or Port Address Translation
- Use Virtual IPs or IP Pools
- Caching
- Whether the source of the traffic is based on address, user, device or a combination
- Whether to treat as regular traffic or IPsec traffic
- What certificates to use
- Security profiles to apply
- Proxy Options
- Traffic Shaping

Firewall policy parameters

As mentioned before, for traffic to flow through the FortiGate firewall there must be a policy that matches its parameters:

Incoming interface(s)

This is the interface or interfaces that the traffic is first connection to the FortiGate unit by. The exception being traffic that the FortiGate generates itself. This is not limited to the physical Ethernet ports found on the device. The incoming interface can also be a logical or virtual interface such as a VPN tunnel, a Virtual WAN link or a wireless interface.

Outgoing interface(s)

After the firewall has processed the traffic it needs to leave a port to get to its destination and this will be the interface or interfaces that the traffic leaves by. This interface, like the **Incoming Interface** is not limited to only physical interfaces.

Source address(es)

The addresses that a policy can receive traffic from can be wide open or tightly controlled. For a public web server that the world at large should be able to access, the best choice will be "all". If the destination is a private web server that only the branch offices of a company should be able to access or a list of internal computers that are the only ones allowed to access an external resource then a group of preconfigured addresses is the better strategy.

Additional parameters under the Source Address, though they are not mandatory are:

- **Source User(s)**

This parameter is based on a user identity that can be from a number of authentication authorities. It will be an account or group that has been set up in advance that can be selected from the drop down menu. The exception to this is the feature that allows the importing of LDAP Users. When the feature is used, a small wizard window will appear to guide the user through the setup. The caveat is that the LDAP server object in the **User and Device > Authentication > LDAP Servers** section has to be already configured to allow the use of this import feature.

- **Source Device Type**

This parameter is for narrowing down the traffic sending devices to those that the FortiGate is familiar with. Again the contents of this parameter need to be a preconfigured object and these are defined at **User and Device > Custom Devices & Groups**. This parameter can limit the devices that can connect to this policy to those specific MAC addresses that are already known by the FortiGate and are approved for the policy.

Destination address(es)

In the same way that the source address may need to be limited, the destination address can be used as a traffic filter. When the traffic is destined for internal resources the specific address of the resource can be defined to better protect the other resources on the network. One of the specialized destination address options is to use a Virtual IP address. The destination address doesn't need to be internal you can define policies that are only for connecting to specific addresses on the Internet.

Internet service(s)

In this context, an Internet service is a combination of one or more addresses and one or more services associated with a service found on the Internet such as an update service for software.

Schedule

The time frame that is applied to the policy. This can be something as simple as a time range that the sessions are allowed to start such as between 8:00 am and 5:00 pm. Something more complex like business hours that include a break for lunch and time of the session's initiation may need a schedule group because it will require multiple time ranges to make up the schedule.

Service

The service or service chosen here represent the TCP/IP suite port numbers that will most commonly be used to transport the named protocols or group of protocols. This will be a little different than Application Control which looks more closely at the packets to determine the actual protocol used to create them.

Without all six (possibly 8) of these things matching, the traffic will be declined. Each traffic flow requires a policy and the direction is important as well. Just because packets can go from point A to point B on port X does not mean that the traffic can flow from point B to point A on port X. A policy must be configured for each direction.

When designing a policy there is often reference to the traffic flow, but most communication is a two way connection so trying to determine the direction of the flow can be somewhat confusing. If traffic is HTTP web traffic the user sends a request to the web site, but most of the traffic flow will be coming from the web site to the user. Is the traffic flow considered to be from the user to the web site, the web site to the user or in both directions? For the purposes of determining the direction for a policy the important factor is the direction of the initiating communication. The user is sending a request to the web site so this is the initial communication and the web site is just responding to it so the traffic will be from the users network to the Internet.

A case where either side can initiate the communication like between two internal interfaces on the FortiGate unit would be a more likely situation to require a policy for each direction.

Application groups for NGFW policies

In addition to parameters like schedule and service, NGFW policies can filter by application or application category.



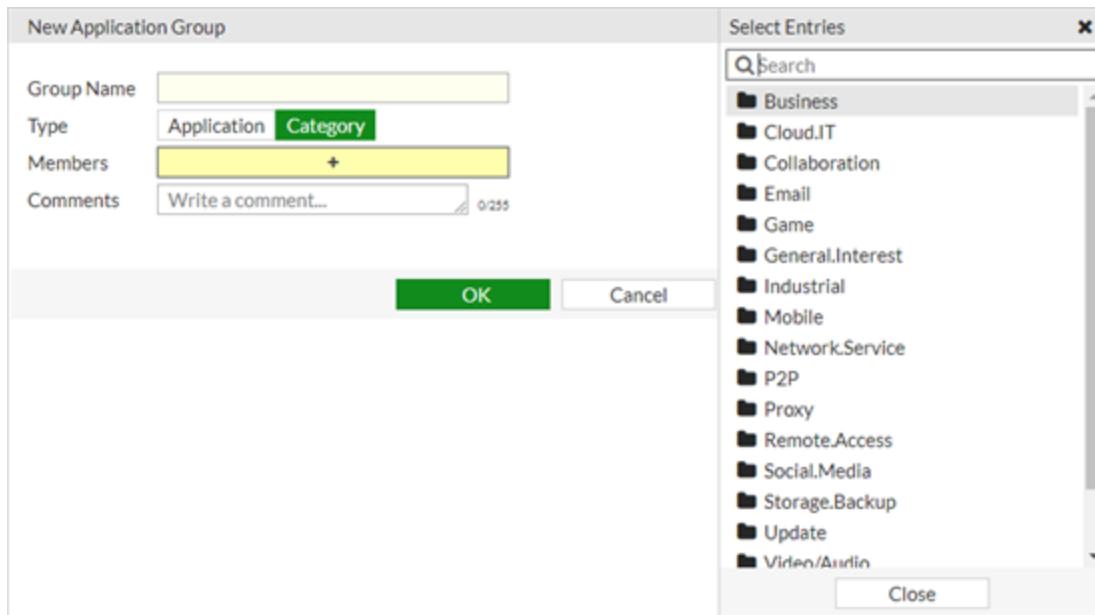
The use of this feature is dependent upon the VDOM having the following **System Operation Settings**:

- **Inspection Mode: Flow-based**
- **NGFW Mode: Policy-based**

To use the feature first create an application group in **Security Profiles > Custom Signatures**.

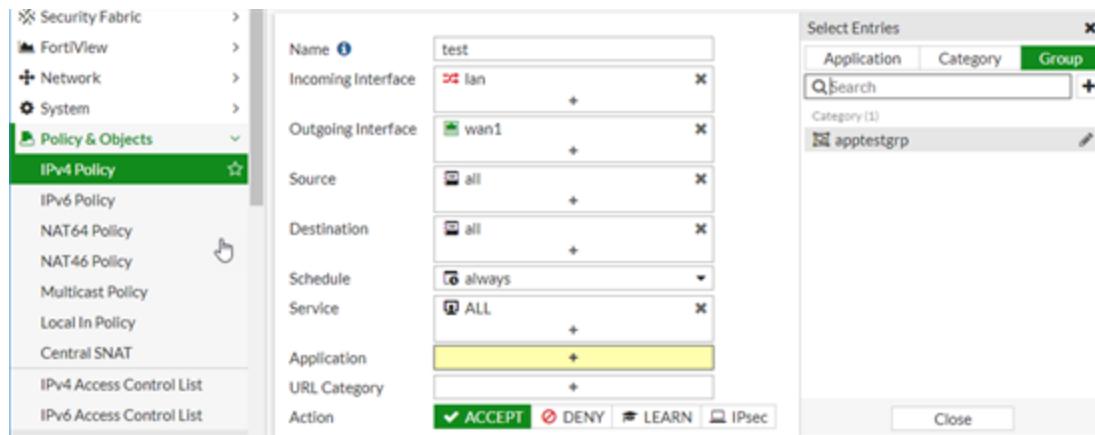
From the editing page for the **New Application Group**, choose a group type of **Application** and select individual applications for membership in the group.

Alternatively, select **Category** and add one or more application categories as group members.



Whichever type of **Application Group** you choose, the available **Members** will be displayed in the selection pane that slides out from the right of the window.

Once the **Application Group** is created, you can apply it to a policy in the **Application** field, by clicking on the **+** in the field and selecting members from the options under the **Group** tab at the top of the pane that slides out from the right of the window.



CLI

To create or edit an application group:

```
config application group
  edit <group_name>
    set comments
    set type {application | category}
    set application <Application ID number>
    set category <category ID number>
  end
```

To add an application group to a policy:

```
config firewall policy
  edit 1
    set app-group "test" "test1"
  end
```

Application ID number

In the CLI, you add applications to a group by using the application ID number. To see the list of application ID numbers, run the following command when type is set to application:

```
set application ? <enter>
```

The start of the list looks like:

```
set application

ID      Select Application ID

38614    1kxun
29025    1und1.Mail
17534    2ch
17535    2ch_Post
16284    3PC
16616    4shared
35760    4shared_File.Download
34742    4shared_File.Upload
44606    5ch
44607    5ch_Post
38923    8tracks
17045    9PFS
16554    126.Mail
23345    360.Safeguard.Update
35963    360.Yunpan
35967    360.Yunpan_File.Download
35966    360.Yunpan_File.Upload
42324    360.Yunpan_Login
16413    A.N
31529    ABC
...
```

Only the first 20 have been listed here.

Category ID number

The ID numbers for the categories in the CLI are found in the same manner as the applications. When the type is set to category, run the command:

```
set category ? <enter>
```

This list is shorter.

```
set category

ID      Select Category ID
2       P2P
3       VoIP
5       Video/Audio
6       Proxy
7       Remote.Access
8       Game
12      General.Interest
15      Network.Service
17      Update
21      Email
22      Storage.Backup
23      Social.Media
25      Web.Client
26      Industrial
28      Collaboration
29      Business
30      Cloud.IT
31      Mobile
```

What is not expressly allowed is denied

One of the fundamental ideas that can be found in just about any firewall is the rule that anything that is not expressly allowed is by default denied. This is the foundation for any strategy of protecting your network. Right out of the box, once you have your FortiGate device connected into your network and hooked up with your ISP, your network is protected. Nothing is getting out or in so it is not very convenient, but you don't have to worry that between the time you hooked it up and the point that you got all of the policies in place that someone could have gotten in and done something to your resources. The reason that this needs to be kept in mind when designing policies is because you cannot assume that any traffic will be allowed just because it makes sense to do so. If you want any kind of traffic to make it past the FortiGate firewall you need to create a policy that will allow that traffic. To maintain the protection of the network should also make sure that the any policy you create allows only the traffic you intend to go only to where you specifically want it to go and when you want it to go there.

Example

You have a web server on your network that is meant to provide a collaborative work environment web site for your employees and a partner company for a project over the course of the next 3 months.

It is theoretically possible to allow connections into your network to any device on that network for any service and at any time. The problem with this is that we might not want just anybody looking at those resources. Sadly, no matter how much it is wished otherwise, not everybody on the Internet can be trusted. Which means we now have to be very specific in our instructions as to what traffic to allow into the network. Each step that we take towards being more specific as to what we allow means that there is that much more that is not allowed and the level of protection of a resources is directly proportional to the amount of traffic that is not allowed. If somebody can't get at it they can't damage or steal it.

Limiting where the traffic is allowed to go to means that other computers on your network besides the web-server are protected.

- Limiting where the traffic is allowed to come from means that, if feasible, you can limit the systems that can access the web server to just employees or the partner company computers.
- Limiting the services to just web traffic means that a malicious person, even if they were connection from a computer at the partner organization could only use the features of web traffic to do anything malicious.
- Limiting the policy to the time span of the project would mean that even if the IT department forgot to remove the policy after the end of the project than no computer from the other company could be used to do anything malicious through the policy that allowed the traffic.

This is just a very basic example but it shows the underlying principles of how the idea that anything not expressly allowed is by default denied can be used to effectively protect your network.

Policy order

Another important factor in how firewall policies work is the concept of precedence of order or if you prefer a more recognizable term, "first come, first served".

It is highly likely that even after only a relatively small number of policies have been created that there will be some that overlap or are subsets of the parameters that the policies use to determine which policy should be matched against the incoming traffic. When this happens there has to be a method to determine which policy should be applied to the packet. The method which is used by most firewalls is based on the order of the sequence of the policies.

If all of the policies were placed in a sequential list the process to match up the packet would start at the top of the list and work its way down. It would compare information about the packet, specifically these points of information:

1. The interface the packet connected to the FortiGate firewall
2. The source of the packet. This can include variations of the address, user credentials or device
3. The destination of the packet. This can include address or Internet service
4. The interface the packet would need to use to get to the destination address based on the routing table
5. The service or port the packet is destined for
6. The time that the packet connected to the FortiGate

As soon as the a policy is reached that matches all of the applicable parameters, the instructions of that policy are applied and the search for any other matching policies is stopped. All subsequent policies are disregarded. Only 1 policy is applied to the packet.

If there is no matching policy among the policies that have been configured for traffic the packet finally drops down to what is always the last policy. It is an implicit policy. One of a few that are referred to by the term "policy0". This policy denies everything.

The implicit policy is made up of the following settings:

- Incoming Interface: any
- Source Address: all
- Outgoing Interface: any
- Destination Address: all
- Action: DENY

The only setting that is editable in the implicit policy is the logging of violation traffic.

A logical best practice that comes from the knowledge of how this process works is to make sure that the more specific or specialized a policy is, the closer to the beginning of the sequence it should be. The more general a policy is the higher the likelihood that it could include in its range of parameters a more specifically targeted policy. The more specific a policy is, the higher the probability that there is a requirement for treating that traffic in a specific way.

Example

For security reasons there is no FTP traffic allowed out of a specific subnet so there is a policy that states that any traffic coming from that subnet is denied if the service is FTP, so the following policy was created:

Policy #1

Source Interface	Internal1
Source Address	192.168.1.0/24
Source User(s)	<left at default setting>
Source Device Type	<left at default setting>
Outgoing Interface	WAN1
Destination Address	0.0.0.0/0.0.0.0
Service	FTP
Schedule	always
Action	deny

Now as these things usually go it turns out that there has to be an exception to the rule. There is one very secure computer on the subnet that is allowed to use FTP and once the content has been checked it can then be distributed to the other computer on the subnet. So a second firewall policy is created.

Policy #2

Source Interface	Internal1
Source Address	192.168.1.38/32
Source User(s)	<left at default setting>
Source Device Type	<left at default setting>
Outgoing Interface	WAN1
Destination Address	0.0.0.0/0.0.0.0
Service	FTP
Schedule	always
Action	Allow

By default, a policy that has just been created will be placed last in the sequence so that it is less likely to interfere with existing policies before it can be moved to its intended position. If you look at Policy #2 you will notice that it is essentially the same as Policy #1 exempt for the Source Address and the Action. You will also notice that the Source Address of the Policy #2 is a subset of the Source address in policy #1. This means that if nothing further is done, Policy #2 will never see any traffic because the traffic will always be matched by Policy #1 and processed before it has a chance to reach the second policy in the sequence. For both policies to work as intended Policy #2 needs to be moved to before Policy #1 in the sequence.

Policy identification

There are two ways to identify a policy. The most obvious is the policy name and this is easily read by humans, but with a little effort it is possible to have a policy without a name, therefore every policy has an ID number.

When looking at the policy listing it can appear as if the policies are identified by the sequence number in the far left column. The problem is that this number changes as the position of the policy in the sequence changes. The column that correctly identifies the policy, and the value sticks with the policy is the "ID" column. This column is not shown by default in the listing but can be added to the displayed columns by right clicking on the column heading bar and selecting it from the list of possible columns.

When looking in the configuration file the sequence is based upon the order of the policies as they are in the file just as they are in the list in the GUI. However, if you need to edit the policy in the CLI you must use the ID number.

UUID support

Universally Unique Identifier (UUID) attributes have been added to policies to improve functionality when working with FortiManager or FortiAnalyzer units. If required, the UUID can be set manually through the CLI.

CLI Syntax:

```
config firewall {policy/policy6/policy46/policy64}
edit 1
set uuid <example uuid: 8289ef80-f879-51e2-20dd-fa62c5c51f44>
next
end
```

NTurbo support CAPWAP traffic

NTurbo is used for IPSec+IPS case. The IPSec SA info is passed to NTurbo as part of VTAG for control packet and will be used for the xmit.



If the packets need to go through IPSec interface, the traffic will be always offloaded to NTurbo. But for the case that SA has not been installed to NP6 because of hardware limitation or SA offload disable, the packets will be sent out through raw socket by IPS instead of NTurbo, since the software encryption is needed in this case.

CLI :

Previously, NTurbo could only be enabled or disabled globally. The setting of np-acceleration has been added to the firewall policy context instead of just the global context.

CLI command in the firewall policy to enable/disable NTurbo acceleration.

```
config firewall policy
edit 1
set np-acceleration [enable|disable]
end
```

When IPS is enabled for VPN IPsec traffic, the data can be accelerated by NTurbo.

Learning mode for policies

The learning mode feature is a quick and easy method for setting a policy to allow everything but to log it all so that it can later be used to determine what restrictions and protections should be applied. The objective is to monitor the traffic not act upon it while in Learning mode.

Once the **Learn** action is enabled, functions produce hard coded profiles that will be enabled on the policy. The following profiles are set up:

- AntiVirus (av-profile)
- Web Filter (webfilter-profile)
- Anti Spam(spamfilter-profile)
- Data Leak Prevention (dlp-sensor)
- Intrusion Prevention (ips-sensor)
- Application Control (application-list)
- Proxy Options (profile-protocol-options)



- These UTM profiles are all using Flow mode
- SSL inspection is always disable for the Learn option
- These profiles are static and cannot be edited.

Profiles that are not being used are:

- DNS Filter (Does not have a Flow mode)
- Web Application Firewall(Does not have a Flow mode)
- CASI(Almost all signatures in CASI require SSL deep inspection. Without SSL inspection, turning on CASI serves little purpose)

The ability to allow policies to be set to a learning mode is enabled on a per VDOM basis.

```
config system settings
  set gui-policy-learning [enable | disable]
  end
```

Once the feature is enabled on the VDOM, Learn is an available **Action** option when editing a policy.



Because this feature requires a minimum level of logging capabilities, it is only available on FortiGates with hard drives. Smaller models may not be able to use this feature.

New Policy

Name	<input type="text"/>
Incoming Interface	<input type="text"/> +
Outgoing Interface	<input type="text"/> +
Source	<input type="text"/> +
Destination Address	<input type="text"/> +
Schedule	always
Service	<input type="text"/> +
Action	<input checked="" type="checkbox"/> ACCEPT <input type="checkbox"/> DENY LEARN <input type="checkbox"/> IPsec

Firewall / Network Options

NAT

Comments Write a comment... 0/1023

Enable this policy

OK **Cancel**

Once the Learning policy has been running for a sufficient time to collect needed information a report can be looked at by going to **Log & Report > Learning Report**.

The Report can be either a **Full Report** or a **Report Summary**

The time frame of the report can be **5 minutes, 1 hour, or 24 hours**.

The Learning Report includes:

Deployment Methodology

- Test Details
 - Start time
 - End time
 - Model

- Firmware
- Policy List

Executive Summary

- Total Attacks Detected
- Top Application Category
- Top Web Category
- Top Web Domain
- Top Host by Bandwidth
- Host with Highest Session Count

Security and Threat Prevention

- High Risk Applications
- Application Vulnerability Exploits
- Malware, botnets and Spyware/Adware
- At-Risk Devices and Hosts

User Productivity

- Application Usage
 - Top Application Categories
 - Top Social Media Applications
 - Top Video/Audio Streaming Applications
 - Top Peer to Peer Applications
 - Top Gaming Applications
- Web Usage
 - Top Web Categories
 - Top Web Applications
 - Top Web Domains

Policy modes

You can operate your FortiGate or individual VDOMs in **Next Generation Firewall (NGFW) Policy Mode**.

You can enable NGFW policy mode by going to **System > Settings**, setting the **Inspection mode** to **Flow-based** and setting the NGFW mode to **Policy-based**. When selecting **NGFW policy-based** mode you also select the SSL/SSH Inspection mode that is applied to all policies

Flow-based inspection with profile-based **NGFW mode** is the default in FortiOS 5.6.



Or use the following CLI command:

```
config system settings
```

```

set inspection-mode flow
set ngfw-mode {profile-based | policy-based}
end

```

NGFW policy mode and NAT

If your FortiGate is operating in NAT mode, rather than enabling source NAT in individual NGFW policies you go to **Policy & Objects > Central SNAT** and add source NAT policies that apply to all matching traffic. In many cases you may only need one SNAT policy for each interface pair. For example, if you allow users on the internal network (connected to port1) to browse the Internet (connected to port2) you can add a port1 to port2 Central SNAT policy similar to the following:

New Central SNAT Policy

Incoming Interface	port1	<input type="button" value="X"/>
Outgoing Interface	port2	<input type="button" value="X"/>
Source address	all	<input type="button" value="X"/>
Destination address	all	<input type="button" value="X"/>

NAT

IP Pool Configuration	<input checked="" type="button" value="Use Outgoing Interface Address"/> <input type="button" value="Use Dynamic IP Pool"/>
Protocol	<input type="button" value="ANY"/> <input type="button" value="TCP"/> <input type="button" value="UDP"/> <input type="button" value="SCTP"/> <input type="button" value="Specify"/> <input type="text" value="0"/>

Application control in NGFW policy mode

You configure **Application Control** simply by adding individual applications to security policies. You can set the action to accept or deny to allow or block the applications.

Name	Block YouTube
Incoming Interface	port1
Outgoing Interface	port2
Source	all +
Destination	all +
Schedule	always
Service	ALL +
Application	YouTube YouTube_Channel.Access YouTube_HD.Streaming YouTube_Video.Access YouTube_Video.Embedded +
URL Category	+
Action	<input checked="" type="button"/> ACCEPT <input type="button"/> DENY <input type="button"/> LEARN

Web filtering in NGFW mode

You configure **Web Filter** by adding URL categories to security policies. You can set the action to accept or deny to allow or block the applications.

Name	<input type="text" value="Block Streaming Websites"/>
Incoming Interface	<input type="text" value="port1"/>
Outgoing Interface	<input type="text" value="port2"/>
Source	<input type="text" value="all"/> X +
Destination	<input type="text" value="all"/> X +
Schedule	<input type="text" value="always"/>
Service	<input type="text" value="ALL"/> X +
Application	<input type="text" value=""/> +
URL Category	<input type="text" value="Streaming Media and Download"/> X +
Action	✓ ACCEPT ✗ DENY ✖ LEARN

Other NGFW policy mode options

You can also combine both application control and web filtering in the same NGFW policy mode policy. Also if the policy accepts applications or URL categories you can also apply Antivirus, DNS Filtering, and IPS profiles in NGFW mode policies as well a logging and policy learning mode.

Security profiles

Where security policies provide the instructions to the FortiGate unit for controlling what traffic is allowed through the device, the Security profiles provide the screening that filters the content coming and going on the network. Security profiles enable you to instruct the FortiGate unit about what to look for in the traffic that you don't want, or want to monitor, as it passes through the device.

A security profile is a group of options and filters that you can apply to one or more firewall policies. Security profiles can be used by more than one security policy. You can configure sets of security profiles for the traffic types handled by a set of security policies that require identical protection levels and types, rather than repeatedly configuring those same security profile settings for each individual security policy.

For example, while traffic between trusted and untrusted networks might need strict antivirus protection, traffic between trusted internal addresses might need moderate antivirus protection. To provide the different levels of protection, you might configure two separate profiles: one for traffic between trusted networks, and one for traffic between trusted and untrusted networks.

Security profiles are available for various unwanted traffic and network threats. Each are configured separately and can be used in different groupings as needed. You configure security profiles in the Security Profiles menu and applied when creating a security policy by selecting the security profile type.

There is a separate handbook for the topic of the Security Profiles, but because the Security Profiles are applied through the Firewall policies it makes sense to have at least a basic idea of what the security profile do and how they integrate into the FortiGate's firewall policies. The following is a listing and a brief description of what the security profiles offer by way of functionality and how they can be configured into the firewall policies.

- HTTP
- SMTP
- POP3
- IMAP
- FTP
- NNTP
- MAPI
- DNS
- IM

AntiVirus

Antivirus is used as a catch all term to describe the technology for protection against the transmission of malicious computer code sometimes referred to as malware. As anyone who has listened to the media has heard that the Internet can be a dangerous place filled with malware of various flavors. Currently, the malware that is most common in the Internet, in descending order, is Trojan horses, viruses, worms, adware, back door exploits, spyware and other variations. In recent years, not only has the volume of malicious software become greater than would have been believed when it first appeared but the level of sophistication has risen as well.

The Antivirus Filter works by inspecting the traffic that is about to be transmitted through the FortiGate. To increase the efficiency of effort it only inspects the traffic being transmitted via the protocols that it has been configured to check. Before the data moves across the FortiGate firewall from one interface to another it is checked for attributes or signatures that have been known to be associated with malware. If malware is detected, it is removed.

Web Filter

Malicious code is not the only thing to be wary of on the Internet. There is also the actual content. While the content will not damage or steal information from your computer there is still a number of reasons that would require protection from it.

In a setting where there are children or other sensitive people using the access provided by a connected computer there is a need to make sure that images or information that is not appropriate is not inadvertently displayed to them. Even if there is supervision, in the time it takes to recognize something that is inappropriate and then properly react can expose those we wish to protect. It is more efficient to make sure that the content cannot reach the screen in the first place.

In an organizational setting, there is still the expectation that organization will do what it can to prevent inappropriate content from getting onto the computer screens and thus provoking an Human Resources incident. There is also the potential loss of productivity that can take place if people have unfiltered access to the Internet.

Some organizations prefer to limit the amount of distractions available to tempt their workers away from their duties.

The Web filter works primarily by looking at the destination location request for a HTTP(S) request made by the sending computer. If the URL is on a list that you have configured to list unwanted sites, the connection will be disallowed. If the site is part of a category of sites that you have configured to deny connections to the session will also be denied. You can also configure the content filter to check for specific key strings of data on the actual web site and if any of those strings of data appear the connection will not be allowed.

The configuration for each of these protocols is handled separately.

DNS filtering is similar to Web filtering from the viewpoint of the user. The difference is under the hood. When using regular Web filtering, the traffic can go through some processing steps before it gets to the point where the web filter determines whether or not the traffic should be accepted or denied. Because the filtering takes place at the DNS level, some sites can be denied before a lot of the additional processing takes place. This can save resource usage on the FortiGate and help performance.

Application Control

Application Control is designed to allow you to determine what applications are operating on your network and to filter the use of these applications as required. Application control is also for outgoing traffic to prevent the use of applications that are against an organization's policy from crossing the network gateway to other networks. An example of this would be the use of proxy servers to circumvent the restrictions put in place using the Web Filter.

Intrusion Prevention (IPS)

Intrusion Prevention System is almost self explanatory. In the same way that there is malware out on the Internet that the network needs to be protected from there are also people out there that take a more targeted approach to malicious cyber activity. No operating system is perfect and new vulnerabilities are being discovered all of the time. An intrusion prevention system is designed to look for activity or behavior that is consistent with attacks against your network. When attack like behavior is detected it can either be dropped or just monitored depending on the approach that you would like to take.

As new vulnerabilities are discovered they can be added to the IPS database so that the protection is current.

Anti-Spam

Spam or unsolicited bulk email is said to account for approximately 90% of the email traffic on the Internet. Sorting through it is both time consuming and frustrating. By putting an email filter on policies that handle email traffic, the amount of spam that users have to deal with can be greatly reduced.

Data Leak Prevention (DLP)

Data Leak Prevention is used to prevent sensitive information from leaving your network. When people think of security in the cyber-world one of the most common images is that of a hacker penetrating your network and making off with your sensitive information, but the other way that you can lose sensitive data is if someone already on the inside of your network sends it out. This does not have to be an act of industrial espionage. It can just be a case of not knowing the policies of the organization or a lack of knowledge of security or laws concerning privacy.

For instance, a company may have a policy that they will not reveal anyone's Social Security number, but an employee emails a number of documents to another company that included a lengthy document that has a Social Security number buried deep within it. There is not malicious intent but if the information got out there could be repercussions.

If an organization has any information in a digital format that it cannot afford for financial or legal reasons, to leave its network, it makes sense to have Data Leak Prevention in place as an additional layer of protection.

VoIP

Voice over IP is essentially the protocols for transmitting voice or other multimedia communications over Internet Protocol networks such as the Internet. The Security Profiles VoIP options apply the SIP Application Level Gateway (ALG) to support SIP through the FortiGate unit. The SIP ALG can also be used to protect networks from SIP-based attacks.

ICAP

Internet Content Adaptation Protocol (ICAP) off loads HTTP traffic to another location for specialized processing. The purpose of this module when triggered is to send the incoming HTTP traffic over to a remote server to be processed thus taking some of the strain off of the resources of the FortiGate unit. The reasons for the specialized process could be anything from more sophisticated Antivirus to manipulation of the HTTP headers and URLs.

Just like other components of the FortiGate, there is the option for different Proxy Option profiles so that you can be very granular in your control of the workings of the FortiGate. In the case of the Proxy Option profiles the thing that you will want to focus on is the matching up of the correct profile to a firewall policy that is using the appropriate protocols. If you are creating a Proxy Option profile that is designed for policies that control SMTP traffic into your network you only want to configure the settings that apply to SMTP. You do not need or want to configure the HTTP components.

The Web Application Firewall performs a similar role as devices such as Fortinet's FortiWeb, though in a more limited fashion. Its function is to protect internal web servers from malicious activity specific to those types of servers. This includes things like SQL injection, Cross site Scripting and trojans. It uses signatures and other straight forward methods to protect the web servers, but it is a case of turning the feature on or off and the actions are limited to **Allow**, **Monitor** or **Block**. To get protection that is more sophisticated, granular and intelligent, as well as having many more features, it is necessary to get a device like the FortiWeb that can devote more resources to the process. However, if your needs are simple, choosing to use the WAF feature built into the FortiGate should provide valuable protection.

The comfort client feature to mitigates this potential issue by feeding a trickle of data while waiting for the scan to complete so as to let the user know that processing is taking place and that there hasn't been a failure in the transmission. This slow transfer rate continues until the antivirus scan is complete. Once the file has been successfully scanned without any indication of viruses the transfer will proceed at full speed.

Security profile groups

It may seem counter intuitive to have a topic on security profile groups in the Firewall Chapter/Handbook when there is already a chapter/handbook on Security Profiles, but there are reasons.

- Security profile groups are used exclusively in the configuration of a firewall policy, which is described in the Firewall Chapter/Handbook.

- The CLI commands for creating and using security profile groups are in the firewall configuration context of the command line structure of settings.

The purpose of security profile groups is just the same as other groups such as Address, Service, and VIP groups. They are used to save time and effort in the administration of the FortiGate when there are a lot of policies with a similar pattern of Security Profile use. In a fairly basic network setup with a handful of policies it doesn't seem like it would be worth the effort to set up groups of security profiles but if you have a large complex configuration with hundreds of policies where many of them use the same security profiles it can definitely save some effort and help prevent missing adding an important profile from a policy. As an added benefit, when it comes time to add or change the profiles for the policies that use the Security Profile Groups, the changes only have to be made to the group, not each policy.

The most difficult part about using security profile groups is making them visible in the GUI.

Making security profile groups visible in the GUI

By default, the Security Profile Groups are not visible in the GUI. Neither the ability to assign one to a policy nor the ability to configure the members of a group are available by default. You will not find the option to enable Security Profile Groups under **System > Feature Visibility** either. Instead, they only become visible in the GUI once one has been created and assigned to a policy. This must be done the first time through the CLI using the following syntax:

```
config system settings
    set gui-dynamic-profile-display enable
end
```

Step 1 - Create a security profile group:

Enter the command:

```
config firewall profile-group
```

Use the edit command to give a name to and create a new Security Profile Group

```
(profile-group) # edit test-group
```

Configure the members of the group by setting the name of the desired profile in the field for the related profile/sensor/list. The options are:

av-profile	Name of an existing Antivirus profile.
webfilter-profile	Name of an existing Web filter profile.
dnsfilter-profile	Name of an existing DNS filter profile.
spamfilter-profile	Name of an existing Spam filter profile.
dlp-sensor	Name of an existing DLP sensor.
ips-sensor	Name of an existing IPS sensor.
application-list	Name of an existing Application list.
voip-profile	Name of an existing VoIP profile.

icap-profile	Name of an existing ICAP profile.
waf-profile	Name of an existing Web application firewall profile.
profile-protocol-options	Name of an existing Protocol options profile.
ssl-ssh-profile	Name of an existing SSL SSH profile.

Example:

```
set av-profile default
set profile-protocol-options default
end
```

Always set the `profile-protocol-options` setting before attempting to save the profile group. If this is not set, you will get the error:



```
node_check_object fail! for profile-protocol-options
Attribute 'profile-protocol-options' MUST be set.
Command fail. Return code -56
```

Step 2 - Add a security profile to a policy

Now that there is group to add to a policy we can configure a policy to allow the use of a Security Policy group. This is also done in the CLI.

In the following example only the command necessary to enable the use and pick of a Security Policy group have been listed.

```
config firewall policy
edit 0
  set utm-status enable
  set profile-type group
  set profile-group test-group
```

Step 3 - The appearance in the GUI of the security profile group configuration features

- Under **Security Profiles** there is a menu item called **Profile Groups** that can be used to create new and edit existing profile groups.
- In the **Edit Policy** window for **IPv4** and **IPv6** policies there is a **Use Security Profile Group** field to enable or disable the use of the groups.
 - In the window, policy groups can be created or edited by clicking on the appropriate icons next to or in the drop down menu
 - In the policy listing window there is a Security Profiles column.
 - Right or left clicking on the icon for the group brings up editing options either via a slide out window or a drop down menu, respectively.

Proxy option components

Any time a security profile that requires the use of a proxy is enabled the Proxy Options field will be displayed. Certain inspections defined in security profiles require that the traffic be held in proxy while the inspection is carried out and so the Proxy Options are there to define the parameters of how the traffic will be processed and to what level the traffic will be processed. In the same way that there can be multiple security profiles of a single type there can also be a number of unique Proxy Option profiles so that as the requirements for a policy differ from one policy to the next you can also configure a different Proxy Option profile for each individual policy or you can use one profile repeatedly.

The Proxy Options refer to the handling of the following protocols:

- HTTP
- SMTP
- POP3
- IMAP
- FTP
- NNTP
- MAPI
- DNS
- IM

The configuration for each of these protocols is handled separately.

It should also be noted that these configurations apply to only the Security Profiles Proxy-based processes and not the Flow-based processes.

The use of different proxy profiles and profile options

Just like other components of the FortiGate, there is the option for different Proxy Option profiles so that you can be very granular in your control of the workings of the FortiGate. In the case of the Proxy Option profiles the thing that you will want to focus on is the matching up of the correct profile to a firewall policy that is using the appropriate protocols. If you are creating a Proxy Option profile that is designed for policies that control SMTP traffic into your network you only want to configure the settings that apply to SMTP. You do not need or want to configure the HTTP components.

Oversized file log

This setting is for those that would like to log the occurrence of oversized files being processed. It does not change how they are processed it only enables the FortiGate unit to log that they were either blocked or allowed through. A common practice is to allow larger files through without antivirus processing. This allows you to get an idea of how often this happens and decide on whether or not to alter the settings relating to the treatment of oversized files.

The setting of the threshold for what is considered to be an oversized file is located in the Oversized File / Email Threshold that is found in some of the protocol options for the Proxy Options.

Protocol port mapping

While each of the protocols listed has a default TCP port that is commonly used, the level of granularity of control on the FortiGate firewall allows that the port used by the protocols can be individually modified in each separate

Profile. It can also be set to inspect any port with flowing traffic for that particular protocol. The headers of the packets will indicate which protocol generated the packet. To optimize the resources of the unit the mapping and inspection of protocols can be enabled or disabled depending on your requirements.

Comfort clients

When proxy-based antivirus scanning is enabled, the FortiGate unit buffers files as they are downloaded. Once the entire file is captured, the FortiGate unit begins scanning the file. During the buffering and scanning procedure, the user must wait. After the scan is completed, if no infection is found, the file is sent to the next step in the process flow. If the file is a large one this part of the process can take some time. In some cases enough time that some users may get impatient and cancel the download.

The comfort client feature mitigates this potential issue by feeding a trickle of data while waiting for the scan to complete so as to let the user know that processing is taking place and that there hasn't been a failure in the transmission. This slow transfer rate continues until the antivirus scan is complete. Once the file has been successfully scanned without any indication of viruses the transfer will proceed at full speed.

If there is evidence of an infection the FortiGate unit caches the URL and drops the connection. The client does not receive any notification of what happened because the download to the client had already started. Instead, the download stops and the user is left with a partially downloaded file. If the user tries to download the same file again within a short period of time, the cached URL is matched and the download is blocked. The client receives the Infection cache message replacement message as a notification that the download has been blocked. The number of URLs in the cache is limited by the size of the cache.

Client comforting is available for HTTP and FTP traffic. If your FortiGate unit supports SSL content scanning and inspection, you can also configure client comforting for HTTPS and FTPS traffic.



Buffering the entire file allows the FortiGate unit to eliminate the danger of missing an infection due to fragmentation because the file is reassembled before examination. Client comforting can send unscanned and therefore potentially infected content to the client. You should only enable client comforting if you are prepared to accept this risk. Keeping the client comforting interval high and the amount low will reduce the amount of potentially infected data that is downloaded.

Oversized file/email threshold

This is another feature that is related to antivirus scanning. The FortiGate unit has a finite amount of resources that can be used to buffer and scan a file. If a large file such as an ISO image or video file was to be downloaded this could not only overwhelm the memory of the FortiGate, especially if there were other large files being downloaded at the same time, but could exceed it as well. For this reason, how to treat large files needs to be addressed.

A threshold is assigned to determine what should be considered an oversize file or email. This can be set at any size from 1 MB to 50 MB. Any file or email over this threshold will not be processed by the Antivirus Security Profiles. Once a file is determined to be oversized it must be then determined whether to allow it or to block it.

These settings are not a technical decision but a policy one that will depend on your comfort level with letting files into your network. As there often is, there is a compromise between convenience or ease of use and security. If you want to go for a high peace of mind level you can configure the firewall to block oversized files and thus no files would be coming into the network that have not been scanned. If you are looking for optimizing the memory of the FortiGate unit and making sure that everybody is getting the files they want, you can lower the threshold and allow files that are over the threshold.



It should be noted that in terms of probability that malware is more likely to be found in smaller files than in larger files. A number of administrators take this into account when they lower the default threshold so as to lessen the impact on memory if they see the FortiGate unit going into conserve mode on a regular basis.

Chunked bypass

The HTTP section allows the enabling of “Chunked Bypass”. This refers to the mechanism in version 1.1 of HTTP that allows a web server to start sending chunks of dynamically generated output in response to a request before actually knowing the actual size of the content. Where dynamically generated content is concerned this means that there is a faster initial response to HTTP requests. From a security stand point it means that the content will not be held in the proxy as an entire file before proceeding.

Allow fragmented messages

The specifications of RFC 2046 allow for the breaking up of emails and sending the fragments in parallel to be rebuilt and read at the other end by the mail server. It was originally designed to increase the performance over slower connections where larger email messages were involved. It will depend on your mail configuration if this is even possible for your network but outside of Microsoft Outlook and Outlook Express, not many email clients are set up to break up messages like this. The drawback of allowing this feature is that if malware is broken up between multiple fragments of the message the risk is run that it will not be detected by some antivirus configurations because the code may not all be present at the same time to identify.

Append email signature

The Append Email Signature is used when an organization would like to ensure that over and above our in this case underneath the existing personal signatures of the sender, all of the emails going out of their network have the appropriate “boilerplate”, for lack of a better term. These appended emails do not replace existing signatures. They are as the feature states, appended to the email.

Examples could include things like:

- Without prior approval the email should not be forwarded.
- Please be environmentally friendly and don't print out emails
- For questions regarding the purchasing of our products please call...

It can be anything that the organization would like as long as it is in text format. The use of this feature usually works best in an environment where there is some standardization of what goes into the personal signatures of the senders so that there is no duplication or contradiction of information in the signatures.

SSL/SSH inspection

While the profile configuration for **SSL/SSH Inspection** is found in the **Security Profiles** section it is enabled in the firewall policy by enabling any of the security profiles. Choosing which of the **SSL/SSH Inspection** profiles is all that can really be done in the policy.

The reason for having this inspection as part of the policy is the wide spread use of encryption by both legitimate and malicious actors. The legitimate users of the Internet use encryption to hide their information from snooping bad guy but the bad guys use encryption to hide their malicious content from being scanned for viruses and other malicious code by security devices.

By using the correct SSL certificates, the FortiGate can open up encrypted traffic and inspect it for malicious content that would otherwise make it past the other profiles because they couldn't read the encrypted traffic.

There are two basic types of inspection:

- Certificate inspection, which only looks at the certificate that encrypted the packets to make sure that it is a recognized and valid certificate.
- Full inspection, or deep inspection, that looks at all of the content of the packet. While more thorough, it also takes up more resources to perform.

HTTP Strict Transport Security (HSTS) Protocol

HSTS is a protocol used by Google and other web browsers to prevent man-in-the-middle attacks.

When performing deep inspection, the FortiGate intercepts the https traffic and would send its own self-signed CA certificate to the browser. If the browser is configured to use HSTS connections, it would refuse the FortiGate CA certificate since it is not on the trusted list for Google servers.

To keep the CA certificate from being refused, the HSTS settings should be cleared from the browser. Instructions for this vary between browsers.

SSH MITM deep inspection

Due to an increase, in recent years of vulnerabilities discovered in the SSH protocol, protections have been incorporated into FortiOS's Intrusion Prevention System (IPS) engine that will aid in protecting against malicious activity coming through the FortiGate against SSH access points. In addition to the protections offered by IPS, additional settings and functionality have been added to protect against man-in-the-middle (MITM) attacks that use SSH as the attack vector.

These protections include support for comprehensive security controls on SSH MITM deep inspections:

- SSH filter profile to control SSH tunnel types and filtering on SSH shell commands.
- SSH proxy policy that can apply a proxy firewall policy or firewall policy using SSH inspection, with user authentication to SSH sessions.
- Support for SSH tunnel policy access control for TCP/IP port forwarding traffic tunneled through SSH proxy. This allows IPS scanning to be applied to the tunneled traffic.
- Support the use of SSH trust status to detect and prevent SSH-based MITM attacks.

Support SSH proxy policy for SSH sessions

Policy check

To enable SSH proxy-policy or SSH tunnel-policy, `ssh-policy-check` or `ssh-tun-policy-check` must be enabled under `ssl-ssh-profile` for the corresponding firewall policy.

```
config firewall ssl-ssh-profile
  edit <name>
    config ssh
      set ssh-policy-check [disable|enable]
      set ssh-tun-policy-check [disable|enable]
      end
    end
```

SSH proxy option

Set the proxy type to ssh in config firewall proxy-policy.

```
config firewall proxy-policy
  edit <pol-id>
    set proxy ssh
    set action {accept|deny}
    set utm-status enable
    set ssh-filter-profile <profile_name>
  end
```

Authentication for SSH

When user or user-group is set in SSH proxy policy, firewall authentication can be implemented for SSH proxy traffic.

```
config authentication rule
  edit <name>
    set protocol ssh
  end
```

Basic authentication scheme:

```
config authentication scheme
  edit "ssh-active"
    set method basic
    set user-database "local" #or LDAP server
  end
```



Under authentication for SSH, with basic authentication scheme, the client is expected to input username/password in the format of:

- <firewall username>:<server username>
- <firewall password>:<server password>

SSH-publickey authentication scheme:

```
config authentication scheme
  edit "ssh-pkey"
    set method ssh-publickey
    set user-database "local"
    set ssh-ca "server-ca"
  end
```



The user name is embedded in ssh-publickey. The user group information will be retrieved if the public key is validated by CA.

FortiOS currently only supports validation by using CA. The CA needs to be configured under config firewall ssh local-ca.

Private-key is not required for client public-key authentication. Once the client offers the public-key signed by the set CA will be authenticated.

Both basic and SSH-publickey authentication scheme:

```
config authentication scheme
  edit "ssh-pkey"
    set method basic ssh-publickey
    set user-database "local"
    set ssh-ca "server-ca"
  end
```

Support SSH tunnel policy to do access control for TCP/IP port forwarding traffic.

Add a proxy type ssh-tunnel into config firewall proxy-policy

```
config firewall proxy-policy
  edit <pol-id>
    set proxy ssh-tunnel
    set action {accept | deny}
    set utm-status enable
```

The feature supports allowing or denying based on the IPS sensor/app-control applied to the traffic.

```
set ips-sensor <sensor_profile_name>
set application-list <application_control_list>
end
```

Support SSH trust to detect and prevent from SSH MITM attacks

Define trusted SSH host key for specific SSH server

```
config firewall ssh host-key
  edit <name>
    set status {trusted|revoked}
    set type {RSA|DSS|ECDSA}
    set nid <NID of ECDSA key>
    set ip <ip>
    set port <port>
    set hostname <name>
    set public-key <host key>
  end
```

Define trusted/untrusted CAs for host key signing.

Any host key signed by trust CA is trusted unless the host key is revoked.

```
config firewall ssh local-ca
  edit <name>
    set password <passwd>
    set public-key <public key>
    set private-key <private key>
    set source {built-in|user}
  end
```



- The system creates two built-in SSH CAs:
 - Fortinet_SSH_CA
 - Fortinet_SSH_CA_Untrusted.
- The CAs are used to re-sign a server host key with local host-key using trusted/untrusted CA when the server host key is trusted or untrusted.

Define local hostkey templates for trusted re-signing.

By default, the local hostkey templates are generated automatically.

```
config firewall ssh local-key
  edit <name>
    set status [trusted|revoked]
    set type [RSA|DSA|ECDSA|ED25519|RSA-CA|DSA-CA|ECDSA-CA|ED25519-CA]
    set nid <NID of ECDSA key>
    set ip <ip>
    set port <port>
    set hostname <name>
    set password <passwd>
    set public-key <public key>
    set private-key <private key>
    set source {build-in|user}
  end
```



Per VDOM SSH settings

```
config firewall ssh setting
  set caname <trusted-ca>
  set untrusted-caname <untrusted-ca>
  set hostkey-rsa <hostkey-rsa>
  set hostkey-dss <hostkey-dss>
  set hostkey-ecdsa256 <hostkey-ecdsa256>
  set hostkey-ecdsa384 <hostkey-ecdsa384>
  set ed25519-key <ed25519-key>
  set host-trusted-check {enable|disable}
  end
```



- When a host key is trusted and signed by a CA, SSH proxy re-signs according to the type of hostkey using trusted CA.
- When a host is trusted but not signed, SSH proxy sends back according type of hostkey.
- When a host key is untrusted and signed by a CA, SSH proxy re-signs a temporary host key (1 hour) using untrusted CA.
- When a host is trusted but not signed, SSH proxy sends back a temporary host key (1 hour).

Add SSH filter profile table

Support options to block/log "x11-filter/ssh-shell/exec/port-forward/sftp"

```
config ssh-filter profile
  edit <name>
    set block {x11|shell|exec|port-forward|tun-forward|sftp|unknown}+
    set log {x11|shell|exec|port-forward|tun-forward|sftp|unknown}+
  end
```

Add shell command filters

```
config ssh-filter profile
  edit <name>
    config shell-commands
      edit <id>
        set type {simple|regex}
        set pattern <cmd-string>
        set action {block|allow}
        set log {block|allow}
        set alert {block|allow}
        set severity {low|medium|high|critical}
      end
    set default-command-log {block|allow}
  end
```

Mirroring SSL inspected traffic

It is possible to "mirror" or send a copy of traffic decrypted by SSL inspection to one or more FortiGate interfaces so that the traffic can be collected by a raw packet capture tool for archiving or analysis. The mirroring occurs after being processed by the SSL decoder and in the same point in the work flow as the decryption of application data. The decrypted application data is wrapped inside a TCP packet (with IP and Ethernet headers), and then sent to the mirror port.

This feature works when the inspection mode is set to flow-based or proxy-based, but not for explicit proxy.



Decryption, storage, inspection, and use decrypted content is subject to local privacy rules. Use of these features could enable malicious users with administrative access to your FortiGate to harvest sensitive information submitted using an encrypted channel.

In this example, the setting enables the policy to send all traffic decrypted by the policy to the FortiGate port1 and port2 interfaces.

```
config firewall policy
  edit 0
    set ssl-mirror enable
    set ssl-mirror-intf port1 port2
  end
```

Encryption strength for proxied SSH sessions

The level of SSH encryption can be set for SSH sessions on a per-profile basis.

Encryption Level	Description
compatible	This level allows for a broader set of encryption algorithms to be used and is better for compatibility.
high-encryption	This level will only allow AES-CTR, AES-GCM and high encryption algorithms to be used for the session.

Syntax:

```
config firewall ssl-ssh-profile
  edit <profile name>
    config ssh
      set ssh-algorithm {compatible|high-encryption}
    end
  end
```

RPC over HTTP

How protocol options profiles and SSL inspection profiles handle RPC (Remote Procedure Calls) over HTTP traffic can be configured separately from normal HTTP traffic. The configuration is done in the CLI.

Configuration in protocol options

```
config firewall profile-protocol-options
  edit 0
    set rpc-over-http [disable|enable]
  end
```

Configuration in SSL/SSH inspection

```
config firewall ssl-ssh-profile
  edit deep inspection
    set rpc-over-http [disable|enable]
  end
```

NAT

NAT or Network Address Translation is the process that enables a single device such as a router or firewall to act as an agent between the Internet or Public Network and a local or private network. This “agent”, in real time, translates the source IP address of a device on one network interface, usually the Internal, to a different IP address as it leaves another interface, usually the interface connected to the ISP and the Internet. This enables a single public address to represent a significantly larger number of private addresses.

The origins of NAT

In order to understand NAT it helps to know why it was created. At one time, every computer that was part of a network had to have its own addresses so that the other computers could talk to it. There were a few protocols in use at the time, some of which were only for use on a single network, but of those that were routable, the one that had become the standard for the Internet was IP (Internet Protocol) version 4.

When IP version 4 addressing was created nobody had any idea how many addresses would be needed. The total address range was based on the concept of 2 to the 32nd power, which works out to be 4 294 967 296 potential addresses. Once you eliminate some of those for reserved addresses, broadcast addresses, network addresses, multicasting, etc., you end up with a workable scope of about 3.2 million addressees. This was thought to be more than enough at the time. The designers were not expecting the explosion of personal computing, the World Wide Web or smart phones. As of the beginning of 2012, some estimate the number of computers in the world in the neighborhood of 1 billion, and most of those computer users are going to want to be on the Internet or Search the World Wide Web. In short, we ran out of addresses.

This problem of an address shortage was realized before we actually ran out, and in the mid 1990s 2 technical papers called RFCs numbered 1631 (<http://www.ietf.org/rfc/rfc1631.txt>) and 1918 (<http://tools.ietf.org/html/rfc1918>), proposed components of a method that would be used as a solution until a new addressing methodology could be implemented across the Internet infrastructure. For more information on this you can look up IP version 6.

RFC 1631 described a process that would allow networking devices to translate a single public address to multiple private IP addresses and RFC 1918 laid out the use of the private addresses. The addresses that were on the Internet (Public IP addresses) could not be duplicated for them to work as unique addresses, but behind a firewall, which most large institutions had, they could use their own Private IP addresses for internal use and the internal computers could share the external or Public IP address.

To give an idea on a small scale how this works, image that a company has a need for 200 computer addresses. Before Private IP addresses and NAT the company would have purchased a full Class C address range which would have been 254 usable IP addresses; wasting about 50 addresses. Now with NAT, that company only needs 1 IP address for its 200 computers and this leaves the rest of the IP addresses in that range available for other companies to do the same thing.

NAT gives better value than it would first appear because it is not 253 companies that can use 254 addresses but each of those 254 companies could set up their networking infrastructures to use up to thousands of Private IP addresses, more if they don't all have to talk to the Internet at the same time. This process enabled the Internet to keep growing even though we technically have many more computers networked than we have addresses.

Dynamic NAT

Dynamic NAT maps the private IP addresses to the first available Public Address from a pool of possible Addresses. In the FortiGate firewall this can be done by using IP Pools.

Overloading

This is a form of Dynamic NAT that maps multiple private IP address to a single Public IP address but differentiates them by using a different port assignment. This is probably the most widely used version of NAT. This is also referred to as PAT (Port Address Translation) or Masquerading.

An example would be if you had a single IP address assigned to you by your ISP but had 50 or 60 computers on your local network.

Say the internal address of the interface connected to the ISP was 256.16.32.65 (again an impossible address) with 256.16.32.64 being the remote gateway. If you are using this form of NAT any time one of your computers accesses the Internet it will be seen from the Internet as 256.16.32.65. If you wish to test this go to 2 different computers and verify that they each have a different private IP address then go to a site that tells you your IP address such as www.ipchicken.com. You will see that the site gives the same result of 256.16.32.65, if it existed, as the public address for both computers.

As mentioned before this is sometimes called Port Address Translation because network device uses TCP ports to determine which internal IP address is associated with each session through the network device. For example, if you have a network with internal addresses ranging from 192.168.1.1 to 192.168.1.255 and you have 5 computers all trying to connect to a web site which is normally listening on port 80 all of them will appear to the remote web site to have the IP address of 256.16.32.65 but they will each have a different sending TCP port, with the port numbers being somewhere between 1 and 65 535, although the port numbers between 1 to 1024 are usually reserved or already in use. So it could be something like the following:

192.168.1.10	256.16.32.65:	port 486
192.168.1.23	256.16.32.65:	port 2409
192.168.1.56	256.16.32.65:	port 53763
192.168.1.109	256.16.32.65:	port 5548
192.168.1.201	256.16.32.65:	port 4396

And the remote web server would send the responding traffic back based on those port numbers so the network device would be able to sort through the incoming traffic and pass it on to the correct computer.

Overlapping

Because everybody is using the relative same small selection of Private IP addresses it is inevitable that there will be two networks that share the same network range that will need to talk with each other. This happens most often over Virtual Private Networks or when one organization ends up merging with another. This is a case where a private IP address may be translated into a different private IP address so there are no issues with conflict of addresses or confusion in terms of routing.

An example of this would be when you have a Main office that is using an IP range of 172.16.0.1 to 172.20.255.255 connecting through a VPN to a recently acquired branch office that is already running with an IP range of 172.17.1.1 to 172.17.255.255. Both of these ranges are perfectly valid but because the Branch office range is included in the Main Office range any time the system from the Main office try to connect to an address in the Branch Office the routing the system will not send the packet to the default gateway because according to the routing table the address is in its own subnet.

The plan here would be to NAT in both directions so that traffic from neither side of the firewall would be in conflict and they would be able to route the traffic. Everything coming from the Branch Office could be assigned an address in the 192.168.1.1 to 192.168.1.255 range and everything from the Main office going to the Branch Office could be assigned to an address in the 192.168.10.1 to 192.168.10.255 range.

Static NAT

In Static NAT one internal IP address is always mapped to the same public IP address.

In FortiGate firewall configurations this is most commonly done with the use of Virtual IP addressing.

An example would be if you had a small range of IP addresses assigned to you by your ISP and you wished to use one of those IP address exclusively for a particular server such as an email server.

Say the internal address of the Email server was 192.168.12.25 and the Public IP address from your assigned addresses range from 256.16.32.65 to 256.16.32.127. Many readers will notice that because one of the numbers

is above 255 that this is not a real Public IP address. The Address that you have assigned to the interface connected to your ISP is 256.16.32.66, with 256.16.32.65 being the remote gateway. You wish to use the address of 256.16.32.70 exclusively for your email server.

When using a Virtual IP address you set the external IP address of 256.16.32.70 to map to 192.168.12.25. This means that any traffic being sent to the public address of 256.16.32.70 will be directed to the internal computer at the address of 192.168.12.25

When using a Virtual IP address, this will have the added function that when ever traffic goes from 192.168.12.25 to the Internet it will appear to the recipient of that traffic at the other end as coming from 256.16.32.70.

You should note that if you use Virtual IP addressing with the Port Forwarding enabled you do not get this reciprocal effect and must use IP pools to make sure that the outbound traffic uses the specified IP address.

Benefits of NAT

More IP addresses available while conserving public IP addresses

As explained earlier, this was the original intent of the technology and does not need to be gone into further.

Financial savings

Because an organization does not have to purchase IP addresses for every computer in use there is a significant cost savings due to using the process of Network Address Translation.

Security enhancements

One of the side benefits of the process of NAT is an improvement in security. Individual computers are harder to target from the outside and if port forwarding is being used computers on the inside of a firewall are less likely to have unmonitored open ports accessible from the Internet.

Ease of compartmentalization of your network

With a large available pool of IP addresses to use internally a network administrator can arrange things to be compartmentalized in a rational and easily remembered fashion and networks can be broken apart easily to isolate for reasons of network performance and security.

Example

You have a large organization that for security reasons has certain departments that do not share network resources.

You can have the main section of the organization set up as follows;

Network Devices	192.168.1.1 to 192.168.1.25
Internal Servers	192.168.1.26 to 192.168.1.50
Printers	192.168.1.51 to 192.168.1.75

Administration Personnel	192.168.1.76 to 192.168.1.100
Sales People	192.168.1.101 to 192.168.1.200
Marketing	192.168.1.201 to 192.168.1.250

You could then have the following groups broken off into separate subnets:

Accounting	192.168.100.1 to 192.168.100.255
Research and Development	172.16.1.1 to 172.16.255.255
Executive Management	192.168.50.1 to 192.168.50.255
Web sites and Email Servers	10.0.50.1 to 10.0.50.255

These addresses do not have to be assigned right away but can be used as planned ranges.

NAT in transparent mode

Similar to operating in NAT mode, when operating a FortiGate unit in transparent mode you can add security policies and:

- Enable NAT to translate the source addresses of packets as they pass through the FortiGate unit.
- Add virtual IPs to translate destination addresses of packets as they pass through the FortiGate unit.
- Add IP pools as required for source address translation

A FortiGate unit operating in transparent mode normally has only one IP address - the management IP. To support NAT in transparent mode, you can add a second management IP. These two management IPs must be on different subnets. When you add two management IP addresses, all FortiGate unit network interfaces will respond to connections to both of these IP addresses.

Use the following steps to configure NAT in transparent mode:

1. Add two management IPs
2. Add an IP pool to the WAN1 interface
3. Add an Internal to WAN1 security policy

You can add the security policy from the web-based manager and then use the CLI to enable NAT and add the IP pool.

The usual practice of NATing in transparent mode makes use of two management IP addresses that are on different subnets, but this is not an essential requirement in every case.

If there is a router between the client systems and the FortiGate unit you can use the router's capabilities of tracking sessions to assign NATed addresses from an IP pool to the clients even if the assigned address don't belong to a subnet on your network.

Example

Client computer has an IP address of 1.1.1.33 on the subnet 1.1.1.0/24.

Router "A" sits between the client computer and the FortiGate (in transparent mode) with the IP address of 1.1.1.1 on the client's side of the router and the IP address of 192.168.1.211 on the FortiGate's side of the router.

Use NAT to assign addresses from an address pool of 9.9.9.1 to 9.9.9.99 to traffic coming from gateway of 192.168.1.211.

To enable the return traffic to get to the original computer, set up a static route than assigns any traffic with a destination of 9.9.9.0/24 to go through the 192.168.1.211 gateway. As long as the session for the outgoing traffic has been maintained, communication between the client computer and the external system on the other side of the FortiGate will work.

Central NAT table

The central NAT table enables you to define, and control with more granularity, the address translation performed by the FortiGate unit. With the NAT table, you can define the rules which dictate the source address or address group and which IP pool the destination address uses.

While similar in functionality to IP pools, where a single address is translated to an alternate address from a range of IP addresses, with IP pools there is no control over the translated port. When using the IP pool for source NAT, you can define a fixed port to guarantee the source port number is unchanged. If no fix port is defined, the port translation is randomly chosen by the FortiGate unit. With the central NAT table, you have full control over both the IP address and port translation.

The FortiGate unit reads the NAT rules in a top-down methodology, until it hits a matching rule for the incoming address. This enables you to create multiple NAT policies that dictate which IP pool is used based on the source address. The NAT policies can be rearranged within the policy list as well. NAT policies are applied to network traffic after a security policy.

NAT64 and NAT46

NAT64 and NAT46 are the terms used to refer to the mechanism that allows IPv6 addressed hosts to communicate with IPv4 addressed hosts and vice versa. Without such a mechanism an IPv6 node on a network such as a corporate LAN would not be able to communicate with a web site that was still in a IPv4 only environment and IPv4 environments would not be able to connect to IPv6 networks.

One of these setups involves having at least 2 interfaces, 1 on an IPv4 network and 1 on an IPv6 network. The NAT64 server synthesizes AAAA records, used by IPv6 from A records used by IPv4. This way client-server and peer to peer communications will be able to work between an IPv6 only client and an IPv4 server without making changes to either of the end nodes in the communication transaction. The IPv6 network attached to the FortiGate unit should be a 32 bit segment, (for instance 64:ff9b::/96, see RFC 6052 and RFC 6146). IPv4 address will be embedded into the communications from the IPv6 client.

Because the IPv6 range of addresses is so much larger than the IPv4 range, a one to one mapping is not feasible. Therefore the NAT64 function is required to maintain any IPv6 to IPv4 mappings that it synthesizes. This can be done either statically by the administrator or automatically by the service as the packets from the IPv6 network go through the device. The first method would be a stateless translation and the second would be a stateful translation. NAT64 is designed for communication initiated from IPv6 hosts to IPv4 addresses. It is address

mapping like this that allows the reverse to occur between established connections. The stateless or manual method is an appropriate solution when the NAT64 translation is taking place in front of legacy IPv4 servers to allow those specific servers to be accessed by remote IPv6-only clients. The stateful or automatic solution is best used closer to the client side when you have to allow some specific IPv6 clients to talk to any of the IPv4-only servers on the Internet.

There are currently issues with NAT64 not being able to make everything accessible. Examples would be SIP, Skype, MSN, Goggle talk, and sites with IPv4 literals. IPv4 literals being IPv4 addresses that are imbedded into content rather than a FQDN.

Policies that employ NAT64 or NAT46 can be configured from the web-based manager as long as the feature is enabled using the Features setting found at **System > Config > Features**.

- To create a NAT64 policy go to **Policy & Objects > NAT64 Policy** and select **Create New**.
- To create a NAT46 policy go to **Policy > NAT46 Policy** and select **Create New**.

The difference between these NAT policies and regular policies is that there is no option to use the security profiles and sensors.



NAT64 CLAT traffic is now supported by the FortiGate. CLAT traffic comes from devices that use the SIIT translator that plays a part in affecting IPv6 - IPv4 NAT translation.

NAT64 CLAT

NAT64 CLAT traffic is supported by FortiOS. CLAT traffic comes from devices that use the SIIT translator that plays a part in affecting IPv6 - IPv4 NAT translation.

NAT66

NAT66 is Network Address Translation between 2 IPv6 network. The basic idea behind NAT66 is no different than the regular NAT between IPv4 networks that we are all used to. The difference are in the mechanics of how it is performed, mainly because of the complexity and size of the addresses that are being dealt with.

In an IPv4 world, the reason for the use of NAT was usually one or a combination of the following 3 reasons:

- Improved security - actual addresses behind NAT are virtually hidden
- Amplification of addresses - hundreds of computers can use as little as a single public IP address
- Internal address stability - there is control of internal addressing. The addresses can stay the same even if Internet Service Providers change.

In these days of security awareness the protective properties of NAT are not something that are not normally depended on by themselves to defend a network and with the vastly enlarged IPv6 address scope there is no longer a need to amplify the available addresses. However, the desire to have internal address control still exists. The most common reason for using NAT66 is likely to be the maintaining of the existing address scheme of the internal network despite changes outside of it. Imagine that you have an internal network of 2000 IP addresses and one day the company changes its ISP and thus the addresses assigned to it. Even if most of the addressing is handled by DHCP, changing the address scheme is going to have an impact on operations.

Addressing stability can be achieved by:

- Keeping the same provider - this would depend on the reason for the change. If the cost of this provider has become too expensive this is unlikely. If the ISP is out of business it becomes impossible.

- Transfer the addresses from the old provider to the new one - There is little motivation for an ISP to do you a favor for not doing business with them.
- Get your own autonomous system number - this can be too expensive for smaller organizations.
- NAT - this is the only one on the list that is in the control of IT.

There are differences between NAT66 and IPv4 NAT. Because there is no shortage of addresses most organizations will be given a /48 network that can be translated into another /48 network. This allows for a one to one translation, no need for port forwarding. This is a good thing because port forwarding is more complicated in IPv6. In fact, NAT66 will actually just be the rewriting of the prefix on the address.

Example

If your current IPv6 address is

2001:db8:cafe::/48

you could change it to

2001:db8:fea7::/48

There is an exception to the one to one translation. NAT66 cannot translate internal networks that contain 0xffff in bits 49 through 63 - this is due to the way checksums are calculated in TCP/IP: they use the one's-complement representation of numbers which assigns the value zero to both 0x0000 and 0xffff.

How FortiOS differentiates sessions when NATing

The basics of NAT are fairly simple. Many private addresses get translated into a smaller number of public addresses, often just one. The trick is how the FortiGate keeps track of the return traffic because the web server, or what ever device that was out on the Internet is going to be sending traffic back not to the private address behind the FortiGate but to the IP address of the interface on the public side of the FortiGate.

The way this is done is by making each session unique. Most of the attributes that are available in the network packets cannot be changed without changing where the packet will go but because the source port has to be changed anyway in case two computer on the network used the same source port this is a useful way of making each listing of network attributes a unique combination. As a packet goes through the NAT process FortiOS assigns different source ports for each of the internally initiated sessions and keeping track of which port was used for each device in a database until the session has ended. It then becomes a matter of how the port number is selected.

In a very simple example of an environment using NAT, we will use a fictitious university with a rather large student population. So large in fact that they use a subnet of 10.0.0.0/8 as their subnet for workstation IP addresses. All of these private IP addresses are NATed out a single IP address. To keep the number of numeric values in this example from getting to a confusing level, we'll just use "u.u.u.u" to refer to the public IP address of the University and the IP address of the web server on the Internet will be "w.w.w.1".

Student A (IP address 10.1.1.56) sends an HTML request to a web server on the Internet with the IP address w.w.w.1. The applicable networking information in the packet breaks down as follows:

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.1.1.56	u.u.u.u
Destination IP address or dst-ip:	w.w.w.1	w.w.w.1

Attribute	Original Packet	Packet after NATing
Source port or src-port:	10000	46372
Destination port or dst-port	80	80

The source IP address is now that of the public facing interface of the FortiGate and source port number is an unused TCP port number on the FortiGate chosen by the FortiGate. Of these variable the only one the that FortiGate can really change and still have the packet reach the correct destination, in both directions, is the source port number.

There are a few methods of assigning the port number. First we'll look at the methods that are or have been used in the industry but aren't used by Fortinet.

Global pool

This method of differentiation focuses on the attribute of the source port number. In this approach a single pool of potential port numbers is set aside for the purposes of NAT. As a pool number is assigned, it is removed from the pool so that two sessions from different computers can not use the same port number. Once the session is over and no longer in use by the computer, the port number is put back into the pool where it can be assigned again.

Example global pool:

	Hexidecimal	Decimal
Start or range	0x7000	28672
End end of range	0xF000	61440
Possible ports in range	215	32768

This is a simple approach to implement and is good if the number of connections is unlikely to reach the pool size. It would be okay for home use, but our example is for a university using 10.1.1.0/8 as a subnet. That means 16,777,214 possible IP addresses; more than this method can handle.

Fortinet does not use this method.

Global per protocol

This method uses the attributes source port number and type of protocol to differentiate between sessions. This approach is a variation of the first one. An additional piece of information is referred to in the packet that describes the protocol. For instance UDP or TCP. This could effectively double the number of potential addresses to NAT.

Example:

Here are two possible packets that would be considered different by the FortiGate so that any responses from the web server would make it back to their correct original sender.

From Student A

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.1.1.56	u.u.u.1
Destination IP address or dst-ip:	w.w.w.1	w.w.w.1
Protocol	tcp	tcp
Source port or src-port:	10000	46372
Destination port or dst-port	80	80

From Student B

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.5.1.233	u.u.u.1
Destination IP address or dst-ip:	w.w.w.1	w.w.w.1
Protocol	udp	udp
Source port or src-port:	26785	46372
Destination port or dst-port	80	80

Even though the source port is the same, because the protocol is different they are considered to be from different sessions and different computers.

The drawback is that it would depend on the protocols being used be evenly distributed between TCP and UDP. Even if this was the case the number would only double; reaching an upper limit of 65,536 possible connections. That number is still far short of the possible more than 16 million for an IP subnet with an eight bit subnet mask like the one in our example.

Fortinet does not use this method.

Per NAT IP pool

This approach adds on to the previous one by adding another variable. In this case that variable is the IP addresses on the public side of the FortiGate. By having a pool of IP addresses to assign as the source IP address when NATing, the same number that was potentially available for the Global per protocol method can be multiplied by the number of external IP addresses in the pool. If you can assign a second IP address to the pool, you can double the potential number of sessions.

Example:

In this example it will be assumed that the FortiGate has 2 IP addresses that it can use. This could happen either by using two ISPs, or by having a pool of IP addresses assigned to a single interface. For simplicity will will refer

to these IP public IP addresses as `u.u.u.1` and `u.u.u.2`.

Here are two possible packets that would be considered different by the FortiGate so that any responses from the web server would make it back to their correct original sender.

From Student A

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.1.1.56	<code>u.u.u.1</code>
Destination IP address or dst-ip:	<code>w.w.w.1</code>	<code>w.w.w.1</code>
Protocol	tcp	tcp
Source port or src-port:	10000	46372
Destination port or dst-port	80	80

From Student B

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.5.1.233	<code>u.u.u.2</code>
Destination IP address or dst-ip:	<code>w.w.w.1</code>	<code>w.w.w.1</code>
Protocol	tcp	tcp
Source port or src-port:	26785	46372
Destination port or dst-port	80	80

In this example we even made the protocol the same. After the NATing process all of the variables are the same except the source addresss. This is still going to make it back to the original sender.

The drawback is that if you have only one IP address for the purposes of NATing this method does not gain you anything over the last method. Or if you do have multiple IP addresses to use it will still take quite a few to reach the 16 million possible that the subnet is capable of handling.

Fortinet does not use this method.

Per NAT IP, destination IP, port, and protocol

This is the approach that FortiOS uses.

It uses all of the differentiation point of the previous methods, NAT IP, port number and protocol, but the additonal information point of the destination IP is also used. So now the network information points in the packet that the FortiGate keeps in its database to differentiate between sessions is:

- Public IP address of the FortiGate assigned by NATing
- Protocol of the traffic
- Source port assigned by the FortiGate
- Destination IP address of the packet

The last one is an especially good way to differentiate because as a theoretical number, the upper limit on that is the number of Public IP addresses on the whole of the Internet. Chances are that while a large number of session from inside the University will be going to a small group of sites such as Google, YouTube, Facebook and some others it is unlikely that they will all be going to them at the same time.

Example:

In this example it will be assumed that the FortiGate has only one IP address. Two possible packets will be described. The only difference in the attributes recorded will be the destination of the HTML request. These packets are still considered to be from different sessions and any responses will make it back to the correct computer.

From Student A

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.1.1.56	u.u.u.1
Destination IP address or dst-ip:	w.w.w.1	w.w.w.1
Protocol	tcp	tcp
Source port or src-port:	10000	46372
Destination port or dst-port	80	80

From Student B

Attribute	Original Packet	Packet after NATing
Source IP address or src-ip	10.5.1.233	u.u.u.1
Destination IP address or dst-ip:	w.w.w.2	w.w.w.2
Protocol	tcp	tcp
Source port or src-port:	26785	46372
Destination port or dst-port	80	80

The reason that these attributes are used to determine differentiation between traffic is based on how the indexes for the sessions are recorded in the database. When a TCP connection is made through a FortiGate unit, a session is created and two indexes are created for the session. The FortiGate unit uses these indexes to guide matching traffic to the session.

This following could be the session record for the TCP connection in the first example.

Attribute	Outgoing Traffic	Returning Traffic
Source IP address	10.78.33.97 (internal address)	w.w.w.1
Destination address	w.w.w.1	u.u.u.1
Protocol	tcp	tcp
Source port	10000 (from original computer) 46372 (assigned by NAT)	80
Destination port	80	46372 (FortiGate assigned port)

Using the FortiGate's approach for session differentiation, FortiOS only has to ensure that the assigned port, along with the other four attributes is a unique combination to identify the session. So for example, if Student A simultaneously makes a HTTP(port 80) connection and a HTTPS(port 443) connection the same web server this would create another session and the index in the reply direction would be:

Attribute	Outgoing Traffic	Returning Traffic
Source IP address	10.78.33.97 (internal address)	w.w.w.1
Destination address	w.w.w.1	u.u.u.1
Protocol	tcp	tcp
Source port	10000 (from original computer) 46372 (assigned by NAT)	443
Destination port	443	46372 (FortiGate assigned port)

These two sessions are different and acceptable because of the different source port numbers on the returning traffic or the destination port depending on the direction of the traffic.

Calculations for possible session numbers

The result of using these four attributes instead of just the one that was originally used is a large increase in the number of possible unique combinations. For those who love math, the maximum number of simultaneous connections that can be supported is:

$$N \times R \times P \times D \times D_p$$

where:

- **N** is the number of NAT IP addresses
- **R** is the port range,

- **P** is the number of protocols,
- **D** is the number of unique destination IP addresses
- **Dp** the number of unique destination ports.

As a rough example let's do some basic calculations

- **N** - In our existing example we have already stated that there is only one public IP address that is being used by NAT. Realistically, for a university this number would likely be larger, but we're keeping it simple.

N = 1

R - The port range for our example has already been described and we will keep it the same.

R = 32768

P - While there are a few protocols that are involved in Internet traffic we will limit this calculation just to TCP traffic.

P = 1

D - As mentioned before the number of unique destination addresses is growing larger every day, so figuring out the upper limit of that number would be difficult to say the least. Instead we will make the assumption that most of the university students, due to their shared interest and similar demographic will concentrate most of their web browsing to the same sites; sites such as YouTube, Facebook, Google, Twitter, Instagram, Wikipedia etc. This is not even taking into account the fact that many of these popular sites use load balancing and multiple IP addresses. As an arbitrary number let's use the number 25.

D = 25

Dp - To keep things simple it is tempting to limit the destination port to port 80, the one that many associate with web browsing, but this would not be realistic. The use of HTTPS, port 443 is on the rise. There is also email, DNS, FTP, NTP and a number of other background services that we use without thinking too closely about. Let's keep it small and say ten of them.

Dp = 10

The math on this very conservative calculation is:

$$1 \times 32768 \times 1 \times 25 \times 10 = 8,192,000 \text{ possible NAT sessions}$$

When you take into account that the chances of everybody being online at the same time, going only to one of those 25 sites and not millions of others, and using only TCP not UDP or any of the other protocols, it starts to look like this method may provide enough potential unique sessions even for a subnet as large as the one described.

IP pools

IP Pools are a mechanism that allow sessions leaving the FortiGate Firewall to use NAT. An IP pool defines a single IP address or a range of IP addresses to be used as the source address for the duration of the session. These assigned addresses will be used instead of the IP address assigned to that FortiGate interface.



When using IP pools for NATing, there is a limitation that must be taken into account. In order for communication to be successful in both directions, it is normal for the source address in the packet header assigned by the NAT process to be an address that is associated with the interface that the traffic is going through. For example, if traffic is going out an interface with the IP address 172.16.100.1, packets would be NATed so that the source IP address would be 172.16.100.1. This way the returning traffic will be directed to the same interface on the same FortiGate that the traffic left from. Even if the packets are assigned a source address that is associated with another interface on the same FortiGate this can cause issues with asymmetrical routing. It is possible to configure the NATed source IP address to be different than the IP address of the interface but you have to make sure that the routing rules of the surrounding network devices take this unorthodox approach into consideration.

There are 4 types of IP Pools that can be configured on the FortiGate firewall:

- One-to-One - in this case the only internal address used by the external address is the internal address that it is mapped to.
 - Overload - this is the default setting. Internal addresses other than the one designated in the policy can use this address for the purposes of NAT.
 - Fixed Port Range - rather than a single address to be used, there is a range of addresses that can be used as the NAT address. These addresses are randomly assigned as the connections are made.
 - Port Block Allocation - this setting is used to allocate a block of port numbers for IP pool users. Two variables will also have to be set. The block size can be set from 64 to 4096 and as the name implies describes the number of ports in one block of port numbers. The number of blocks per user determines how many of these blocks will be assigned. This number can range from 1 to 128.
-



Be careful when calculating the values of the variables. The maximum number of ports that are available on an address is 65,536. If you chose the maximum value for both variables you will get a number far in excess of the available port numbers.

$$4096 \times 128 = 524,288$$

One of the more common examples is when you have an email server behind your FortiGate firewall and the range of IP addresses assigned to you by your ISP is more than one. If an organization is assigned multiple IP addresses it is normally considered a best practice to assign a specific address other than the one used for the Firewall to the mail server. However, when normal NAT is used the address assigned to the firewall is also assigned to any outbound sessions. Anti-spam services match the source IP address of mail traffic that they receive to the MX record on DNS servers as an indicator for spam. If there is a mismatch the mail may not get through so there is a need to make sure that the NATed address assigned matches the MX record.

You can also use the Central NAT table as a way to configure IP pools.

Source IP address and IP pool address matching when using a range

When the source addresses are translated to an IP pool that is a range of addresses, one of the following three cases may occur:

Scenario 1:

The number of source addresses equals that of IP pool addresses

In this case, the FortiGate unit always matches the IP addressed one to one.

If you enable fixed port in such a case, the FortiGate unit preserves the original source port. This may cause conflicts if more than one security policy uses the same IP pool, or the same IP addresses are used in more than one IP pool.

Scenario 2:

The number of source addresses is more than that of IP pool addresses

In this case, the FortiGate unit translates IP addresses using a wrap-around mechanism. If you enable fixed port in such a case, the FortiGate unit preserves the original source port. But conflicts may occur since users may have different sessions using the same TCP 5 tuples.

Scenario 3:

The number of source addresses is fewer than that of IP pool addresses

In this case, some of the IP pool addresses are used and the rest of them are not be used.

ARP replies

If a FortiGate firewall interface IP address overlaps with one or more IP pool address ranges, the interface responds to ARP requests for all of the IP addresses in the overlapping IP pools. For example, consider a FortiGate unit with the following IP addresses for the port1 and port2 interfaces:

- port1 IP address: 1.1.1.1/255.255.255.0 (range is 1.1.1.0-1.1.1.255)
- port2 IP address: 2.2.2.2/255.255.255.0 (range is 2.2.2.0-2.2.2.255)

And the following IP pools:

- IP_pool_1: 1.1.1.10-1.1.1.20
- IP_pool_2: 2.2.2.10-2.2.2.20
- IP_pool_3: 2.2.2.30-2.2.2.40

The port1 interface overlap IP range with IP_pool_1 is:

$$(1.1.1.0-1.1.1.255) \text{ and } (1.1.1.10-1.1.1.20) = 1.1.1.10-1.1.1.20$$

The port2 interface overlap IP range with IP_pool_2 is:

$$(2.2.2.0-2.2.2.255) \& (2.2.2.10-2.2.2.20) = 2.2.2.10-2.2.2.20$$

The port2 interface overlap IP range with IP_pool_3 is:

$$(2.2.2.0-2.2.2.255) \& (2.2.2.30-2.2.2.40) = 2.2.2.30-2.2.2.40$$

And the result is:

- The port1 interface answers ARP requests for 1.1.1.10-1.1.1.20
- The port2 interface answers ARP requests for 2.2.2.10-2.2.2.20 and for 2.2.2.30-2.2.2.40

Select Enable NAT in a security policy and then select Dynamic IP Pool. Select an IP pool to translate the source address of packets leaving the FortiGate unit to an address randomly selected from the IP pool. Whether or not the external address of an IP Pool will respond to an ARP request can be disabled. You might want to disable the ability to responded to ARP requests so that these address cannot be used as a way into your network or show up on a port scan.

IP pools and zones

Because IP pools are associated with individual interfaces IP pools cannot be set up for a zone. IP pools are connected to individual interfaces.

Fixed port

Some network configurations do not operate correctly if a NAT policy translates the source port of packets used by the connection. NAT translates source ports to keep track of connections for a particular service.

However, enabling the use of a fixed port means that only one connection can be supported through the firewall for this service. To be able to support multiple connections, add an IP pool, and then select Dynamic IP pool in the policy. The firewall randomly selects an IP address from the IP pool and assigns it to each connection. In this case, the number of connections that the firewall can support is limited by the number of IP addresses in the IP pool.

Match-VIP

The match-vip feature allows the FortiGate unit to log virtual IP traffic that gets implicitly dropped. This feature eliminates the need to create two policies for virtual IPs; one that allows the virtual IP, and the other to get proper log entry for DROP rules.

For example, you have a virtual IP security policy and enabled the match-vip feature; the virtual IP traffic that is not matched by the policy is now caught.

The match-vip feature is available only in the CLI. By default, the feature is disabled.

Services and TCP ports

There are a number of different services and protocols in use on the Internet. The most commonly known is HTTP which is used by web servers to transmit requests and responses for unencrypted web pages. These services are set up to listen for requests on a numbered port. These services and protocols can use any port from 1 to 65,535. To keep things simple for everyone a large number of the more commonly used services started using a standardized list of ports. For instance, though it is not required, by default, most web servers listen for HTTP requests on port 80 and by default, web browsers will send HTTP traffic to port 80. If you wish to use another port such as 8080 you would put “:8080” at the end of the URL to indicate that you want the browser to use 8080 instead of the default port.

Example

Default URL for HTTP traffic when the web server is listening on the standard HTTP port:

`http://fortinet.com`

URL to the same address when the web server is listening for HTTP traffic on port 8080

`http://fortinet.com:8080`

Services represent typical traffic types and application packets that pass through the FortiGate unit. Firewall services define one or more protocols and port numbers associated with each service. Security policies use

service definitions to match session types. You can organize related services into service groups to simplify your security policy list.

Many well-known traffic types have been predefined on the FortiGate unit. If there is a service that does not appear on the list you can create a service or edit an existing one. You need to know the ports, IP addresses or protocols of that particular service or application uses, to create a service.

Best Practices



While you can edit a predefined service it is best to leave those ones alone and create a new service and name it something similar such as the same service name with a descriptive identifier appended.

Based on the previous example, instead of the name “HTTP” you could name the service “HTTP8080” or use the application that is using that port, “HTTP-Application”.

Protocol types

One of the fundamental aspects of a service is the type of protocol that is used to define it. When a service is defined one of the following categories of protocol needs to be determined:

- TCP/UDP/SCTP
- ICMP
- ICMPv6
- IP

Depending on which of these protocol categories is chosen another set of specifications will also be defined.

Protocol Type	Related specifications
TCP/UDP/SCTP	This is the most commonly used service protocol category. Once this category has been selected the other available options to choose are an address, either IP or FQDN, and the protocol and port number. The protocol will be TCP, UDP or SCTP.
ICMP or ICMP6	When ICMP or ICMP6 is chosen the available options are the ICMP Type and its code.
IP	When IP is the chosen protocol type the additional option is the Protocol Number.

TCP/UDP/SCTP

TCP

Transmission Control Protocol (TCP) is one of the core or fundamental protocols of the Internet. It is part of the Transport Layer of the OSI Model. It is designed to provide reliable delivery of data from a program on one device on the network or Internet to another program on another device on the network or Internet. TCP achieves its reliability because it is a connection based protocol. TCP is stream-oriented. It transports streams of data reliably and in order.

TCP establishes a prior connection link between the hosts before sending data. This is often referred to as the handshake. Once the link is established the protocol uses checks to verify that the data transmitted. If an error check fails the data is retransmitted. This makes sure that the data is getting to the destination error free and in the correct order so that it can be put back together into a form that is identical to the way they were sent.

TCP is configured more for reliability than for speed and because of this TCP will likely be slower than a connectionless protocol such as UDP. This is why TCP is generally not used for real time applications such as voice communication or online gaming.

Some of the applications that use TCP are:

- World Wide Web (HTTP and HTTPS)
- Email (SMTP, POP3, IMAP4)
- Remote administration (RDP)
- File transfer (FTP)

UDP

User Datagram Protocol (UDP) like TCP is one of the core protocols of the Internet and part of the Transport Layer of the OSI Model. UDP is designed more for speed than reliability and is generally used for different applications than TCP. UDP sends messages, referred to as datagrams across the network or Internet to other hosts without establishing a prior communication link. In other words, there is no handshake.

UDP is an unreliable service as the datagrams can arrive out of order, duplicated or go missing without any mechanism to verify them. UDP works on the assumption that any error checking is done by the application or is not necessary for the function of the application. This way it avoids the overhead that is required to verify the integrity of the data.

This lack of overhead improves the speed of the data transfer and is why UDP is often used by applications that are time sensitive in nature. UDP's stateless nature is also great for applications that answer a large number of small queries from a large number of clients.

Common uses for UDP are:

- Domain Name Resolution (DNS)
- Time (NTP)
- Streaming media (RTSP, RTP and RTCP)
- Telephone of the Internet (VoIP)
- File Transfer (TFTP)
- Logging (SNMP)
- Online games (GTP and OGP)

SCTP

Stream Control Transmission Protocol (SCTP) is part of the Transport Layer of the OSI Model just like TCP and UDP and provides some of the features of both of those protocols. It is message or datagram orientated like UDP but it also ensures reliable sequential transport of data with congestion control like TCP.

SCTP provides the following services:

- Acknowledged error-free non-duplicated transfer of user data
- Data fragmentation to conform to discovered path MTU size

- Sequenced delivery of user messages within multiple streams, with an option for order-of-arrival delivery of individual user messages
- Optional bundling of multiple user messages into a single SCTP packet
- Network-level fault tolerance through supporting of multi-homing at either or both ends of an association
- Congestion avoidance behavior and resistance to flooding and masquerade attacks

SCTP uses multi-streaming to transport its messages which means that there can be several independent streams of messages traveling in parallel between the points of the transmission. The data is sent out in larger chunks of data than is used by TCP just like UDP but the messages include a sequence number within each message in the same way that TCP does so that the data can be reassembled at the other end of the transmission in the correct sequence without the data having to arrive in the correct sequence.

SCTP is effective as the transport protocol for applications that require monitoring and session-loss detection. For such applications, the SCTP path and session failure detection mechanisms actively monitor the connectivity of the session. SCTP differs from TCP in having multi-homing capabilities at either or both ends and several streams within a connection, typically referred to as an association. A TCP stream represents a sequence of bytes; an SCTP stream represents a sequence of messages.

Some common applications of SCTP include supporting transmission of the following protocols over IP networks:

- SCTP is important in 3G and 4G/LTE networks (for example, HomeNodeB = FemtoCells)
- SS7 over IP (for example, for 3G mobile networks)
- SCTP is also defined and used for SIP over SCTP and H.248 over SCTP
- Transport of Public Switched Telephone Network (PSTN) signaling messages over IP networks.

SCTP is a much newer protocol. It was defined by the IETF Signaling Transport (SIGTRAN) working group in 2000. It was introduced by RFC 3286 and more fully define by RFC 4960.

The FortiGate firewall can apply security policies to SCTP sessions in the same way as TCP and UDP sessions. You can create security policies that accept or deny SCTP traffic by setting the service to "ALL". FortiOS does not include pre-defined SCTP services. To configure security policies for traffic with specific SCTP source or destination ports you must create custom firewall services for SCTP.

FortiGate units route SCTP traffic in the same way as TCP and UDP traffic. You can configure policy routes specifically for routing SCTP traffic by setting the protocol number to 132. SCTP policy routes can route SCTP traffic according to the destination port of the traffic if you add a port range to the policy route.

You can configure a FortiGate unit to perform stateful inspection of different types of SCTP traffic by creating custom SCTP services and defining the port numbers or port ranges used by those services. FortiGate units support SCTP over IPv4. The FortiGate unit performs the following checks on SCTP packets:

- Source and Destination Port and Verification Tag.
- Chunk Type, Chunk Flags and Chunk Length
- Verify that association exists
- Sequence of Chunk Types (INIT, INIT ACK, etc)
- Timer checking
- Four way handshake checking
- Heartbeat mechanism
- Protection against INIT/ACK flood DoS attacks, and long-INIT flooding
- Protection against association hijacking

FortiOS also supports SCTP sessions over IPsec VPN tunnels, as well as full traffic and event logging for SCTP sessions.

Protocol port values

The source and destination ports for TCP/UDP/SCTP services are important to get correct. If they are reversed the service will not work. The destination port(s) are the ones that refer to the ports that the computer will be listening on. These are the port numbers that most people are familiar with when they associate a port number to a protocol. In most cases the source port will be one that is randomly assigned by the computer that is not being already used by another service.

Most people associate HTTP with port 80. This means that a web-server will be listening on port 80 for any http requests being sent to the computer. The computer that is sending the request can use any port that is not already assigned to another service or communication session. There are 65,535 ports that it can randomly assign, but because the ports from 1 to 1024 are normally used for listening for incoming communications it is usually not in that range. It is unless there is a specific instance when you know that a communication will be coming from a predefined source port it is best practice to set the source port range from 1 to 65,535.

ICMP

The Internet Control Message Protocol (ICMP) is a protocol layered onto the Internet Protocol Suite to provide error reporting, flow control and first-hop gateway redirection. It is normally used by the operating systems of networked computers to send connectivity status query, response and error messages. It is assigned protocol number 1. There is a separate version of the protocol for both IPv4 and for IPv6. It is not designed to be absolutely reliable like TCP.

ICMP is not typically used for transporting data or for end-user network applications with the exception of some diagnostic utilities such as ping and traceroute.

ICMP messages are sent in several situations, for example:

- when a datagram cannot reach its destination,
- time exceeded messages
- redirect messages
- when the gateway does not have the buffering capacity to forward a datagram
- when the gateway can direct the host to send traffic on a shorter route.

Some of the specific ICMP message types are:

- ICMP_ECHO
- ICMP_TIMESTAMP
- ICMP_INFO_REQUEST
- ICMP_ADDRESS

For ICMP error messages, only those reporting an error for an existing session can pass through the firewall. The security policy will allow traffic to be routed, forwarded or denied. If allowed, the ICMP packets will start a new session. Only ICMP error messages of a corresponding security policy is available will be sent back to the source. Otherwise, the packet is dropped. That is, only ICMP packets for a corresponding security policy can traverse the FortiGate unit.

ICMP types and codes

ICMP has a number of messages that are identified by the “Type” field. Some of these types have assigned “Code” fields as well. The table below shows the different types of ICMP Types with their associated codes if there are any.

ICMP types and codes

Type Number	Type Name	Optional Code(s)
0	Echo Reply	
1	Unassigned	
2	Unassigned	0 Net Unreachable 1 Host Unreachable 2 Protocol Unreachable 3 Port Unreachable 4 Fragmentation Needed and Don't Fragment was Set 5 Source Route Failed 6 Destination Network Unknown 7 Destination Host Unknown
3	Destination Unreachable	8 Source Host Isolated 9 Communication with Destination Network is Administratively Prohibited 10 Communication with Destination Host is Administratively Prohibited 11 Destination Network Unreachable for Type of Service 12 Destination Host Unreachable for Type of Service 13 Communication Administratively Prohibited 14 Host Precedence Violation 15 Precedence cutoff in effect
4	Source Quench	

Type Number	Type Name	Optional Code(s)
5	Redirect	0 Redirect Datagram for the Network (or subnet)
		1 Redirect Datagram for the Host
		2 Redirect Datagram for the Type of Service and Network
		3 Redirect Datagram for the Type of Service and Host
6	Alternate Host Address	
7	Unassigned	
8	Echo	
9	Router Advertisement	
10	Router Selection	
11	Time Exceeded	0 Time to Live exceeded in Transit
		1 Fragment Reassembly Time Exceeded
12	Parameter Problem	0 Pointer indicates the error
		1 Missing a Required Option
		2 Bad Length
13	Timestamp	
14	Timestamp Reply	
15	Information Request	
16	Information Reply	
17	Address Mask Request	
18	Address Mask Reply	
19	Reserved (for Security)	

Type Number	Type Name	Optional Code(s)
20 - 29	Reserved (for Robustness Experiment)	
30	Traceroute	
31	Datagram Conversion Error	
32	Mobile Host Redirect	
33	IPv6 Where-Are-You	
34	IPv6 I-Am-Here	
35	Mobile Registration	
36	Mobile Registration Reply	
37	Domain Name Request	
38	Domain Name Reply	
39	SKIP	
40	Photuris	
41 - 255	Reserved	

log-invalid-packet

The `log-invalid-packet` CLI setting is one that is intended to log invalid ICMP packets. The exact definition being:

If the FortiGate unit receives an ICMP error packet that contains an embedded IP (A, B) | TCP (C, D) header, then if FortiOS can locate the A:C -> B:D session it checks to make sure that the sequence number in the TCP header is within the range recorded in the session. If the sequence number is not in range then the ICMP packet is dropped.

When this field is enabled, the FortiGate also log messages that are not ICMP error packets.

Types of logs covered by `log-invalid-packet`

- Invalid ICMP
 - If ICMP error message verification (see "check-reset-range") is enabled
- Invalid DNS packets
 - DNS packets that contain requests for non-existing domains
- `iprope` check failed
- reverse path check fail
- denied and broadcast traffic
- no session matched

Some other examples of messages that are not errors that will be logged, based on [RFC792](#):

Type 3 messages correspond to "Destination Unreachable Message"

- Type 3, Code 1 = host unreachable
- Type 3, Code 3 = port unreachable

Type 11 messages correspond to "Time Exceeded Message"

- Type 11, Code 0 = time to live exceeded in transit

ICMPv6

Internet Control Message Protocol version 6 (ICMPv6) is the new implementation of the Internet Control Message Protocol (ICMP) that is part of Internet Protocol version 6 (IPv6). The ICMPv6 protocol is defined in [RFC 4443](#).

ICMPv6 is a multipurpose protocol. It performs such things as:

- error reporting in packet processing
- diagnostic functions
- Neighbor Discovery process
- IPv6 multicast membership reporting

It also designed as a framework to use extensions for use with future implementations and changes.

Examples of extensions that have already been written for ICMPv6:

- Neighbor Discovery Protocol (NDP) - a node discovery protocol in IPv6 which replaces and enhances functions of ARP.
- Secure Neighbor Discovery Protocol (SEND) - an extension of NDP with extra security.
- Multicast Router Discovery (MRD) - allows discovery of multicast routers.

ICMPv6 messages use IPv6 packets for transportation and can include IPv6 extension headers. ICMPv6 includes some of the functionality that in IPv4 was distributed among protocols such as ICMPv4, ARP (Address Resolution Protocol), and IGMP (Internet Group Membership Protocol version 3).

ICMPv6 has simplified the communication process by eliminating obsolete messages.

ICMPv6 messages are subdivided into two classes: error messages and information messages.

Error Messages are divided into four categories:

1. Destination Unreachable
2. Time Exceeded

3. Packet Too Big
4. Parameter Problems

Information messages are divided into three groups:

1. Diagnostic messages
2. Neighbor Discovery messages
3. Messages for the management of multicast groups.

ICMPv6 types and codes

ICMPv6 has a number of messages that are identified by the “Type” field. Some of these types have assigned “Code” fields as well. The table below shows the different types of ICMP Types with their associated codes if there are any.

Type codes 0 – 127 are error messages and type codes 128 – 255 are for information messages.

ICMPv6 types and codes

Type Number	Type Name	Code
0	Reserved	0 - no route to destination 1 - communication with destination administratively prohibited 2 - beyond scope of source address 3 - address unreachable 4 - port unreachable 5 - source address failed ingress/egress policy 6 - reject route to destination 7 - Error in Source Routing Header
1	Destination Unreachable	
2	Packet Too Big	
3	Time Exceeded	0 - hop limit exceeded in transit 1 - fragment reassembly time exceeded
4	Parameter Problem	0 - erroneous header field encountered 1 - unrecognized Next Header type encountered 2 - unrecognized IPv6 option encountered

Type Number	Type Name	Code
100	Private Experimentation	
101	Private Experimentation	
102 - 126	Unassigned	
127	Reserved for expansion if ICMPv6 error messages	
128	Echo Request	
129	Echo Reply	
130	Multicast Listener Query	
131	Multicast Listener Report	
132	Multicast Listener Done	
133	Router Solicitation	
134	Router Advertisement	
135	Neighbor Solicitation	
136	Neighbor Advertisement	
137	Redirect Message	
138	0 - Router Renumbering Command	
	1 - Router Renumbering Result	
	255 - Sequence Number Reset	

Type Number	Type Name	Code
139	ICMP Node Information Query	0 - The Data field contains an IPv6 address which is the Subject of this Query. 1 - The Data field contains a name which is the Subject of this Query, or is empty, as in the case of a NOOP. 2 - The Data field contains an IPv4 address which is the Subject of this Query.
140	ICMP Node Information Response	0 - A successful reply. The Reply Data field may or may not be empty. 1 - The Responder refuses to supply the answer. The Reply Data field will be empty. 2 - The Qtype of the Query is unknown to the Responder. The Reply Data field will be empty.
141	Inverse Neighbor Discovery Solicitation Message	
142	Inverse Neighbor Discovery Advertisement Message	
143	Version 2 Multicast Listener Report	
144	Home Agent Address Discovery Request Message	
145	Home Agent Address Discovery Reply Message	
146	Mobile Prefix Solicitation	
147	Mobile Prefix Advertisement	
148	Certification Path Solicitation Message	

Type Number	Type Name	Code
149	Certification Path Advertisement Message	
150	ICMP messages utilized by experimental mobility protocols such as Seamoby	
151	Multicast Router Advertisement	
152	Multicast Router Solicitation	
153	Multicast Router Termination	
154	FMIPv6 Messages	
155	RPL Control Message	
156	ILNPv6 Locator Update Message	
157	Duplicate Address Request	
158	Duplicate Address Confirmation	
159 – 199	Unassigned	
200	Private experimentation	
201	Private experimentation	
255	Reserved for expansion of ICMPv6 informational messages	

IP

Internet Protocol (IP) is the primary part of the Network Layer of the OSI Model that is responsible for routing traffic across network boundaries. It is the protocol that is responsible for addressing. IPv4 is probably the version that most people are familiar with and it has been around since 1974. IPv6 is its current successor and due to a shortage of available IPv4 addresses compared to the explosive increase in the number of devices that use IP addresses, IPv6 is rapidly increasing in use.

When IP is chosen as the protocol type the available option to further specify the protocol is the protocol number. This is used to narrow down which protocol within the Internet Protocol Suite and provide a more granular control.

Protocol number

IP is responsible for more than the address that it is most commonly associated with and there are a number of associated protocols that make up the Network Layer. While there are not 256 of them, the field that identifies them is a numeric value between 0 and 256.

In the Internet Protocol version 4 (IPv4) [RFC791] there is a field called “Protocol” to identify the next level protocol. This is an 8 bit field. In Internet Protocol version 6 (IPv6) [RFC2460], this field is called the “Next Header” field.

Protocol numbers

#	Protocol	Protocol's Full Name
0	HOPOPT	IPv6 Hop-by-Hop Option
1	ICMP	Internet Control Message Protocol
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4	IPv4	IPv4 encapsulation Protocol
5	ST	Stream
6	TCP	Transmission Control Protocol
7	CBT	CBT
8	EGP	Exterior Gateway Protocol
9	IGP	Any private interior gateway (used by Cisco for their IGRP)
10	BBN-RCC-MON	BBN RCC Monitoring
11	NVP-II	Network Voice Protocol
12	PUP	PUP

#	Protocol	Protocol's Full Name
13	ARGUS	ARGUS
14	EMCON	EMCON
15	XNET	Cross Net Debugger
16	CHAOS	Chaos
17	UDP	User Datagram Protocol
18	MUX	Multiplexing
19	DCN-MEAS	DCN Measurement Subsystems
20	HMP	Host Monitoring
21	PRM	Packet Radio Measurement
22	XNS-IDP	XEROX NS IDP
23	TRUNK-1	Trunk-1
24	TRUNK-2	Trunk-2
25	LEAF-1	Leaf-1
26	LEAF-2	Leaf-2
27	RDP	Reliable Data Protocol
28	IRTP	Internet Reliable Transaction
29	ISO-TP4	ISO Transport Protocol Class 4
30	NETBLT	Bulk Data Transfer Protocol
31	MFE-NSP	MFE Network Services Protocol
32	MERIT-INP	MERIT Internodal Protocol
33	DCCP	Datagram Congestion Control Protocol
34	3PC	Third Party Connect Protocol
35	IDPR	Inter-Domain Policy Routing Protocol
36	XTP	XTP

#	Protocol	Protocol's Full Name
37	DDP	Datagram Delivery Protocol
38	IDPR-CMTP	IDPR Control Message Transport Proto
39	TP++	TP++ Transport Protocol
40	IL	IL Transport Protocol
41	IPv6	IPv6 encapsulation
42	IPv6	SDRPSource Demand Routing Protocol
43	IPv6-Route	Routing Header for IPv6
44	IPv6-Frag	Fragment Header for IPv6
45	IDRP	Inter-Domain Routing Protocol
46	RSVP	Reservation Protocol
47	GRE	General Routing Encapsulation
48	DSR	Dynamic Source Routing Protocol
49	BNA	BNA
50	ESP	Encap Security Payload
51	AH	Authentication Header
52	I-NLSP	Integrated Net Layer Security TUBA
53	SWIPE	IP with Encryption
54	NARP	NBMA Address Resolution Protocol
55	MOBILE	IP Mobility
56	TLSP	Transport Layer Security Protocol using Kryptonet key management
57	SKIP	SKIP
58	IPv6-ICMP	ICMP for IPv6
59	IPv6-NoNxt	No Next Header for IPv6
60	IPv6-Opts	Destination Options for IPv6

#	Protocol	Protocol's Full Name
61		any host internal protocol
62	CFTP	CFTP
63		any local network
64	SAT-EXPAK	SATNET and Backroom EXPAK
65	KRYPTOLAN	Kryptolan
66	RVD	MIT Remote Virtual Disk Protocol
67	IPPC	Internet Pluribus Packet Core
68		any distributed file system
69	SAT-MON	SATNET Monitoring
70	VISA	VISA Protocol
71	IPCV	Internet Packet Core Utility
72	CPNX	Computer Protocol Network Executive
73	CPHB	Computer Protocol Heart Beat
74	WSN	Wang Span Network
75	PVP	Packet Video Protocol
76	BR-SAT-MON	Backroom SATNET Monitoring
77	SUN-ND	SUN ND PROTOCOL-Temporary
78	WB-MON	WIDEBAND Monitoring
79	WB-EXPAK	WIDEBAND EXPAK
80	ISO-IP	ISO Internet Protocol
81	VMTP	VMTP
82	SECURE-VMTP	SECURE-VMTP
83	VINES	VINES
84	TTP	TTP

#	Protocol	Protocol's Full Name
84	IPTM	Protocol Internet Protocol Traffic
85	NSFNET-IGP	NSFNET-IGP
86	DGP	Dissimilar Gateway Protocol
87	TCF	TCF
88	EIGRP	EIGRP
89	OSPFIGP	OSPFIGP
90	Sprite-RPC	Sprite RPC Protocol
91	LARP	Locus Address Resolution Protocol
92	MTP	Multicast Transport Protocol
93	AX.25	AX.25 Frames
94	IPIP	IP-within-IP Encapsulation Protocol
95	MICP	Mobile Internetworking Control Pro.
96	SCC-SP	Semaphore Communications Sec. Pro.
97	ETHERIP	Ethernet-within-IP Encapsulation
98	ENCAP	Encapsulation Header
99		any private encryption scheme
100	GMTP	GMTP
101	IFMP	Ipsilon Flow Management Protocol
102	PNNI	PNNI over IP
103	PIM	Protocol Independent Multicast
104	ARIS	ARIS
105	SCPS	SCPS
106	QNX	QNX
107	A/N	Active Networks

#	Protocol	Protocol's Full Name
108	IPComp	IP Payload Compression Protocol
109	SNP	Sitara Networks Protocol
110	Compaq-Peer	Compaq Peer Protocol
111	IPX-in-IP	IPX in IP
112	VRRP	Virtual Router Redundancy Protocol
113	PGM	PGM Reliable Transport Protocol
114		any 0-hop protocol
115	L2TP	Layer Two Tunneling Protocol
116	DDX	D-II Data Exchange (DDX)
117	IATP	Interactive Agent Transfer Protocol
118	STP	Schedule Transfer Protocol
119	SRP	SpectraLink Radio Protocol
120	UTI	UTI
121	SMP	Simple Message Protocol
122	SM	SM
123	PTP	Performance Transparency Protocol
124		ISIS over IPv4
125	FIRE	
126	CRTP	Combat Radio Transport Protocol
127	CRUDP	Combat Radio User Datagram
128	SSCOPMCE	
129	IPLT	
130	SPS	Secure Packet Shield
131	PIPE	Private IP Encapsulation within IP

#	Protocol	Protocol's Full Name
132	SCTP	Stream Control Transmission Protocol
133	FC	Fibre Channel
134	RSVP-E2E-IGNORE	
135	Mobility Header	
136	UDPLite	
137	MPLS-in-IP	
138	manet	
139	HIP	
140	Shim6	
141	WESP	
142	ROHC	
143 – 252	Unassigned	Unassigned
253		Use for experimentation and testing
254		Use for experimentation and testing
255	Reserved	

Further information can be found by researching RFC 5237.

Protocol number

IP is responsible for more than the address that it is most commonly associated with and there are a number of associated protocols that make up the Network Layer. While there are not 256 of them, the field that identifies them is a numeric value between 0 and 256.

In the Internet Protocol version 4 (IPv4) [RFC791] there is a field called “Protocol” to identify the next level protocol. This is an 8 bit field. In Internet Protocol version 6 (IPv6) [RFC2460], this field is called the “Next Header” field.

Protocol numbers

#	Protocol	Protocol's Full Name
0	HOPOPT	IPv6 Hop-by-Hop Option

#	Protocol	Protocol's Full Name
1	ICMP	Internet Control Message Protocol
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4	IPv4	IPv4 encapsulation Protocol
5	ST	Stream
6	TCP	Transmission Control Protocol
7	CBT	CBT
8	EGP	Exterior Gateway Protocol
9	IGP	Any private interior gateway (used by Cisco for their IGRP)
10	BBN-RCC-MON	BBN RCC Monitoring
11	NVP-II	Network Voice Protocol
12	PUP	PUP
13	ARGUS	ARGUS
14	EMCON	EMCON
15	XNET	Cross Net Debugger
16	CHAOS	Chaos
17	UDP	User Datagram Protocol
18	MUX	Multiplexing
19	DCN-MEAS	DCN Measurement Subsystems
20	HMP	Host Monitoring
21	PRM	Packet Radio Measurement
22	XNS-IDP	XEROX NS IDP
23	TRUNK-1	Trunk-1
24	TRUNK-2	Trunk-2

#	Protocol	Protocol's Full Name
25	LEAF-1	Leaf-1
26	LEAF-2	Leaf-2
27	RDP	Reliable Data Protocol
28	IRTP	Internet Reliable Transaction
29	ISO-TP4	ISO Transport Protocol Class 4
30	NETBLT	Bulk Data Transfer Protocol
31	MFE-NSP	MFE Network Services Protocol
32	MERIT-INP	MERIT Internodal Protocol
33	DCCP	Datagram Congestion Control Protocol
34	3PC	Third Party Connect Protocol
35	IDPR	Inter-Domain Policy Routing Protocol
36	XTP	XTP
37	DDP	Datagram Delivery Protocol
38	IDPR-CMTP	IDPR Control Message Transport Proto
39	TP++	TP++ Transport Protocol
40	IL	IL Transport Protocol
41	IPv6	IPv6 encapsulation
42	IPv6	SDRPSource Demand Routing Protocol
43	IPv6-Route	Routing Header for IPv6
44	IPv6-Frag	Fragment Header for IPv6
45	IDRP	Inter-Domain Routing Protocol
46	RSVP	Reservation Protocol
47	GRE	General Routing Encapsulation
48	DSR	Dynamic Source Routing Protocol

#	Protocol	Protocol's Full Name
49	BNA	BNA
50	ESP	Encap Security Payload
51	AH	Authentication Header
52	I-NLSP	Integrated Net Layer Security TUBA
53	SWIPE	IP with Encryption
54	NARP	NBMA Address Resolution Protocol
55	MOBILE	IP Mobility
56	TLSP	Transport Layer Security Protocol using Kryptonet key management
57	SKIP	SKIP
58	IPv6-ICMP	ICMP for IPv6
59	IPv6-NoNxt	No Next Header for IPv6
60	IPv6-Opts	Destination Options for IPv6
61		any host internal protocol
62	CFTP	CFTP
63		any local network
64	SAT-EXPAK	SATNET and Backroom EXPAK
65	KRYPTOLAN	Kryptolan
66	RVD	MIT Remote Virtual Disk Protocol
67	IPPC	Internet Pluribus Packet Core
68		any distributed file system
69	SAT-MON	SATNET Monitoring
70	VISA	VISA Protocol
71	IPCV	Internet Packet Core Utility
72	CPNX	Computer Protocol Network Executive

#	Protocol	Protocol's Full Name
73	CPHB	Computer Protocol Heart Beat
74	WSN	Wang Span Network
75	PVP	Packet Video Protocol
76	BR-SAT-MON	Backroom SATNET Monitoring
77	SUN-ND	SUN ND PROTOCOL-Temporary
78	WB-MON	WIDEBAND Monitoring
79	WB-EXPAK	WIDEBAND EXPAK
80	ISO-IP	ISO Internet Protocol
81	VMTP	VMTP
82	SECURE-VMTP	SECURE-VMTP
83	VINES	VINES
84	TTP	TTP
84	IPTM	Protocol Internet Protocol Traffic
85	NSFNET-IGP	NSFNET-IGP
86	DGP	Dissimilar Gateway Protocol
87	TCF	TCF
88	EIGRP	EIGRP
89	OSPFIGP	OSPFIGP
90	Sprite-RPC	Sprite RPC Protocol
91	LARP	Locus Address Resolution Protocol
92	MTP	Multicast Transport Protocol
93	AX.25	AX.25 Frames
94	IPIP	IP-within-IP Encapsulation Protocol
95	MICP	Mobile Internetworking Control Pro.

#	Protocol	Protocol's Full Name
96	SCC-SP	Semaphore Communications Sec. Pro.
97	ETHERIP	Ethernet-within-IP Encapsulation
98	ENCAP	Encapsulation Header
99		any private encryption scheme
100	GMTP	GMTP
101	IFMP	Ipsilon Flow Management Protocol
102	PNNI	PNNI over IP
103	PIM	Protocol Independent Multicast
104	ARIS	ARIS
105	SCPS	SCPS
106	QNX	QNX
107	A/N	Active Networks
108	IPComp	IP Payload Compression Protocol
109	SNP	Sitara Networks Protocol
110	Compaq-Peer	Compaq Peer Protocol
111	IPX-in-IP	IPX in IP
112	VRRP	Virtual Router Redundancy Protocol
113	PGM	PGM Reliable Transport Protocol
114		any 0-hop protocol
115	L2TP	Layer Two Tunneling Protocol
116	DDX	D-II Data Exchange (DDX)
117	IATP	Interactive Agent Transfer Protocol
118	STP	Schedule Transfer Protocol
119	SRP	SpectralLink Radio Protocol

#	Protocol	Protocol's Full Name
120	UTI	UTI
121	SMP	Simple Message Protocol
122	SM	SM
123	PTP	Performance Transparency Protocol
124	ISIS over IPv4	
125	FIRE	
126	CRTP	Combat Radio Transport Protocol
127	CRUDP	Combat Radio User Datagram
128	SSCOPMCE	
129	IPLT	
130	SPS	Secure Packet Shield
131	PIPE	Private IP Encapsulation within IP
132	SCTP	Stream Control Transmission Protocol
133	FC	Fibre Channel
134	RSVP-E2E-IGNORE	
135	Mobility Header	
136	UDPLite	
137	MPLS-in-IP	
138	manet	
139	HIP	
140	Shim6	
141	WESP	
142	ROHC	
143 – 252	Unassigned	Unassigned

#	Protocol	Protocol's Full Name
253		Use for experimentation and testing
254		Use for experimentation and testing
255	Reserved	

Further information can be found by researching RFC 5237.

VPN policies

At one point, if you wanted to have secure digital communications between 2 points a private network would be created. This network would only allow the people that were intended to get the communications on it. This is very straightforward if the 2 points are in the same room or even in the same building. It can all be done physically. If you are supposed to be on the secure network

VPNs are an answer to one of today's biggest concerns, how to make digital communications secure between 2 points that must communicate over the Internet which anybody can have access to.

There are two types of VPNs supported by FortiOS, SSL and IPsec. They are differentiated by the security protocol suites that are used to secure the traffic. These are both described in more detail in the VPN section, but the IPsec VPN can be configured as an **Action** with a firewall policy.

IPsec policies

IPsec policies allow IPsec VPN traffic access to the internal network from a remote location. These policies include authentication information that authenticates users and user group or groups. These policies specify the following:

- the FortiGate firewall interface that provides the physical connection to the remote VPN gateway, usually an interface connected to the Internet
- the FortiGate firewall interface that connects to the private network
- IP addresses associated with data that has to be encrypted and decrypted
- optional: a schedule that restricts when the VPN can operate, and services (or types of data) that can be sent.

For a route-based (interface mode) VPN, you do not configure an IPsec security policy. Instead, you configure two regular ACCEPT security policies, one for each direction of communication, with the IPsec virtual interface as the source or destination interface, as appropriate.

DSRI

The Disable Server Response Inspection (DSRI) options is available for configuration in the CLI. This is used to assist performance when only URL filtering is being used. This allows the system to ignore the HTTP server responses. The setting is configured to be disabled by default.

CLI syntax for changing the status of the DSRI setting

In IPv4 or IPv6 firewall policies

```
config firewall policy|policy6
  edit 0
    set dsri enable|disable
  end
```

In IPv4 or IPv6 interface policies

```
config firewall interface-policy|interface-policy6
  edit 0
    set dsri enable|disable
  end
```

When using the sniffer

```
config firewall sniffer
  edit 0
    set dsri enable|disable
  end
```

Interface policies

Interface policies are implemented before the “security” policies and are only flow based. They are configured in the CLI.

This feature allows you to attach a set of IPS policies with the interface instead of the forwarding path, so packets can be delivered to IPS before entering firewall. This feature is used for following IPS deployments:

- One-Arm: by defining interface policies with IPS and DoS anomaly checks and enabling sniff-mode on the interface, the interface can be used for one-arm IDS;
- IPv6 IPS: IPS inspection can be enabled through interface IPv6 policy. Only IPS signature scan is supported in FortiOS 4.0. IPv6 DoS protection is not supported;
- Scan traffics that destined to FortiGate;
- Scan and log traffics that are silently dropped or flooded by Firewall or Multicast traffic.

IPS sensors can be assigned to an interface policy. Both incoming and outgoing packets are inspected by IPS sensor (signature).

Here is an example of an interface policy,

```
# show full-configuration
```

```
config firewall interface-policy
  edit 1
    set status enable
    set comments 'test interface policy #1'
    set logtraffic utm
    set interface "port9"
    set srcaddr "all"
    set dstaddr "all"
```

```
set service "ALL"
set application-list-status disable
set ips-sensor-status disable
set dsri disable
set av-profile-status enable
set av-profile "default"
set webfilter-profile-status disable
set spamfilter-profile-status disable
set dlp-sensor-status disable
set scan-botnet-connections disable
next
end
```

DoS protection

Denial of Service (DoS) policies are primarily used to apply DoS anomaly checks to network traffic based on the FortiGate interface it is entering as well as the source and destination addresses. DoS checks are a traffic anomaly detection feature to identify network traffic that does not fit known or common traffic patterns and behavior. A common example of anomalous traffic is the denial of service attack. A denial of service occurs when an attacking system starts an abnormally large number of sessions with a target system. The large number of sessions slows down or disables the target system, so that legitimate users can no longer use it.

DoS policies are similar to firewall policies except that instead of defining the way traffic is allowed to flow, they keep track of certain traffic patterns and attributes and will stop traffic displaying those attributes. Further, DoS policies affect only incoming traffic on a single interface. You can further limit a DoS policy by source address, destination address, and service.

DoS configurations have been changed a couple of times in the past. In FortiOS 4.0, DoS protection is moved to the interface policy, so when it is enabled, it is the first thing checked when a packet enters FortiGate. Because of this early detection, DoS policies are a very efficient defense that uses few resources. Denial of service attacks, for example, are detected and its packets dropped before requiring security policy look-ups, antivirus scans, and other protective but resource-intensive operations.

A DoS policy examines network traffic arriving at an interface for anomalous patterns usually indicating an attack. This does not mean that all anomalies experience by the firewall are the result of an intentional attack.

Because an improperly configured DoS anomaly check can interfere with network traffic, no DoS checks are preconfigured on a factory default FortiGate unit. You must create your own before they will take effect. Thresholds for newly created sensors are preset with recommended values that you can adjust to meet the needs of your network.

To create a Denial of Service policy determine if it needs to be an IPv4 or IPv6 policy, then go to:

Policy & Objects > IPv4 DoS Policy for IPv4.

Policy & Objects > IPv6 DoS Policy for IPv6.



The **Enable SSH Deep Scan** feature is enabled by default when creating a new SSL/SSH Inspection profile. There are situations where this feature can cause issues so be sure that you would like it enabled before applying it.

Settings used in configuring DoS

Incoming interface

The interface to which this security policy applies. It will be the that the traffic is coming into the firewall on.

Source address

This will be the address that the traffic is coming from and must be a address listed in the Address section of the Firewall Objects. This can include the predefined “all” address which covers any address coming in on any interface. Multiple addresses or address groups can be chosen

Destination address

This will be the address that the traffic is addressed to. In this case it must be an address that is associated with the firewall itself. For instance it could be one of the interface address of the firewall, a secondary IP address or the interface address assigned to a Virtual IP address. Just like with the Source Address this address must be already configured before being used in the DoS policy. Multiple addresses, virtual IPs or virtual IP groups can be chosen.

Service

While the Service field allows for the use of the ALL service some administrators prefer to optimize the resources of the firewall and only check on the services that will be answered on an interface. Multiple services or service groups can be chosen.

Anomalies

The anomalies can not be configured by the user. They are predefined sensors set up for specific patterns of anomalous traffic

The anomalies that have been predefined for use in the DoS Policies are:

Anomaly Name	Description	Recommended Threshold
tcp_syn_flood	If the SYN packet rate of new TCP connections, including retransmission, to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second.
tcp_port_scan	If the SYN packet rate of new TCP connections, including retransmission, from one source IP address exceeds the configured threshold value, the action is executed.	1000 packets per second.
tcp_src_session	If the number of concurrent TCP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.

Anomaly Name	Description	Recommended Threshold
tcp_dst_session	If the number of concurrent TCP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
udp_flood	If the UDP traffic to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second.
udp_scan	If the number of UDP sessions originating from one source IP address exceeds the configured threshold value, the action is executed.	2000 packets per second.
udp_src_session	If the number of concurrent UDP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
udp_dst_session	If the number of concurrent UDP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
icmp_flood	If the number of ICMP packets sent to one destination IP address exceeds the configured threshold value, the action is executed.	250 packets per second.
icmp_sweep	If the number of ICMP packets originating from one source IP address exceeds the configured threshold value, the action is executed.	100 packets per second.
icmp_src_session	If the number of concurrent ICMP connections from one source IP address exceeds the configured threshold value, the action is executed.	300 concurrent sessions
icmp_dst_session	If the number of concurrent ICMP connections to one destination IP address exceeds the configured threshold value, the action is executed.	3000 concurrent sessions
ip_src_session	If the number of concurrent IP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
ip_dst_session	If the number of concurrent IP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
sctp_flood	If the number of SCTP packets sent to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second

Anomaly Name	Description	Recommended Threshold
sctp_scan	If the number of SCTP sessions originating from one source IP address exceeds the configured threshold value, the action is executed.	1000 packets per second
sctp_src_session	If the number of concurrent SCTP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions
sctp_dst_session	If the number of concurrent SCTP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions

Status

The status field is enabled to enable the sensor for the associated anomaly. In terms of actions performed there is no difference between disabling a sensor and having the action as “Pass” but by disabling sensors that are not being used for blocking or logging you can save some resources of the firewall that can be better used elsewhere.

Logging

Regardless of whether the traffic is blocked or passed through the anomalous traffic will be logged.

Pass

Allows the anomalous traffic to pass through unimpeded.

Block

For Thresholds based on the number of concurrent sessions blocking the anomaly will not allow more than the number of concurrent sessions set as the threshold.

For rate based thresholds where the threshold is measured in packets per second, the Action setting “Block” prevents the overwhelming of the firewall by anomalous traffic in one of 2 ways. Setting which of those 2 ways will be issued is determined in the CLI.

- continuous - blocks packets once an anomaly is detected. This overrides individual anomaly settings.
- periodical - allows matching anomalous traffic up to the rate set by the threshold.



If the period for a particular anomaly is 60 seconds, such as those where the threshold is measured in concurrent sessions, after the 60 second timer has expired, the number of allowed packets that match the anomaly criteria is reset to zero. This means that if you allow 10 sessions through before blocking, after the 60 seconds is up, another 10 will be allowed. The attrition of sessions from expiration should keep the allowed sessions from reaching the maximum.

To set the type of block action for the rate based anomaly sensors:

```
config ips global
  set anomaly-mode continuous
  set anomaly-mode periodical
end
```

Threshold

The threshold can be either in terms of concurrent session or in packets per second depending on which sensor is being referred to.

Quarantine

The quarantine feature is found in the CLI. This setting is used to block any further traffic from a source address that is now considered to be a malicious actor or a source of traffic dangerous to the network. Not only is no more traffic accepted for the duration of the quarantine through the DoS policy but the source IP address of the traffic is added to the banned source ip list. This list is kept in the kernel and used by

- Antivirus
- Data Leak Prevention (DLP)
- Denial of Service (DoS)
- Intrusion Prevention System (IPS)

Any policies that use any of these features will block traffic from the attacker's IP address.

Syntax

```
config firewall {DoS-policy|DoS-policy6}
  edit <policyid>
    set quarantine {none|attacker}
    set quarantine-expiry {string}
    set quarantine-log {enable|disable}
  end
```

Option	Description
quarantine	Quarantine method. <ul style="list-style-type: none"> • none - Quarantine is disabled. • attacker - Block all traffic sent from the attacker's IP address. The quarantined IP address is also added to the banned ip list. The destination address is not affected.
quarantine-expiry	Duration of quarantine The format is <code>##d##h##m</code> , ranging from 1 minute to 364 days, 23 hours, and 59 minutes starting from now. The default is <code>0d0h5m</code> . Requires quarantine set to attacker.
quarantine-log	Enables or disables the logging of quarantine events.

One-Arm IDS

Interface-based policy only defines what and how IPS functions are applied to the packets transmitted by the interface. It works no matter if the port is used in a forwarding path or used as an One-Arm device.

To enable One-Arm IDS, the user should first enable sniff-mode on the interface,

```
config system interface
```

```
edit port2
    set ips-sniffer-mode enable
next
end
```

Once sniff-mode is turned on, both incoming and outgoing packets will be dropped after IPS inspections. The port can be connected to a hub or a switch's SPAN port. Any packet picked up by the interface will still follow the interface policy so different IPS and DoS anomaly checks can be applied.

IPv6 IPS

IPv6 IPS signature scan can be enabled by interface policy. The user can create an normal IPS sensor and assign it to the IPv6 interface policy.

```
config firewall interface-policy6
edit 1
    set interface "port1"
    set srcaddr6 "all"
    set dstaddr6 "all"
    set service6 "ALL"
    set ips-sensor-status enable
    set ips-sensor "all_default"
next
end
```

Traffic destined to the FortiGate unit

IPS enabled in firewall policies can only inspect the traffic pass through FortiGate unit, not the traffic destined to FortiGate unit. Enabling IPS in interface-policy allows IPS to pick up any packet on the interface so it is able to inspect attacks targeting FGT.

Dropped, flooded, broadcast, multicast and L2 packets

In many evaluation or certification tests, FortiGate firewall is often required to log any packets dropped by the firewall. In most of cases, these packets are of invalid headers so firewall just drops them silently. It is natural to forward all these packets to IPS first so FortiGate firewall is able to generate logs for invalid packets.

Flooded, broadcast and multicast traffics do not reach any of services in the forwarding path. They can be inspected by the interface policy as long as they match the addresses defined. Potentially, L2 packets can also be sent to IPS for inspection through interface-policy, but it is not enabled in FortiOS 4.0.

GUI and CLI

Now in FortiGate, there are two places that IPS can be enabled, in a firewall policy and in an interface policy. In the firewall policy implementation, IPS sensor can be configured in both CLI and GUI. When adding an IPS sensor to an interface policy it must be done through the CLI. There is no GUI input window for the "Interface Policy". There is however, a DoS Policy section in the GUI.

Local-In policies

On the FortiGate unit, there are a number of protocols and traffic that is specific to the internal workings of FortiOS. For many of these traffic sources, you can identify a specific port/IP address for this self-originating traffic. The following traffic can be configured to a specific port/IP address:

- SNMP
- Syslog
- alert email
- FortiManager connection IP
- FortiGuard services
- FortiAnalyzer logging
- NTP
- DNS
- Authorization requests such as RADIUS
- FSSO

Security policies control the flow of traffic through the FortiGate unit. The FortiGate unit also includes the option of controlling internal traffic, that is, management traffic.

Each interface includes an allow access configuration to allow management access for specific protocols. Local policies are set up automatically to allow all users all access. Local-in policies takes this a step further, to enable or restrict the user with that access. This also extends beyond the allow access selection.

Local-in policies are configured in the CLI with the commands:

```
config firewall local-in-policy
  edit <policy_number>
    set intf <source_interface>
    set srcaddr <source_address>
    set dstaddr <destination_address>
    set action {accept | deny}
    set service <service name>
    set schedule <schedule_name>
    set comments <string>
  end
```

For example, you can configure a local-in policy so that only administrators can access the FortiGate unit on weekends from a specific management computer at 192.168.21.12, represented by the address object mgmt-comp1, using SSH on port 3 (192.168.21.77 represented by the address object FG-port3) using the Weekend schedule which defines the time the of access.

```
config firewall local-in-policy
  edit <1>
    set intf port3
    set srcaddr mgmt-comp1
    set dstaddr FG-port3
    set action accept
    set service SSH
    set schedule Weekend
  end
```

You can also disable a policy should there be a requirement to turn off a policy for troubleshooting or other purpose. To disable a policy enter the commands:

```
config firewall local-in-policy
edit <policy_number>
  set status disable
end
```

Use the same commands with a status of enable to use the policy again.

It is also an option to dedicate the interface as HA management interface by using the setting:

```
set ha-mgmt-intf-only enable
```

Local-in policies are also supported for IPv6 by entering the command:

```
config firewall local-in-policy6.
```



While there is a section under **Policy & Objects** for viewing the existing **Local In Policy** configuration, policies cannot be created or edited here in the GUI. The Local In policies can only be created or edited in the CLI.

Security policy 0

Any security policy that is automatically added by the FortiGate unit has a policy ID number of zero (0). The most common reasons the FortiGate unit creates this policy is:

- The IPsec policy for FortiAnalyzer (and FortiManager version 3.0) is automatically added when an IPsec connection to the FortiAnalyzer unit or FortiManager is enabled.
- The policy to allow FortiGuard servers to be automatically added has a policy ID number of zero.
- The (default) drop rule that is the last rule in the policy and that is automatically added has a policy ID number of zero.
- When a network zone is defined within a VDOM, the intra-zone traffic set to allow or block is managed by policy 0 if it is not processed by a configured security policy.

This policy can appear in logs but will never appear in the security policy list, and therefore, can never be repositioned in the list.

When viewing the FortiGate firewall logs, you may find a log field entry indicating policyid=0. The following log message example indicates the log field policyid=0 in bold.

```
2008-10-06 00:13:49 log_id=0022013001 type=traffic subtype=violation pri=warning
vd=root SN=179089 duration=0 user=N/A group=N/A rule=0 policyid=0 proto=17
service=137/udp app_type=N/A status=deny src=10.181.77.73 srcname=10.181.77.73
dst=10.128.1.161 dstname=10.128.1.161 src_int=N/A dst_int="Internal" sent=0 rcvd=0
src_port=137 dst_port=137 vpn=N/A tran_ip=0.0.0.0 tran_port=0
```

DNS traffic in NGFW policy-mode

FortiOS has an option to enable the creation of an implicit policy to allow DNS traffic.

Certain Application Control profiles may not work properly if DNS traffic is not allowed. Enabling the `implicit-allow-dns` option adds an implicit policy to allow the DNS traffic. This policy is situated in the policy sequence

just above the implicit deny policy. Since this is a `config system settings` command, this option can be enabled per VDOM.

CLI

```
config system settings
  set implicit-allow-dns {enable|disable}
end
```

Deny policies

Deny security policies deny traffic that is coming into the network. The FortiGate unit automatically blocks traffic that is associated with a deny security policy.

Deny security policies are usually configured when you need to restrict specific traffic, for example, SSH traffic. Deny security policies can also help when you want to block a service, such as DNS, but allow a specific DNS server.

Accept policies

Accept security policies accept traffic that is coming into the network. These policies allow traffic through the FortiGate unit, where the packets are scanned, translated if NAT is enabled, and then sent out to its destination.

Accept security policies are the most common security policies that are created in FortiOS. These security policies are basic policies, such as allowing Internet access, as well as complex policies, such as IPsec VPN.

Fixed port

Some network configurations do not operate correctly if a NAT policy translates the source port of packets used by the connection. NAT translates source ports to keep track of connections for a particular service.

From the CLI you can enable `fixedport` when configuring a security policy for NAT policies to prevent source port translation.

```
config firewall policy
  edit <policy-id>
    ...
    set fixedport enable
    ...
  end
```

However, enabling `fixedport` means that only one connection can be supported through the firewall for this service. To be able to support multiple connections, add an IP pool, and then select Dynamic IP pool in the policy. The firewall randomly selects an IP address from the IP pool and assigns it to each connection. In this case, the number of connections that the firewall can support is limited by the number of IP addresses in the pool.

Endpoint security

Endpoint security enforces the use of the FortiClient End Point Security (FortiClient and FortiClient Lite) application on your network. It can also allow or deny endpoints access to the network based on the application installed on them.

By applying endpoint security to a security policy, you can enforce this type of security on your network. FortiClient enforcement can check that the endpoint is running the most recent version of the FortiClient application, that the antivirus signatures are up-to-date, and that the firewall is enabled. An endpoint is usually often a single PC with a single IP address being used to access network services through a FortiGate unit.

With endpoint security enabled on a policy, traffic that attempts to pass through, the FortiGate unit runs compliance checks on the originating host on the source interface. Non-compliant endpoints are blocked. If someone is browsing the web, the endpoints are redirected to a web portal which explains the non-compliance and provides a link to download the FortiClient application installer. The web portal is already installed on the FortiGate unit, as a replacement message, which you can modify if required.

Endpoint Security requires that all hosts using the security policy have the FortiClient Endpoint Security agent installed. Currently, FortiClient Endpoint Security is available for Microsoft Windows 2000 and later only.

For more information about endpoint security, see the Security Profiles chapter in the FortiOS Handbook.

Traffic logging

When you enable logging on a security policy, the FortiGate unit records the scanning process activity that occurs, as well as whether the FortiGate unit allowed or denied the traffic according to the rules stated in the security policy. This information can provide insight into whether a security policy is working properly, as well as if there needs to be any modifications to the security policy, such as adding traffic shaping for better traffic performance.

Depending on what the FortiGate unit has in the way of resources, there may be advantages in optimizing the amount of logging taking places. This is why in each policy you are given 3 options for the logging:

- **Disable Log Allowed Traffic** - Does not record any log messages about traffic accepted by this policy.

If you enable Log Allowed Traffic, the following two options are available:

- **Security Events** - This records only log messages relating to security events caused by traffic accepted by this policy.
- **All Sessions** - This records all log messages relating to all of the traffic accepted by this policy.

Depending on the model, if the Log all Sessions option is selected there may be 2 additional options. These options are normally available in the GUI on the higher end models such as the FortiGate 600C or larger.

- **Generate Logs when Session Starts**
- **Capture Packets**

You can also use the CLI to enter the following command to write a log message when a session starts:

```
config firewall policy
  edit <policy-index>
    set logtraffic-start
  end
```

Traffic is logged in the traffic log file and provides detailed information that you may not think you need, but do. For example, the traffic log can have information about an application used (web: HTTP.Image), and whether or not the packet was SNAT or DNAT translated. The following is an example of a traffic log message.

```
2011-04-13
05:23:47
log_id=4
type=traffic
subtype=other
pri=notice
vd=root
status="start"
src="10.41.101.20"
srcname="10.41.101.20"
src_port=58115
dst="172.20.120.100"
dstname="172.20.120.100"
dst_country="N/A"
dst_port=137
tran_ip="N/A"
tran_port=0
tran_sip="10.31.101.41"
tran_sport=58115
service="137/udp"
proto=17
app_type="N/A"
duration=0
rule=1
policyid=1
sent=0
rcvd=0
shaper_drop_sent=0
shaper_drop_rcvd=0
perip_drop=0
src_int="internal"
dst_int="wan1"
SN=97404 app="N/A"
app_cat="N/A"
carrier_ep="N/A"
```

If you want to know more about logging, see the Logging and Reporting chapter in the FortiOS Handbook. If you want to know more about traffic log messages, see the FortiGate Log Message Reference.

IPv6

Internet Protocol version 6 (IPv6) will succeed IPv4 as the standard networking protocol of the Internet. IPv6 provides a number of advances over IPv4 but the primary reason for its replacing IPv4 is its limitation in addresses. IPv4 uses 32 bit addresses which means there is a theoretical limit of 2^{32} . The IPv6 address scheme is based on a 128 bit address or a theoretical limit of 2^{128} .

IPv6 addressing

Possible addresses:

- IPv4 = 4,294,967,296 (over 4 billion)
- IPv6 = 340,282,366,920,938,463,463,374,607,431,768,211,456 (over 340 undecillion - We had to look that term up. We didn't know what a number followed by 36 digits was either)

Assuming a world population of approximately 8 billion people, IPv6 would allow for each individual to have approximately 42,535,295,865,117,200,000,000,000 devices with an IP address. That's 42 quintillion devices.

There is little likelihood that you will ever need to worry about these numbers as any kind of serious limitation in addressing but they do give an idea of the scope of the difference in the available addressing.

IPv6 address syntax

Aside from the difference of possible addresses there is also the different formatting of the addresses that will need to be addressed.

A computer would view an IPv4 address as a 32 bit string of binary digits made up of 1s and 0s, broken up into 4 octets of 8 digits separated by a period “.”

Example:

10101100.00010000.11111110.00000001

To make number more user friendly for humans we translate this into decimal, again 4 octets separated by a period “.” which works out to:

172.16.254.1

A computer would view an IPv6 address as a 128 bit string of binary digits made up of 1s and 0s, broken up into 8 octets of 16 digits separated by a colon “:”

1000000000000001:000011011011000:10101100001000:111111000000001:0000000000000000
0:0000000000000000:0000000000000000:0000000000000000

To make number a little more user friendly for humans we translate this into hexadecimal, again 8 octets separated by a colon “:” which works out to:

8001:0DB8:AC10:FE01:0000:0000:0000:0000:

Because any four-digit group of zeros within an IPv6 address may be reduced to a single zero or altogether omitted, this address can be shortened further to:

8001:0DB8:AC10:FE01:0:0:0:0

or

8001:0DB8:AC10:FE01::

IPv6 packet structure

Each IPv6 packet consists of a mandatory fixed header and optional extension headers, and carries a payload, which is typically either a datagram and/or Transport Layer information. The payload could also contain data for the Internet Layer or Link Layer. Unlike IPv4, IPv6 packets aren't fragmented by routers, requiring hosts to implement Maximum Transmission Unit (MTU) Path Discovery for MTUs larger than the smallest MTU (which is 1280 octets).

Jumbograms and jumbo payloads

In IPv6, packets which exceed the MTU of the underlying network are labeled jumbograms, which consist of a jumbo payload. A jumbogram typically exceeds the IP MTU size limit of 65,535 octets, and provides the jumbo payload option, which can allow up to nearly 4GiB of payload data, as defined in [RFC 2675](#). When the MTU is determined to be too large, the receiving host sends a 'Packet too Big' ICMPv6 type 2 message to the sender.

Fragmentation and reassembly

As noted, packets that are too large for the MTU require hosts to perform MTU Path Discovery to determine the maximum size of packets to send. Packets that are too large require a 'Fragment' extension header, to divide the payload into segments that are 8 octets in length (except for the last fragment, which is smaller). Packets are reassembled according to the extension header and the fragment offset.

Benefits of IPv6

In addition to the expanded number of addresses, some of the other benefits of IPv6 include:

- More efficient routing
- Reduced management requirement
- Stateless auto-reconfiguration of hosts
- Improved methods to change Internet Service Providers
- Better mobility support
- Multi-homing
- Security
- Scoped address: link-local, site-local and global address space

IPv6 in FortiOS

From an administrative point of view IPv6 works almost the same as IPv4 in FortiOS. The primary differences are the use of IPv6 format for addresses and fewer address types for IPv6. There is also no need for NAT if the FortiGate firewall is the interface between IPv6 networks. If the subnets attached to the FortiGate firewall are IPv6 and IPv4 NAT can be configured between the 2 different formats. This will involve either configuring a dual stack routing or IPv4 tunneling configuration. The reason for this is simple. NAT was developed primarily for the purpose of extending the number of usable IPv4 addresses. IPv6's addressing allows for enough available addresses so the NAT is no longer necessary.

When configuring IPv6 in FortiOS, you can create a dual stack route or IPv4-IPv6 tunnel. A dual stack routing configuration implements dual IP layers, supporting both IPv4 and IPv6, in both hosts and routers. An IPv4-IPv6 tunnel is essentially similar, creating a tunnel that encapsulates IPv6 packets within IPv4 headers that carry these IPv6 packets over IPv4 tunnels. The FortiGate unit can also be easily integrated into an IPv6 network. Connecting the FortiGate unit to an IPv6 network is exactly the same as connecting it to an IPv4 network, the only difference is that you are using IPv6 addresses.

By default the IPv6 settings are not displayed in the Web-based Manager. It is just a matter of enabling the display of these feature to use them through the web interface. To enable them just go to **System > Feature Select** and select **IPv6**. Once enabled, you will be able to use IPv6 addresses as well as the IPv4 addressing for the following FortiGate firewall features:

- Static routing
- Policy Routing
- Packet and network sniffing
- Dynamic routing (RIPv6, BGP4+, and OSPFv3)
- IPsec VPN
- DNS
- DHCP
- SSL VPN
- Network interface addressing
- Security Profiles protection
- Routing access lists and prefix lists
- NAT and transparent mode
- NAT 64 and NAT 66
- IPv6 tunnel over IPv4 and IPv4 tunnel over IPv6
- Logging and reporting
- Security policies
- SNMP
- Authentication
- Virtual IPs and groups
- IPv6 over SCTP
- IPv6-specific troubleshooting, such as ping6

IPv6 features

In order to configure IPv6 features using the web-based manager, IPv6 must be enabled using Feature Select. Go to **System > Config > Features**, enable IPv6, and click **Apply**.

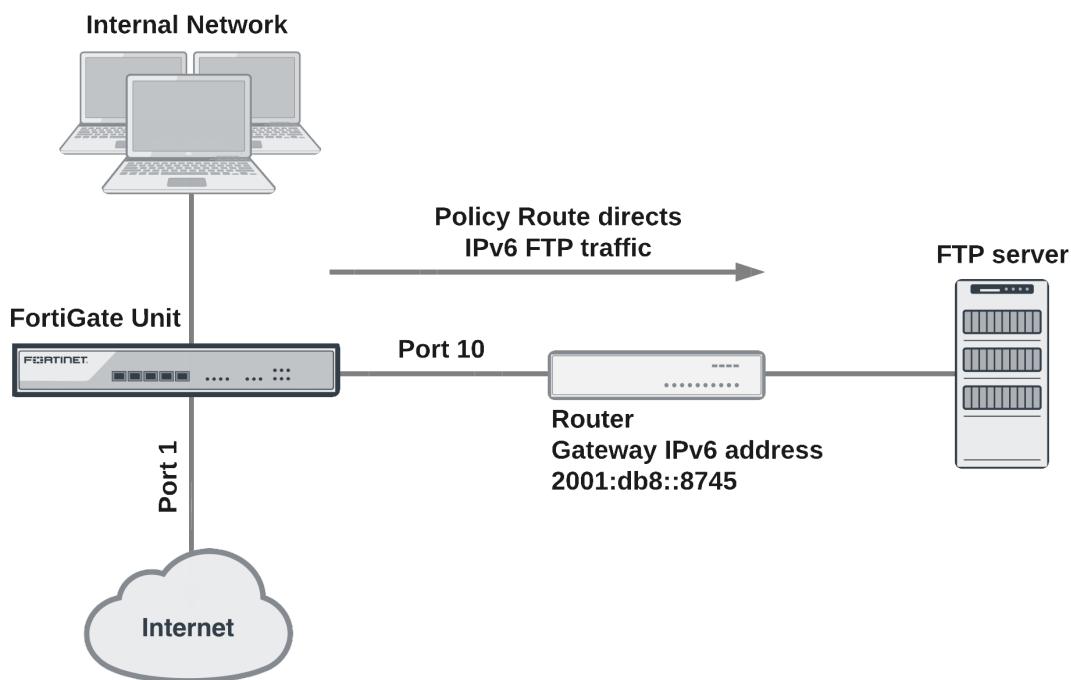
The following IPv6 features are available from the FortiOS web manager:

IPv6 policies

IPv6 security policies are created both for an IPv6 network and a transitional network. A transitional network is a network that is transitioning over to IPv6 but must still have access to the Internet or must connect over an IPv4 network.

These policies allow for this specific type of traffic to travel between the IPv6 and IPv4 networks. The IPv6 options for creating these policies is hidden by default. You must enable this feature under **System > Config > Features**.

IPv6 policy route



IPv6 policy routing

IPv6 policy routing functions in the same was as IPv4 policy routing. To add an IPv6 policy route, go to **Network > Policy Routes** and select **Create New > IPv6 Policy Route**.

Adding an IPv6 Policy route

New Routing Policy

If incoming traffic matches:

Protocol	TCP	UDP	SCTP	ANY	Specify	0
Incoming interface	Ednet (SSID: Student-net)					
Source address / mask	::/0					
Destination address / mask	::/0					
Type of Service	Bit Pattern	0x00	Bit Mask	0x00		

Then:

Outgoing interface	Click to set...
Gateway Address	::
Comments	0/255

OK **Cancel**

You can also use the following command to add IPv6 policy routes:

```
config router policy6
  edit 0
    set input-device <interface>
    set src <ipv6_ip>
    set dst <ipv6_ip>
    set protocol <0-255>
    set gateway <ipv6_ip>
    set output-device <interface>
    set tos <bit_pattern>
    set tos-mask <bit_mask>
  end
```

IPv6 security policies

IPv6 security policies support all the features supported by IPv4 security policies:

- Policy types and subtypes.
- NAT support including using the destination interface IP address, fixed port, and dynamic IP pools.
- All security features (antivirus, web filtering, application control, IPS, email filtering, DLP, VoIP, and ICAP).
- All traffic shaping options, including: shared traffic shaping, reverse shared traffic shaping, and per-IP traffic shaping.
- All user and device authentication options.

IPv6 explicit web proxy

You can use the explicit web proxy for IPv6 traffic. To do this you need to:

- Enable the IPv6 explicit web proxy from the CLI.
- Enable the explicit web proxy for one or more FortiGate interfaces. These interfaces also need IPv6 addresses.
- Add IPv6 web proxy security policies to allow the explicit web proxy to accept IPv6 traffic.

Use the following steps to set up a FortiGate unit to accept IPv6 traffic for the explicit web proxy at the Internal interface and forward IPv6 explicit proxy traffic out the wan1 interface to the Internet.

1. Enter the following CLI command to enable the IPv6 explicit web proxy:

```
config web-proxy explicit
  set status enable
  set ipv6-status enable
end
```

2. Go to **Network > Interfaces** and edit the **internal** interface, select **Enable Explicit Web Proxy** and select **OK**.
3. Go to **Policy & Objects > Proxy Policy** and select **Create New** to add an IPv6 explicit web proxy security policy with the following settings shown.

This IPv6 explicit web proxy policy allows traffic from all IPv6 IP addresses to connect through the explicit web proxy and through the wan1 interface to any IPv6 addresses that are accessible from the wan1 interface.



If you have enabled both the IPv4 and the IPv6 explicit web proxy, you can combine IPv4 and IPv6 addresses in a single explicit web proxy policy to allow both IPv4 and IPv6 traffic through the proxy.

Example IPv6 Explicit Web Proxy security policy

New Policy

Explicit Proxy Type Enabled On Source Address Outgoing Interface Destination Address Schedule Action	<div style="display: flex; justify-content: space-between;"> <input checked="" type="radio"/> Web <input type="radio"/> FTP </div> <div style="display: flex; justify-content: space-between;"> internal ? </div> <div style="display: flex; justify-content: space-between;"> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 10px;"> all </div> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 10px;"> wan1 </div> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 10px;"> all </div> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 10px;"> always </div> <div style="border: 1px solid #ccc; padding: 2px;"> ACCEPT </div> </div>
---	--

Restricting the IP address of the explicit IPv6 web proxy

You can use the following command to restrict access to the IPv6 explicit web proxy using only one IPv6 address. The IPv6 address that you specify must be the IPv6 address of an interface that the explicit HTTP proxy is enabled on. You might want to use this option if the explicit web proxy is enabled on an interface with multiple IPv6 addresses.

For example, to require users to connect to the IPv6 address 2001:db8:0:2::30 to connect to the explicit IPv6 HTTP proxy, use the following command:

```
config web-proxy explicit
  set incoming-ipv6 2001:db8:0:2::30
```

```
end
```

Restricting the outgoing source IP address of the IPv6 explicit web proxy

You can use the following command to restrict the source address of outgoing web proxy packets to a single IPv6 address. The IP address that you specify must be the IPv6 address of an interface that the explicit HTTP proxy is enabled on. You might want to use this option if the explicit HTTP proxy is enabled on an interface with multiple IPv6 addresses.

For example, to restrict the outgoing packet source address to 2001:db8:0:2::50:

```
config http-proxy explicit
  set outgoing-ip6 2001:db8:0:2::50
end
```

VIP64

VIP64 policies can be used to configure static NAT virtual IPv6 address for IPv4 addresses. VIP64 can be configured from the CLI using the following commands:

```
config firewall vip64
  edit <zname_str>
    set arp-reply {enable | disable}
    set color <color_int>
    set comment <comment_str>
    set extip <address_ipv6>[-address_ipv6]
    set extport <port_int>
    set id <id_num_str>
    set mappedip [<start_ipv4>-<end_ipv4>]
    set mappedport <port_int>
    set portforward {enable | disable}
    set src-filter <addr_str>
  end
```

VIP64 CLI Variables and Defaults

Variable	Description	Default
<zname_str>	Enter the name of this virtual IP address.	No default.
arp-reply {enable disable}	Select to respond to ARP requests for this virtual IP address.	enable
color <color_int>	Enter the number of the color to use for the group icon in the web-based manager.	0
comment <comment_str>	Enter comments relevant to the configured virtual IP.	No default.

Variable	Description	Default
<code>extip <address_ipv6>[-address_ipv6]</code>	<p>Enter the IP address or address range on the external interface that you want to map to an address or address range on the destination network.</p> <p>If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as the first IP address in the external IP address range, and calculates the last IP address required to create an equal number of external and mapped IP addresses for one-to-one mapping.</p> <p>To configure a dynamic virtual IP that accepts connections destined for any IP address, set <code>extip</code> to <code>::</code>.</p>	<code>::</code>
<code>extport <port_int></code>	<p>Enter the external port number that you want to map to a port number on the destination network.</p> <p>This option only appears if <code>portforward</code> is enabled.</p> <p>If <code>portforward</code> is enabled and you want to configure a static NAT virtual IP that maps a range of external port numbers to a range of destination port numbers, set <code>extport</code> to the first port number in the range. Then set <code>mappedport</code> to the start and end of the destination port range. The FortiGate unit automatically calculates the end of the <code>extport</code> port number range.</p>	0
<code>id <id_num_str></code>	Enter a unique identification number for the configured virtual IP. Not checked for uniqueness. Range 0 - 65535.	No default.

Variable	Description	Default
<code>mappedip [<start_ipv4>-<end_ipv4>]</code>	<p>Enter the IP address or IP address range on the destination network to which the external IP address is mapped.</p> <p>If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as the first IP address in the external IP address range, and calculates the last IP address required to create an equal number of external and mapped IP addresses for one-to-one mapping.</p> <p>If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as a single IP address to create a one-to-many mapping.</p>	0.0.0.0
<code>mappedport <port_int></code>	<p>Enter the port number on the destination network to which the external port number is mapped.</p> <p>You can also enter a port number range to forward packets to multiple ports on the destination network.</p> <p>For a static NAT virtual IP, if you add a map to port range the FortiGate unit calculates the external port number range.</p>	0
<code>portforward {enable disable}</code>	Select to enable port forwarding. You must also specify the port forwarding mappings by configuring <code>extport</code> and <code>mappedport</code> .	disable
<code>src-filter <addr_str></code>	Enter a source address filter. Each address must be in the form of an IPv4 subnet (x:x:x:x:x:x/n). Separate addresses with spaces.	null

VIP46 policies can be used to configure static NAT virtual IPv4 address for IPv6 addresses. VIP46 can be configured from the CLI using the following commands (see the table above for variable details):

```
config firewall vip46
  edit <name_str>
    set arp-reply {enable | disable}
    set color <color_int>
    set comment <comment_str>
    set extip <address_ipv4>[-address_ipv4]
    set extport <port_int>
```

```

set id <id_num_str>
set mappedip [<start_ipv6>-<end_ipv6>]
set mappedport <port_int>
set portforward {enable | disable}
set src-filter <add_str>
end

```

VIP46 CLI Variables and Defaults

Variable	Description	Default
<name_str>	Enter the name of this virtual IP address.	No default.
arp-reply {enable disable}	Select to respond to ARP requests for this virtual IP address.	enable
color <color_int>	Enter the number of the color to use for the group icon in the web-based manager.	0
comment <comment_str>	Enter comments relevant to the configured virtual IP.	No default.
extip <address_ipv4>[-address_ipv4]	Enter the IP address or address range on the external interface that you want to map to an address or address range on the destination network. If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as the first IP address in the external IP address range, and calculates the last IP address required to create an equal number of external and mapped IP addresses for one-to-one mapping. To configure a dynamic virtual IP that accepts connections destined for any IP address, set <code>extip</code> to 0.0.0.0.	0.0.0.0

Variable	Description	Default
<code>extport <port_int></code>	<p>Enter the external port number that you want to map to a port number on the destination network.</p> <p>This option only appears if <code>portforward</code> is enabled.</p> <p>If <code>portforward</code> is enabled and you want to configure a static NAT virtual IP that maps a range of external port numbers to a range of destination port numbers, set <code>extport</code> to the first port number in the range. Then set <code>mappedport</code> to the start and end of the destination port range. The FortiGate unit automatically calculates the end of the <code>extport</code> port number range.</p>	0
<code>id <id_num_str></code>	Enter a unique identification number for the configured virtual IP. Not checked for uniqueness. Range 0 - 65535.	No default.
<code>mappedip [<start_ipv6>-<end_ipv6>]</code>	<p>Enter the IP address or IP address range on the destination network to which the external IP address is mapped.</p> <p>If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as the first IP address in the external IP address range, and calculates the last IP address required to create an equal number of external and mapped IP addresses for one-to-one mapping.</p> <p>If <code>mappedip</code> is an IP address range, the FortiGate unit uses <code>extip</code> as a single IP address to create a one-to-many mapping.</p>	::

Variable	Description	Default
mappedport <port_int>	<p>Enter the port number on the destination network to which the external port number is mapped.</p> <p>You can also enter a port number range to forward packets to multiple ports on the destination network.</p> <p>For a static NAT virtual IP, if you add a map to port range the FortiGate unit calculates the external port number range.</p>	0
portforward {enable disable}	Select to enable port forwarding. You must also specify the port forwarding mappings by configuring extport and mappedport.	disable
src-filter <addr_str>	Enter a source address filter. Each address must be in the form of an IPv4 subnet (x.x.x.x/n). Separate addresses with spaces.	null

IPv6 network address translation

NAT66, NAT64, and DNS64 are now supported for IPv6. These options provide IPv6 NAT and DNS capabilities with IPv6-IPv4 tunneling or dual stack configurations. The commands are available only in the CLI.

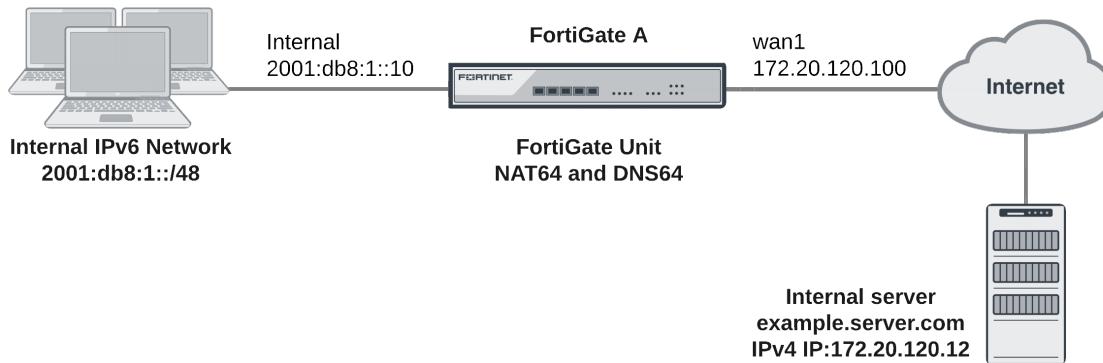
Fortinet supports all features described in [RFC 6146](#). However, for DNS64 there is no support for handling Domain Name System Security Extensions (DNSSEC). DNSSEC is for securing types of information that are provided by the DNS as used on an IP network or networks. You can find more information about DNS64 in [RFC 6147](#).

NAT64 and DNS64 (DNS proxy)

NAT64 is used to translate IPv6 addresses to IPv4 addresses so that a client on an IPv6 network can communicate transparently with a server on an IPv4 network.

NAT64 is usually implemented in combination with the DNS proxy called DNS64. DNS64 synthesizes AAAA records from A records and is used to synthesize IPv6 addresses for hosts that only have IPv4 addresses. 'DNS proxy' and 'DNS64' are interchangeable terms.

Example NAT64 configuration



With a NAT64 and DNS64 configuration in place on a FortiGate unit, clients on an IPv6 network can transparently connect to addresses on an IPv4 network. NAT64 and DNS64 perform the IPv4 to IPv6 transition, allowing clients that have already switched to IPv6 addresses to continue communicating with servers that still use IPv4 addresses.

To enable NAT64 and DNS64, use the following CLI commands:

Enable NAT64

```
config system nat64
    set status enable
end
```

Enable the DNS proxy on the IPv6 interface

```
config system dns-server
    edit internal
end
```

In your DHCP6 configuration, configure the IPv6 interface IP address as the DNS6 server IP address. The FortiGate will proxy DNS requests to the system DNS server.

```
config system dhcp6 server
    edit 1
        set interface internal
        config ip-range
            edit 1
                set start-ip 2001:db8:1::11
                set end-ip 2001:db8:1::20
            end
            set dns-server1 2001:db8:1::10
        end
    end
```

NAT64 policies

You can configure security policies for NAT64 using the web-based manager. For these options to appear, the feature must be enabled using **System > Feature Visibility**. You can then configure the policies under **Policy & Objects > NAT64 Policy**.

NAT64 policies and can also be configured from the CLI using the following command:

```
config firewall policy64
```

In the following section, you will configure a NAT64 policy that allows connections from an internal IPv6 network to an external IPv4 network.

Configuring NAT64 to allow a host on the IPv6 network to connect to the Internet server

In this example, the Internal IPv6 network address is 2001:db8:1::/48 and the external IPv4 network address is 172.20.120.0/24. NAT64 is configured to allow a user on the internal network to connect to the server at IPv4 address 172.20.120.12. In this configuration, sessions exiting the wan1 interface must have their source address changed to an IPv4 address in the range 172.20.120.200 to 172.20.120.210.

Enter the following command to enable NAT64:

```
config system nat64
    set status enable
end
```

Enabling NAT64 with the `config system nat64` command means that all IPv6 traffic received by the current VDOM can be subject to NAT64 if the source and destination address matches an NAT64 security policy.

By default, the setting `always-synthesize-aaaa-record` is enabled. If you disable this setting, the DNS proxy (DNS64) will attempt to find an AAAA records for queries to domain names and therefore resolve the host names to IPv6 addresses. If the DNS proxy cannot find an AAAA record, it synthesizes one by adding the NAT64 prefix to the A record.

By using the `nat64-prefix` option of the `config system nat64` command to change the default `nat64` prefix from the well-known prefix of `64:ff9b::/96` and setting `always-synthesize-aaaa-record` to enable (default), the DNS proxy does not check for AAAA records but rather synthesizes AAAA records.

As an alternative to the above entry, there is the optional configuration that would allow the resolution of CNAME queries.

```
config system nat64
    set status enable
    set nat64-prefix 64:ff9b::/96
    set always-synthesize-aaaa-record enable
end
```

Enter the following command to add an IPv6 firewall address for the internal network:

```
config firewall address6
    edit internal-net6
        set ip6 2001:db8:1::/48
    end
```

Enter the following command to add an IPv4 firewall address for the external network:

```
config firewall address
    edit external-net4
        set subnet 172.20.120.0/24
        set associated-interface wan1
    end
```

Enter the following command to add an IP pool containing the IPv4 address that should become the source address of the packets exiting the wan1 interface:

```
config firewall ippool
```

```
edit exit-pool4
  set startip 172.20.120.200
  set endip 172.20.120.210
end
```

Enter the following command to add a NAT64 policy that allows connections from the internal IPv6 network to the external IPv4 network:

```
config firewall policy64
  edit 0
    set srcintf internal
    set srcaddr internal-net6
    set dstintf wan1
    set dstaddr external-net4
    set action accept
    set schedule always
    set service ALL
    set logtraffic enable
    set ippool enable
    set poolname exit-pool4
  end
```

The `srcaddr` can be any IPv6 firewall address and the `dstaddr` can be any IPv4 firewall address.

Other NAT64 policy options include `fixedport`, which can be used to prevent NAT64 from changing the destination port. You can also configure traffic shaping for NAT64 policies.

How a host on the internal IPv6 network communicates with example.server.com that only has IPv4 address on the Internet

1. The host on the internal network does a DNS lookup for example.server.com by sending a DNS query for an AAAA record for example.server.com.
2. The DNS query is intercepted by the FortiGate DNS proxy.
3. The DNS proxy attempts to resolve the query with a DNS server on the Internet and discovers that there are no AAAA records for example.server.com.
4. The previous step is skipped if `always-synthesize-aaaa-record` is enabled.
5. The DNS proxy performs an A-record query for example.server.com and gets back an RRSet containing a single A record with the IPv4 address 172.20.120.12.
6. The DNS proxy then synthesizes an AAAA record. The IPv6 address in the AAAA record begins with the configured NAT64 prefix in the upper 96 bits and the received IPv4 address in the lower 32 bits. By default, the resulting IPv6 address is 64:ff9b::172.20.120.12.
7. The host on the internal network receives the synthetic AAAA record and sends a packet to the destination address 64:ff9b::172.20.120.12.
8. The packet is routed to the FortiGate internal interface where it is accepted by the NAT64 security policy.
9. The FortiGate unit translates the destination address of the packets from IPv6 address 64:ff9b::172.20.120.12 to IPv4 address 172.20.120.12 and translates the source address of the packets to 172.20.120.200 (or another address in the IP pool range) and forwards the packets out the wan1 interface to the Internet.

NAT66

NAT66 is used for translating an IPv6 source or destination address to a different IPv6 source or destination address. NAT66 is not as common or as important as IPv4 NAT, as many IPv6 addresses do not need NAT66 as

much as IPv4 NAT. However, NAT66 can be useful for a number of reasons. For example, you may have changed the IP addresses of some devices on your network but want traffic to still appear to be coming from their old addresses. You can use NAT66 to translate the source addresses of packets from the devices to their old source addresses.

In FortiOS, NAT66 options can be added to an IPv6 security policy from the CLI. Configuring NAT66 is very similar to configuring NAT in an IPv4 security policy. For example, use the following command to add an IPv6 security policy that translates the source address of IPv6 packets to the address of the destination interface (similar to IPv4 source NAT):

```
config firewall policy6
  edit 0
    set srcintf internal
    set dstintf wan1
    set srcaddr internal_net
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set nat enable
  end
```

It's also can be useful to translate one IPv6 source address to another address that is not the same as the address of the exiting interface. You can do this using IP pools. For example, enter the following command to add an IPv6 IP pool containing one IPv6 IP address:

```
config firewall ippool6
  edit example_6_pool
    set startip 2001:db8::
    set endip 2001:db8::
  end
```

Enter the following command to add an IPv6 firewall address that contains a single IPv6 IP address.

```
config firewall address6
  edit device_address
    set ip6 2001:db8::132/128
  end
```

Enter the following command to add an IPv6 security policy that accepts packets from a device with IP address 2001:db8::132 and translates the source address to 2001:db8::.

```
config firewall policy6
  edit 0
    set srcintf internal
    set dstintf wan1
    set srcaddr device_address
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set nat enable
    set ippool enable
    set poolname example_6_pool
  end
```

NAT66 destination address translation

NAT66 can also be used to translate destination addresses. This is done in an IPv6 policy by using IPv6 virtual IPs. For example, enter the following command to add an IPv6 virtual IP that maps the destination address 2001:db8::dd to 2001:db8::ee.

```
config firewall vip6
  edit example-vip6
    set extip 2001:db8::dd
    set mappedip 2001:db8::ee
  end
```

Enter the following command to add an IPv6 security policy that accepts packets with a destination address 2001:db8::dd and translates that destination address to 2001:db8::ee.

```
config firewall policy6
  edit 0
    set srcintf internal
    set dstintf wan1
    set srcaddr all
    set dstaddr example-vip6
    set action accept
    set schedule always
    set service ALL
  end
```

NAT64 and NAT66 session failover

The FortiGate Clustering Protocol (FGCP) supports IPv6, NAT64, and NAT66 session failover. If session pickup is enabled, these sessions are synchronized between cluster members and, after an HA failover, the sessions will resume with only minimal interruption.

NAT46

NAT46 is used to translate IPv4 addresses to IPv6 addresses so that a client on an IPv4 network can communicate transparently with a server on an IPv6 network.

To enable NAT46, use the following CLI command:

```
config firewall vip46
```

NAT46 policies

Security policies for NAT46 can be configured from the web-based manager. For these options to appear in the web-based manager, this feature must be enabled using **System > Feature Visibility**. You can then configure the policies under **Policy & Objects > NAT46 Policy**.

NAT46 policies and can also be configured from the CLI using the following command:

```
config firewall policy46
```

IPv6 tunneling

IPv6 Tunneling is the act of tunneling IPv6 packets from an IPv6 network through an IPv4 network to another IPv6 network. This is different than Network Address Translation (NAT) because once the packet reaches its final destination the true originating address of the sender will still be readable. The IPv6 packets are encapsulated

within packets with IPv4 headers, which carry their IPv6 payload through the IPv4 network. This type of configuration is more appropriate for those who have completely transitional over to IPv6, but need an Internet connection, which is still mostly IPv4 addresses.

The key to IPv6 tunneling is the ability of the 2 devices, whether they are a host or a network device, to be dual stack compatible. They have to be able to work with both IPv4 and IPv6 at the same time. In the process the entry node of the tunnel portion of the path will create an encapsulating IPv4 header and transmit the encapsulated packet. The exit node at the end of the tunnel receives the encapsulated packet. The IPv4 header is removed. The IPv6 header is updated and the IPv6 packet is processed.

There are two types of tunnels in IPv6:

Automatic tunnels	Automatic tunnels are configured by using IPv4 address information embedded in an IPv6 address – the IPv6 address of the destination host includes information about which IPv4 address the packet should be tunneled to.
Configured tunnels	Configured tunnels must be configured manually. These tunnels are used when using IPv6 addresses that do not have any embedded IPv4 information. The IPv6 and IPv4 addresses of the endpoints of the tunnel must be specified.

Tunnel configurations

There are a few ways in which the tunneling can be performed depending on which segment of the path between the end points of the session the encapsulation takes place.

Network Device to Network Device	Dual stack capable devices connected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans one segment of the path taken by the IPv6 packets.
Host to Network Device	Dual stack capable hosts can tunnel IPv6 packets to an intermediary IPv6 or IPv4 network device that is reachable through an IPv4 infrastructure. This type of tunnel spans the first segment of the path taken by the IPv6 packets.
Host to Host	Dual stack capable hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans the entire path taken by the IPv6 packets.
Network Device to Host	Dual stack capable network devices can tunnel IPv6 packets to their final destination IPv6 or IPv4 host. This tunnel spans only the last segment of the path taken by the IPv6 packets.

Regardless of whether the tunnel starts at a host or a network device, the node that does the encapsulation needs to maintain soft state information, such as the maximum transmission unit (MTU), about each tunnel in order to process the IPv6 packets.

Use the following command to tunnel IPv6 traffic over an IPv4 network. The IPv6 interface is configured under config system interface. The command to do the reverse is config system ipv6-tunnel. These commands are not available in transparent mode.

```
config system sit-tunnel
  edit <tunnel name>
    set destination <tunnel _address>
    set interface <name>
```

```

set ip6 <address_ipv6>
set source <address_ipv4>
end

```

Variable	Description	Default
edit <tunnel_name>	Enter a name for the IPv6 tunnel.	No default.
destination <tunnel_address>	The destination IPv4 address for this tunnel.	0.0.0.0
interface <name>	The interface used to send and receive traffic for this tunnel.	No default.
ip6 <address_ipv6>	The IPv6 address for this tunnel.	No default.
source <address_ipv4>	The source IPv4 address for this tunnel.	0.0.0.0

Tunneling IPv6 through IPsec VPN

A variation on the tunneling IPv6 through IPv4 is using an IPsec VPN tunnel between to FortiGate devices. FortiOS supports IPv6 over IPsec. In this sort of scenario, 2 networks using IPv6 behind FortiGate units are separated by the Internet, which uses IPv4. An IPsec VPN tunnel is created between the 2 FortiGate units and a tunnel is created over the IPv4 based Internet but the traffic in the tunnel is IPv6. This has the additional advantage of make the traffic secure as well.

For configuration information, see [IPv6 IPsec VPN on page 1](#).

IPv6 support for GRE tunnels

You can use IPv6 addresses can be used at both ends of a GRE tunnel in the same way as with IPv4.

The configuration is similar to how you set up the tunnel for IPv4. However, when you configure the specific tunnel, you need to set the `ip-version` option to 6. This will enable IPv6-specific options for the tunnel.

CLI

```

config system gre-tunnel
  edit <name of tunnel>
    set ip-version 6
    set remote-gw6 <IPv6 address of the remote gateway>
    set local-gw-6 <IPv6 address of the local gateway>
  end

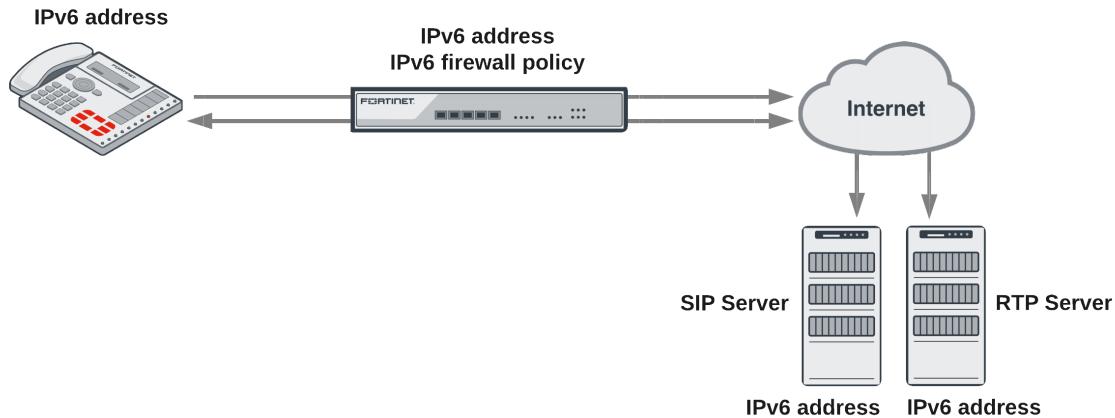
```

SIP over IPv6

FortiOS supports Sessions Initiate Protocol (SIP) over IPv6. The SIP application-level gateway (ALG) can process SIP messages that use IPv6 addresses in the headers, bodies, and in the transport stack. The SIP ALG cannot modify the IPv6 addresses in the SIP headers so FortiGate units cannot perform SIP or RTP NAT over IPv6 and also cannot translate between IPv6 and IPv4 addresses.

In the scenario shown below, a SIP phone connects to the Internet through a FortiGate unit operating. The phone and the SIP and RTP servers all have IPv6 addresses.

The FortiGate unit has IPv6 security policies that accept SIP sessions. The SIP ALG understands IPv6 addresses and can forward IPv6 sessions to their destinations. Using SIP application control features the SIP ALG can also apply rate limiting and other settings to SIP sessions.



To enable SIP support for IPv6 add an IPv6 security policy that accepts SIP packets and includes a VoIP profile.

New Fortinet FortiGate IPv6 MIB fields

The following IPv6 MIB fields have been added to the Fortinet FortiGate MIB. These MIB entries can be used to display IPv6 session and policy statistics.

- **IPv6 Session Counters:**

```
fgSysSes6Count
fgSysSes6Rate1
fgSysSes6Rate10
fgSysSes6Rate30
fgSysSes6Rate60
```

- **IPv6 Policy Statistics:**

```
fgFwPol6StatsTable
fgFwPol6StatsEntry
FgFwPol6StatsEntry
fgFwPol6ID
fgFwPol6PktCount
fgFwPol6ByteCount
```

- **IPv6 Session Statistics:**

```
fgIp6SessStatsTable
fgIp6SessStatsEntry
FgIp6SessStatsEntry
fgIp6SessNumber
```

The fgSysSesCount and fgSysSesRateX MIBs report statistics for IPv4 plus IPv6 sessions combined. This behavior was not changed.

New OIDs

The following OIDs have been added:

```

FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgSystem.fgSystemInfo
    .fgSysSes6Count    1.3.6.1.4.1.12356.101.4.1.15
    .fgSysSesRate1     1.3.6.1.4.1.12356.101.4.1.16
    .fgSysSesRate10    1.3.6.1.4.1.12356.101.4.1.17
    .fgSysSesRate30    1.3.6.1.4.1.12356.101.4.1.18
    .fgSysSesRate60    1.3.6.1.4.1.12356.101.4.1.19

FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgFirewall.fgFwPolicies.fgFwPolTables
    .fgFwPol6StatsTable.fgFwPol6StatsEntry.fgFwPol6ID    1.3.6.1.4.1.12356.101.5.1.2.2.1.1
    .fgFwPol6StatsTable.fgFwPol6StatsEntry.fgFwPol6PktCount    1.3.6.1.4.1.12356.101.5.1.2.2.1.
    .2
    .fgFwPol6StatsTable.fgFwPol6StatsEntry.fgFwPol6ByteCount    1.3.6.1.4.1.12356.101.5.1.2.2.1
    .3

FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgInetProto.fgInetProtoTables
    .fgIp6SessStatsTable.fgIp6SessStatsEntry.fgIp6SessNumber 1.3.6.1.4.1.12356.101.11.2.3.1.1

```

EXAMPLE SNMP get/walk output

```

// Session6 stats excerpt from sysinfo:
snmpwalk -v2c -cpublic 192.168.1.111 1.3.6.1.4.1.12356.101.4
FORTINET-FORTIGATE-MIB::fgSysSes6Count.0 = Gauge32: 203
FORTINET-FORTIGATE-MIB::fgSysSes6Rate1.0 = Gauge32: 10 Sessions Per Second
FORTINET-FORTIGATE-MIB::fgSysSes6Rate10.0 = Gauge32: 2 Sessions Per Second
FORTINET-FORTIGATE-MIB::fgSysSes6Rate30.0 = Gauge32: 1 Sessions Per Second
FORTINET-FORTIGATE-MIB::fgSysSes6Rate60.0 = Gauge32: 0 Sessions Per Second

// FwPolicy6 table:
snmpwalk -v2c -cpublic 192.168.1.111 1.3.6.1.4.1.12356.101.5.1.2.2
FORTINET-FORTIGATE-MIB::fgFwPol6ID.1.3 = INTEGER: 3
FORTINET-FORTIGATE-MIB::fgFwPol6ID.1.4 = INTEGER: 4
FORTINET-FORTIGATE-MIB::fgFwPol6PktCount.1.3 = Counter64: 4329
FORTINET-FORTIGATE-MIB::fgFwPol6PktCount.1.4 = Counter64: 0
FORTINET-FORTIGATE-MIB::fgFwPol6ByteCount.1.3 = Counter64: 317776
FORTINET-FORTIGATE-MIB::fgFwPol6ByteCount.1.4 = Counter64: 0

// IP6SessNumber:
snmpwalk -v2c -cpublic 192.168.1.111 1.3.6.1.4.1.12356.101.11.2.3.1
FORTINET-FORTIGATE-MIB::fgIp6SessNumber.1 = Counter32: 89

```

IPv6 per-IP traffic shaper

You can add any Per-IP traffic shaper to an IPv6 security policy using the following command:

```

config firewall policy6
    edit 0
        set per-ip-shaper "new-perip-shaper"
    end

```

DHCPv6

You can use DHCP with IPv6 using the CLI. To configure DHCP, ensure IPv6 is enabled by going to **System > Feature Visibility** and enabling **IPv6**.

Use the CLI command

```
config system dhcp6
```

For more information on the configuration options, see the [FortiGate CLI Reference](#).

DHCP delegated mode

Downstream IPv6 interfaces can receive address assignments on delegated subnets from a DHCP server that serves an upstream interface.

DHCPv6-PD configuration

Enable DHCPv6 Prefix Delegation on upstream interface (port10):

```
config system interface
  edit "port10"
    config ipv6
      set dhcp6-prefix-delegation enable
    end
  end
```

Assign delegated prefix on downstream interface (port1). Optionally, specific delegated prefixes can be specified:

```
config system interface
  edit "port1"
    config ipv6
      set ip6-mode delegated
      set ip6-upstream-interface "port10"
      set ip6-subnet ::1:0:0:0:1/64
      set ip6-send-adv enable
      config ipv6-delegated-prefix-list
        edit 1
          set upstream-interface "port10"
          set autonomous-flag enable
          set onlink-flag enable
          set subnet 0:0:0:100::/64
        end
      end
    end
```

DHCPv6 server configuration

Configuring a server that uses delegated prefix and DNS from upstream:

```
config system dhcp6 server
  edit 1
    set dns-service delegated
```

```
set interface "wan2"
set upstream-interface "wan1"
set ip-mode delegated
set subnet 0:0:0:102::/64
end
```

DHCPv6 relay

You can use the following command to configure a FortiGate interface to relay DHCPv6 queries and responses from one network to a network with a DHCPv6 server and back. The command enables DHCPv6 relay and includes adding the IPv6 address of the DHCP server that the FortiGate unit relays DHCPv6 requests to:

```
config system interface
  edit internal
    config ipv6
      set dhcp6-relay-service enable
      set dhcp6-relay-type regular
      set dhcp6-relay-ip 2001:db8:0:2::30
    end
  end
```

IPv6 forwarding

Policies, IPS, Application Control, flow-based antivirus, web filtering, and DLP

FortiOS fully supports flow-based inspection of IPv6 traffic. This includes full support for IPS, application control, virus scanning, and web filtering.

To add flow-based inspection to IPv6 traffic go to **Policy & Objects > IPv6 Policy** and select **Create New** to add an IPv6 Security Policy. Configure the policy to accept the traffic to be scanned. Under **Security Profiles**, select the profiles to apply to the traffic.

Obtaining IPv6 addresses from an IPv6 DHCP server

From the CLI, you can configure any FortiGate interface to get an IPv6 address from an IPv6 DHCP server. For example, to configure the wan2 interface to get an IPv6 address from an IPv6 DHCP server enter the following command:

```
config system interface
  edit wan2
    config ipv6
      set ip6-mode dhcp
    end
  end
```

Authentication support

RADIUS

FortiOS supports IPv6 RADIUS authentication. When configuring the FortiGate interface and the RADIUS server (under `config system interface` and `config user radius` respectively), the server IP address can be set as IPv6.

Captive portal

Captive portal supports IPv6. It works with remote RADIUS authentication and WiFi interfaces.

FSSO

FortiGate FSSO supports connecting to an FSSO agent over IPv6 and collecting and sending IPv6 details about endpoints. This is all enforced the same way as IPv4 FSSO traffic.

CLI

```
config user fss
  edit <fss agent name>
    set source-ip6 <IPv6 address for source>
  end
```



The source-ip6 option accepts either an IPv6 format address or a Fully Qualified Domain Name (FQDN). The FQDN can parse to an IPv6 address.

IPv6 Neighbor Discovery Proxy

This feature provides support for proxying the IPv6 Neighbor Discovery (ND) protocol to allow the forwarding of the following ICMP messages between upstream and downstream interfaces:

- Router Advertisement (RA)
- Neighbor Solicitation (NS)
- Neighbor Advertisement (NA)
- Router Solicitation (RS)
- Redirect



Normally, only one interface will receive RA traffic. This will automatically be considered the upstream interface.

The Neighbor Discovery (ND) protocol is used to discover the Link Layer address of IPv6 destinations. In IPv4, this is achieved by using ARP.

Configure ND Proxy in the CLI using the following syntax:

```
config system nd-proxy
  set status {enable|disable}
  set member <interface> <interface> [<interface>...]
end
```

Option	Description
status	Enable/disable the use of neighbor discovery proxy
member	List of interfaces using the neighbor discovery proxy

An example of a configuration can be found in the IPv6 Configuration section under [IPv6 Neighbor Discovery Proxy on page 164](#)



More information on Neighbor Discovery Proxies (ND Proxy) is available at [RFC 4389](#).

IPv6 configuration

This section contains configuration information for IPv6 on FortiOS. Attempts are made to include scenarios in each section to better assist with the configuration and to orient the information toward a particular task.

You will find information on the following:



By default IPv6 configurations do not appear in the web-based manager. You need to enable the feature first.

To enable IPv6:

1. Go to **System > Features**.
2. Select **IPv6** and click **Apply**.

IPv6 address groups

To create IPv6 address groups from existing IPv6 addresses - web-based manager

Your company has 3 internal servers with IPv6 addresses that it would like to group together for the purposes of a number of policies.

1. Go to **Policy & Objects > Addresses** and select **Create New > Address Group**.
2. Select **IPv6 Group**, and fill out the fields with the following information:

Group Name	Web_Server_Cluster
Members	Web_Server-1
	Web_Server-2
	Web_Server-3

3. Select **OK**.

To create IPv6 address groups from existing IPv6 addresses - CLI

```
config firewall addrgrp6
  edit Web_Server_Cluster
    set member Web_Server-1 Web_Server-2 Web_Server-3
  end
```

To verify that the addresses were added correctly

1. Go to **Policy & Objects > Addresses**. Check that the addresses have been added to the address list and that they are correct.
2. From the CLI, enter the following commands:

```
config firewall addgrp6
  edit <the name of the address that you wish to verify>
    Show full-configuration
```

IPv6 address ranges

You can configure IPv6 address ranges in both the GUI and the CLI.

To configure IPv6 address ranges - web-based manager:

1. Go to **Policy & Objects > Addresses**.
2. Set the **Type** to **IP Range** and enter the IPv6 addresses as shown:

Edit Address		
Category	<input type="radio"/> Address <input checked="" type="radio"/> IPv6 Address <input type="radio"/> Multicast Address	
Name	ipv6range	
Type	IP Range	
Subnet / IP Range	2001:db8:0:2::30-2001:db8:0:2::31	
Visibility	<input checked="" type="checkbox"/>	
Comments	Write a comment... 0/255	

OK Cancel

To configure IPv6 address ranges - CLI:

```
config firewall address6
  edit ipv6range
    set type iprange
    set start-ip 2001:db8:0:2::30
    set end-ip 2001:db8:0:2::31
  end
```

IPv6 firewall addresses

Scenario: Mail server

You need to create an IPv6 address for the Mail Server on Port1 of your internal network. This server is on the network off of port1.

- The IP address is 2001:db8:0:2::20/128
- There should be a tag for this address being for a server.

Configuring the Example using the GUI

1. Go to **Policy & Objects > Objects > Addresses** and select **Create New > Address**.
2. Select **IPv6 Address** and fill out the fields with the following information

Name	Mail_Server
Type	Subnet
IPv6 Address	2001:db8:0:2::20/128

3. Select **OK**.

Configuring the Example using the CLI

Enter the following CLI command:

```
config firewall address6
  edit Mail_Server
    set type ipprefix
    set subnet 2001:db8:0:2::20/128
  end
```

Scenario: First floor network

You need to create an IPv6 address for the subnet of the internal network off of Port1. These computers connect to port1. The network uses the IPv6 addresses: fdde:5a7d:f40b:2e9d:xxxx:xxxx:xxxx:xxxx

There should be a reference to this being the network for the 1st floor of the building.

1. Go to **Policy & Objects > Objects > Addresses**
2. Select **Create New > Address**. Select **IPv6 Address** and fill out the fields with the following information:

Name	Internal_Subnet_1
Type	Subnet / IP Range
IPv6 Address	2001:db8:0:2::/64
Comments	Network for 1st Floor

3. Select **OK**.
4. Enter the following CLI command:

```
config firewall address6
  edit Internal_Subnet_1
    set comment "Network for 1st Floor"
    set type ipprefix
    set subnet 2001:db8:0:2::/64
  end
```

Scenario: Accounting team

You need to create an IPv6 address for the Accounting Team that's on the 1st Floor. These users are off of various ports of the FortiGate, but they have all been assigned addresses between 2001:db8:0:2::2000 and 2001:db8:0:2::a000

Configuring the example using the GUI

1. Go to **Policy & Objects > Objects > Addresses** and select **Create New > Address**.
2. Select **IPv6 Address** and fill out the fields with the following information

Name	Accounting_Team
Type	IP Range
Subnet / IP Range	2001:db8:0:2::2000-2001:db8:0:2::a000

3. Select **OK**.

Configuring the Example using the CLI

Enter the following CLI command:

```
config firewall address6
  edit Accounting_Team
    set type iprange
    set visibility enable
    set start-ip 2001:db8:0:2::2000
    set end-ip 2001:db8:0:2::a000
  end
```

To verify that the addresses were added correctly:

1. Go to **Policy & Objects > Objects > Addresses**. Check that the addresses have been added to the address list and that they are correct.
2. Enter the following CLI command:

```
config firewall address6
  edit <the name of the address that you wish to verify>
    Show full-configuration
```

IPv6 SSH

FortiGate supports SSH traffic through IPv6. When the proxy option is set to `ssh` in a proxy policy, IPv6 source and destination address options become available and SSH profiles can be assigned to IPv6 firewall policies.

Syntax in IPv6 firewall policy

```
config firewall policy6
  edit 1
    set utm-status enable
    set ssh-filter-profile <example>
  end
```

Syntax in proxy policy

```
config firewall proxy-policy
  edit 1
    set proxy ssh
    set srcaddr6 "all"
    set dstaddr6 "all"
  end
```

Logging

When a proxy policy is being used, SSH traffic logs are generated by `wad` instead of the kernel.

ICMPv6

The IT Manager is doing some diagnostics and would like to temporarily block the successful replies of ICMP Node information Responses between 2 IPv6 networks.

The ICMP type for ICMP Node information responses is 140. The codes for a successful response is 0.

To configure ICMPv6 - web-based manager:

1. Go to **Policy & Objects > Services** and select **Create New > Service**.
2. Fill out the fields with the following information

Name	diagnostic-test1
Service Type	Firewall
Show in Service List	Enabled
Category	Uncategorized
Protocol Type	ICMP6
Type	140

3. Select **OK**.
4. Enter the following CLI command:

```
config firewall service custom
  edit diagnostic-test1
    set protocol ICMP6
    set icmptype 140
    set icmpcode 0
    set visibility enable
  end
```

To verify that the category was added correctly:

1. Go to **Policy & Objects > Services**. Check that the services have been added to the services list and that they are correct.
2. Enter the following CLI command:

```
config firewall service custom
  edit <the name of the service that you wish to verify>
```

```
show full-configuration
```

IPv6 IPsec VPN

This chapter describes how to configure your FortiGate unit's IPv6 IPsec VPN functionality.



By default IPv6 configurations do not appear in the web-based manager. You need to enable the feature first.

To enable IPv6:

1. Go to **System > Features**.
2. Select **IPv6** and click **Apply**.

The topics in this section include:

- Overview of IPv6 IPsec support
- Configuring IPv6 IPsec VPNs
- Site-to-site IPv6 over IPv6 VPN example
- Site-to-site IPv4 over IPv6 VPN example
- Site-to-site IPv6 over IPv4 VPN example

Overview of IPv6 IPsec support

FortiOS supports route-based IPv6 IPsec, but not policy-based. This section describes how IPv6 IPsec support differs from IPv4 IPsec support.

Where both the gateways and the protected networks use IPv6 addresses, sometimes called IPv6 over IPv6, you can create either an auto-keyed or manually-keyed VPN. You can also combine IPv6 and IPv4 addressing in an auto-keyed VPN in the following ways:

IPv4 over IPv6	The VPN gateways have IPv6 addresses. The protected networks have IPv4 addresses. The phase 2 configurations at either end use IPv4 selectors.
IPv6 over IPv4	The VPN gateways have IPv4 addresses. The protected networks use IPv6 addresses. The phase 2 configurations at either end use IPv6 selectors.

Compared with IPv4 IPsec VPN functionality, there are some limitations:

- Except for IPv6 over IPv4, remote gateways with Dynamic DNS are not supported.
- Selectors cannot be firewall address names. Only IP address, address range and subnet are supported.
- Redundant IPv6 tunnels are not supported.

Certificates

On a VPN with IPv6 phase 1 configuration, you can authenticate using VPN certificates in which the common name (cn) is an IPv6 address. The `cn-type` keyword of the `user peer` command has an option, `ipv6`, to

support this.

Configuring IPv6 IPsec VPNs

Configuration of an IPv6 IPsec VPN follows the same sequence as for an IPv4 route-based VPN: phase 1 settings, phase 2 settings, security policies, and routing.

Phase 1 configuration

In the web-based manager, you define the Phase 1 as IPv6 in the Advanced settings. Enable the IPv6 Version check box. You can then enter an IPv6 address for the remote gateway.

In the CLI, you define an IPsec phase 1 configuration as IPv6 by setting `ip-version` to 6. Its default value is 4. Then, the `local-gw` and `remote-gw` keywords are hidden and the corresponding `local-gw6` and `remote-gw6` keywords are available. The values for `local-gw6` and `remote-gw6` must be IPv6 addresses. For example:

```
config vpn ipsec phase1-interface
  edit tunnel16
    set ip-version 6
    set remote-gw6 0:123:4567::1234
    set interface port3
    set proposal 3des-md5
  end
```

Phase 2 configuration

To create an IPv6 IPsec phase 2 configuration in the web-based manager, you need to define IPv6 selectors in the Advanced settings. Change the default “0.0.0.0/0” address for Source address and Destination address to the IPv6 value “::/0”. If needed, enter specific IPv6 addresses, address ranges or subnet addresses in these fields.

In the CLI, set `src-addr-type` and `dst-addr-type` to `ip6`, `range6` or `subnet6` to specify IPv6 selectors. By default, zero selectors are entered, “::/0” for the `subnet6` address type, for example. The simplest IPv6 phase 2 configuration looks like this:

```
config vpn ipsec phase2-interface
  edit tunnel16_p2
    set phasename tunnel16
    set proposal 3des-md5
    set src-addr-type subnet6
    set dst-addr-type subnet6
  end
```

Security policies

To complete the VPN configuration, you need a security policy in each direction to permit traffic between the protected network's port and the IPsec interface. You need IPv6 policies unless the VPN is IPv4 over IPv6.

Routing

Appropriate routing is needed for both the IPsec packets and the encapsulated traffic within them. You need a route, which could be the default route, to the remote VPN gateway via the appropriate interface. You also need a route to the remote protected network via the IPsec interface.

To create a static route - web-based manager:

1. Go to **Network > Static Routes**.
2. Select the drop-down arrow on the **Create New** button and select **IPv6 Route**.
3. Enter the information and select **OK**.

To create a static route - CLI:

1. In the CLI, use the `router static6` command. For example, where the remote network is `fec0:0000:0000:0004::/64` and the IPsec interface is `toB`:

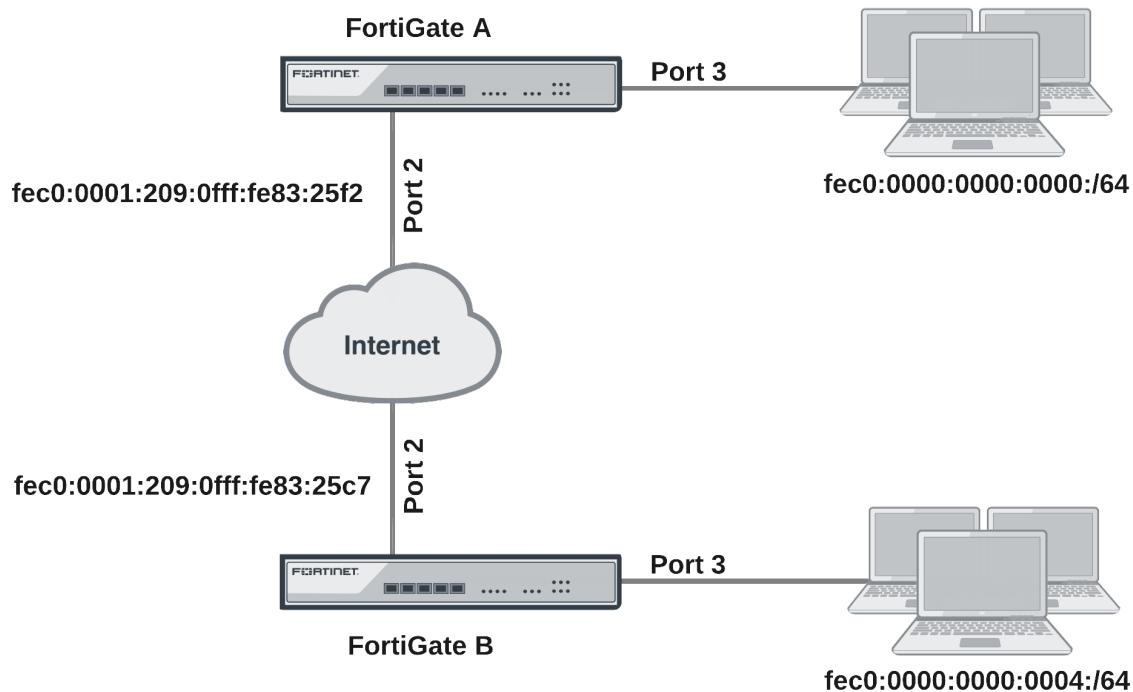
```
config router static6
  edit 1
    set device port2
    set dst 0::/0
  next
  edit 2
    set device toB
    set dst fec0:0000:0000:0004::/64
  next
end
```

If the VPN is IPv4 over IPv6, the route to the remote protected network is an IPv4 route. If the VPN is IPv6 over IPv4, the route to the remote VPN gateway is an IPv4 route.

Site-to-site IPv6 over IPv6 VPN example

In this example, computers on IPv6-addressed private networks communicate securely over public IPv6 infrastructure.

Example IPv6-over-IPv6 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the public network and port 3 connects to the local network.

```
config system interface
  edit port2
    config ipv6
      set ip6-address fec0::0001:209:0fff:fe83:25f2/64
    end
  next
  edit port3
    config ipv6
      set ip6-address fec0::0000:209:0fff:fe83:25f3/64
    end
  next
end
```

Configure FortiGate A IPsec settings

The phase 1 configuration creates a virtual IPsec interface on port 2 and sets the remote gateway to the public IP address FortiGate B. This configuration is the same as for an IPv4 route-based VPN, except that `ip-version` is set to 6 and the `remote-gw6` keyword is used to specify an IPv6 remote gateway address.

```
config vpn ipsec phasel-interface
  edit toB
    set ip-version 6
    set interface port2
```

```

    set remote-gw6 fec0:0000:0000:0003:209:0fff:fe83:25c7
    set dpd enable
    set psksecret maryhadalittlelamb
    set proposal 3des-md5 3des-sha1
end

```

By default, phase 2 selectors are set to accept all subnet addresses for source and destination. The default setting for `src-addr-type` and `dst-addr-type` is `subnet`. The IPv6 equivalent is `subnet`. The default subnet addresses are `0.0.0.0/0` for IPv4, `::/0` for IPv6.

```

config vpn ipsec phase2-interface
edit toB2
    set phase1name toB
    set proposal 3des-md5 3des-sha1
    set pfs enable
    set replay enable
    set src-addr-type subnet6
    set dst-addr-type subnet6
end

```

Configure FortiGate A security policies

Security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. The address `all6` must be defined using the `firewall address6` command as `::/0`.

```

config firewall policy6
edit 1
    set srcintf port3
    set dstintf toB
    set srcaddr all6
    set dstaddr all6
    set action accept
    set service ALL
    set schedule always
next
edit 2
    set srcintf toB
    set dstintf port3
    set srcaddr all6
    set dstaddr all6
    set action accept
    set service ALL
    set schedule always
end

```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB. A default route sends all IPv6 traffic out on port2.

```

config router static6
edit 1
    set device port2
    set dst 0::/0
next
edit 2
    set device toB
    set dst fec0:0000:0000:0004::/64
end

```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the public IP address of FortiGate A. Security policies enable traffic to pass between the private network and the IPsec interface. Routing ensures traffic for the private network behind FortiGate A goes through the VPN and that all IPv6 packets are routed to the public network.

```
config system interface
  edit port2
    config ipv6
      set ip6-address fec0::0003:209:0fff:fe83:25c7/64
    end
  next
  edit port3
    config ipv6
      set ip6-address fec0::0004:209:0fff:fe83:2569/64
    end
  end
config vpn ipsec phasel-interface
  edit toA
    set ip-version 6
    set interface port2
    set remote-gw6 fec0:0000:0000:0001:209:0fff:fe83:25f2
    set dpd enable
    set psksecret maryhadalittlelamb
    set proposal 3des-md5 3des-sha1
  end
config vpn ipsec phase2-interface
  edit toA2
    set phase1name toA
    set proposal 3des-md5 3des-sha1
    set pfs enable
    set replay enable
    set src-addr-type subnet6
    set dst-addr-type subnet6
  end
config firewall policy6
  edit 1
    set srcintf port3
    set dstintf toA
    set srcaddr all6
    set dstaddr all6
    set action accept
    set service ALL
    set schedule always
  next
  edit 2
    set srcintf toA
    set dstintf port3
    set srcaddr all6
    set dstaddr all6
    set action accept
    set service ALL
    set schedule always
  end
config router static6
  edit 1
```

```

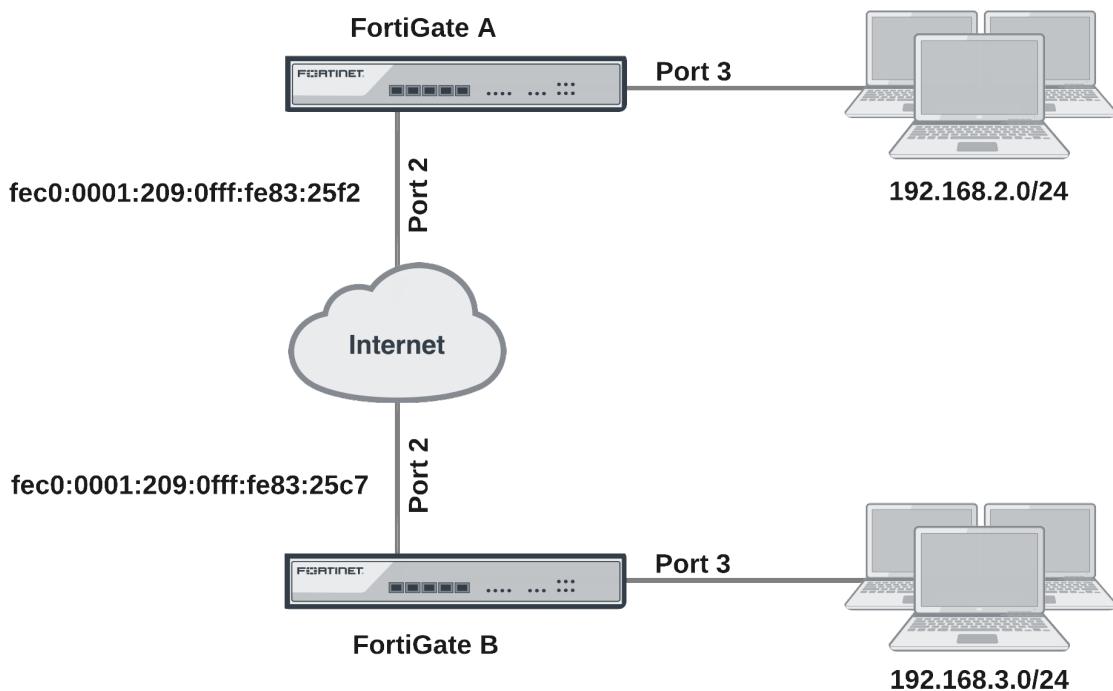
        set device port2
        set dst 0::/0
    next
    edit 2
        set device toA
        set dst fec0:0000:0000:0000::/64
    end

```

Site-to-site IPv4 over IPv6 VPN example

In this example, two private networks with IPv4 addressing communicate securely over IPv6 infrastructure.

Example IPv4-over-IPv6 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the IPv6 public network and port 3 connects to the IPv4 LAN.

```

config system interface
    edit port2
        config ipv6
            set ip6-address fec0::0001:209:0fff:fe83:25f2/64
        end
    next
    edit port3
        set 192.168.2.1/24
    end

```

Configure FortiGate A IPsec settings

The phase 1 configuration is the same as in the IPv6 over IPv6 example.

```
config vpn ipsec phase1-interface
  edit toB
    set ip-version 6
    set interface port2
    set remote-gw6 fec0:0000:0000:0003:209:0fff:fe83:25c7
    set dpd enable
    set psksecret maryhadalittlelamb
    set proposal 3des-md5 3des-sha1
  end
```

The phase 2 configuration is the same as you would use for an IPv4 VPN. By default, phase 2 selectors are set to accept all subnet addresses for source and destination.

```
config vpn ipsec phase2-interface
  edit toB
    set phase1name toB
    set proposal 3des-md5 3des-sha1
    set pfs enable
    set replay enable
  end
```

Configure FortiGate A security policies

Security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. These are IPv4 security policies.

```
config firewall policy
  edit 1
    set srcintf port3
    set dstintf toB
    set srcaddr all
    set dstaddr all
    set action accept
    set service ALL
    set schedule always
  next
  edit 2
    set srcintf toB
    set dstintf port3
    set srcaddr all
    set dstaddr all
    set action accept
    set service ALL
    set schedule always
  end
```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB using an IPv4 static route. A default route sends all IPv6 traffic, including the IPv6 IPsec packets, out on port2.

```
config router static6
  edit 1
    set device port2
```

```
    set dst 0::/0
next
edit 2
    set device toB
    set dst 192.168.3.0/24
end
```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the public IP address of FortiGate A. The IPsec phase 2 configuration has IPv4 selectors.

IPv4 security policies enable traffic to pass between the private network and the IPsec interface. An IPv4 static route ensures traffic for the private network behind FortiGate A goes through the VPN and an IPv6 static route ensures that all IPv6 packets are routed to the public network.

```
config system interface
edit port2
    config ipv6
        set ip6-address fec0::0003:fe83:25c7/64
    end
next
edit port3
    set 192.168.3.1/24
end
config vpn ipsec phase1-interface
edit toA
    set ip-version 6
    set interface port2
    set remote-gw6 fec0:0000:0000:0001:209:0fff:fe83:25f2
    set dpd enable
    set psksecret maryhadalittlelamb
    set proposal 3des-md5 3des-sha1
end
config vpn ipsec phase2-interface
edit toA2
    set phase1name toA
    set proposal 3des-md5 3des-sha1
    set pfs enable
    set replay enable
end
config firewall policy
edit 1
    set srcintf port3
    set dstintf toA
    set srcaddr all
    set dstaddr all
    set action accept
    set service ALL
    set schedule always
next
edit 2
    set srcintf toA
    set dstintf port3
    set srcaddr all
    set dstaddr all
```

```

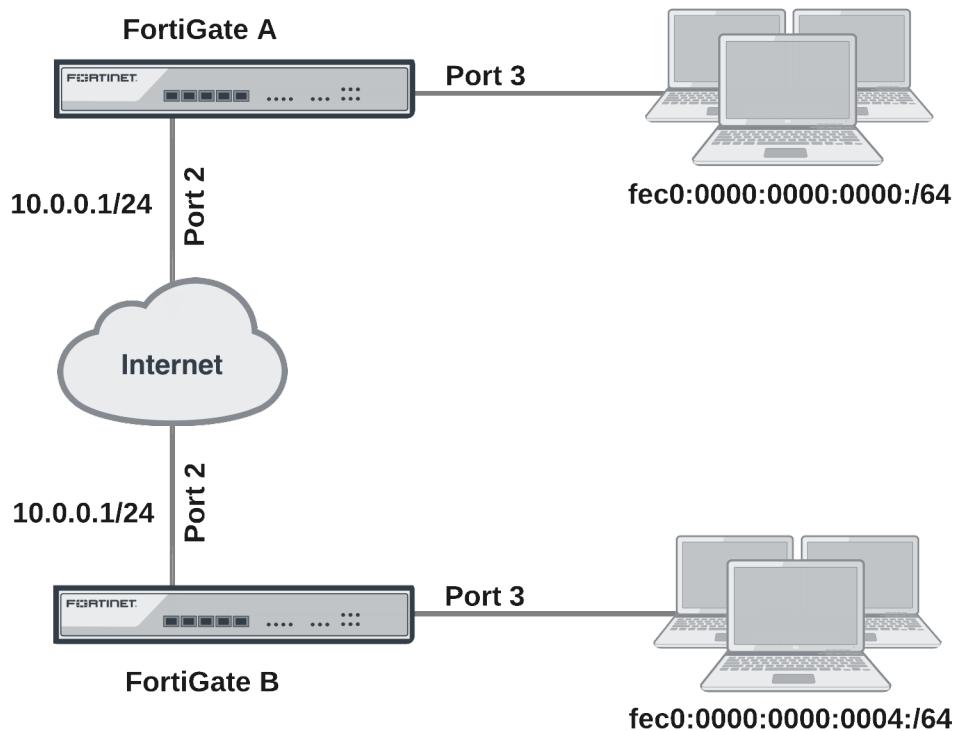
        set action accept
        set service ALL
        set schedule always
    end
config router static6
    edit 1
        set device port2
        set dst 0::/0
    next
    edit 2
        set device toA
        set dst 192.168.2.0/24
    end

```

Site-to-site IPv6 over IPv4 VPN example

In this example, IPv6-addressed private networks communicate securely over IPv4 public infrastructure.

Example IPv6-over-IPv4 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the IPv4 public network and port 3 connects to the IPv6 LAN.

```

config system interface
    edit port2
        set 10.0.0.1/24

```

```

next
edit port3
config ipv6
    set ip6-address fec0::0001:209:0fff:fe83:25f3/64
end

```

Configure FortiGate A IPsec settings

The phase 1 configuration uses IPv4 addressing.

```

config vpn ipsec phase1-interface
    edit toB
        set interface port2
        set remote-gw 10.0.1.1
        set dpd enable
        set psksecret maryhadalittlelamb
        set proposal 3des-md5 3des-sha1
    end

```

The phase 2 configuration uses IPv6 selectors. By default, phase 2 selectors are set to accept all subnet addresses for source and destination. The default setting for `src-addr-type` and `dst-addr-type` is `subnet`. The IPv6 equivalent is `subnet6`. The default subnet addresses are `0.0.0.0/0` for IPv4, `::/0` for IPv6.

```

config vpn ipsec phase2-interface
    edit toB2
        set phase1name toB
        set proposal 3des-md5 3des-sha1
        set pfs enable
        set replay enable
        set src-addr-type subnet6
        set dst-addr-type subnet6
    end

```

Configure FortiGate A security policies

IPv6 security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. Define the address `all16` using the `firewall address6` command as `::/0`.

```

config firewall policy6
    edit 1
        set srcintf port3
        set dstintf toB
        set srcaddr all16
        set dstaddr all16
        set action accept
        set service ALL
        set schedule always
    next
    edit 2
        set srcintf toB
        set dstintf port3
        set srcaddr all16
        set dstaddr all16
        set action accept
        set service ALL
        set schedule always
    end

```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB using an IPv6 static route. A default route sends all IPv4 traffic, including the IPv4 IPsec packets, out on port2.

```
config router static6
  edit 1
    set device toB
    set dst fec0:0000:0000:0004::/64
  end
config router static
  edit 1
    set device port2
    set dst 0.0.0.0/0
    set gateway 10.0.0.254
  end
```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the IPv4 public IP address of FortiGate A. The IPsec phase 2 configuration has IPv6 selectors.

IPv6 security policies enable traffic to pass between the private network and the IPsec interface. An IPv6 static route ensures traffic for the private network behind FortiGate A goes through the VPN and an IPv4 static route ensures that all IPv4 packets are routed to the public network.

```
config system interface
    edit port2
        set 10.0.1.1/24
    next
    edit port3
        config ipv6
            set ip6-address fec0::0004:209:0fff:fe83:2569/64
        end
config vpn ipsec phase1-interface
    edit toA
        set interface port2
        set remote-gw 10.0.0.1
        set dpd enable
        set psksecret maryhadalittlelamb
        set proposal 3des-md5 3des-sha1
    end
config vpn ipsec phase2-interface
    edit toA2
        set phase1name toA
        set proposal 3des-md5 3des-sha1
        set pfs enable
        set replay enable
        set src-addr-type subnet6
        set dst-addr-type subnet6
    end
config firewall policy6
    edit 1
        set srcintf port3
        set dstintf toA
        set srcaddr all6
        set dstaddr all6
        set action accept
        set service ALL
        set schedule always
    next
    edit 2
        set srcintf toA
        set dstintf port3
        set srcaddr all6
        set dstaddr all6
        set action accept
        set service ALL
        set schedule always
    end
config router static6
    edit 1
        set device toA
```

```

        set dst fec0:0000:0000:0000::/64
    end
config router static
    edit 1
        set device port2
        set gateway 10.0.1.254
    end

```

TCP MSS values

TCP MSS values, which control the maximum amount of data that can be sent in a single packet, can be set for IPv6 policies (for both the sender and the receiver). You can configure TCP MSS values in IPv6 using the following CLI commands:

```

config firewall policy6
    edit <index_int>
        set tcp-mss-sender <value>
        set tcp-mss-receiver <value>
    end

```

BGP and IPv6

FortiGate units support IPv6 over BGP using the same `config router bgp` command as IPv4, but different subcommands.

The main CLI keywords have IPv6 equivalents that are identified by the “6” on the end of the keyword, such as `with config network6` or `set allowas-in6`.

IPv6 BGP commands include:

```

config router bgp
    set activate6 {enable | disable}
    set allowas-in6 <max_num_AS_integer>
    set allowas-in-enable6 {enable | disable}
    set as-override6 {enable | disable}
    set attribute-unchanged6 [as-path] [med] [next-hop]
    set capability-default-originate6 {enable | disable}
    set capability-graceful-restart6 {enable | disable}
    set default-originate-route-map6 <routemap_str>
    set distribute-list-in6 <access-list-name_str>
    set distribute-list-out6 <access-list-name_str>
    set filter-list-in6 <aspath-list-name_str>
    set filter-list-out6 <aspath-list-name_str>
    set maximum-prefix6 <prefix_integer>
    set maximum-prefix-threshold6 <percentage_integer>
    set maximum-prefix-warning-only6 {enable | disable}
    set next-hop-self6 {enable | disable}
    set prefix-list-in6 <prefix-list-name_str>
    set prefix-list-out6 <prefix-list-name_str>
    set remove-private-as6 {enable | disable}
    set route-map-in6 <routemap-name_str>
    set route-map-out6 <routemap-name_str>
    set route-reflector-client6 {enable | disable}
    set route-server-client6 {enable | disable}
    set send-community6 {both | disable | extended | standard}
    set soft-reconfiguration6 {enable | disable}
    set unsuppress-map6 <route-map-name_str>

```

```
config network6
  config redistribute6
end
```

RIPng — RIP and IPv6

RIP next generation, or RIPng, is the version of RIP that supports IPv6.

This is an example of a typical small network configuration using RIPng routing.

Your internal R&D network is working on a project for a large international telecom company that uses IPv6. For this reason, you have to run IPv6 on your internal network and you have decided to use only IPv6 addresses.

Your network has two FortiGate units running the RIPng dynamic routing protocol. Both FortiGate units are connected to the ISP router and the internal network. This configuration provides some redundancy for the R&D internal network enabling it to reach the internet at all times.

This section includes the following topics:

- [Network layout and assumptions](#)
- [Configuring the FortiGate units system information](#)
- [Configuring RIPng on FortiGate units](#)
- [Configuring other network devices](#)
- [Testing the configuration](#)
- [Debugging IPv6 on RIPng](#)

Network layout and assumptions

Basic network layout

Your internal R&D network is working on a project for a large international telecom company that uses IPv6. For this reason, you have to run IPv6 on your internal network and you have decided to use only IPv6 addresses.

Your network has two FortiGate units running the RIPng dynamic routing protocol. Both FortiGate units are connected to the ISP router and the internal network. This configuration provides some redundancy for the R&D internal network enabling it to reach the internet at all times.

All internal computers use RIP routing, so no static routing is required. And all internal computers use IPv6 addresses.

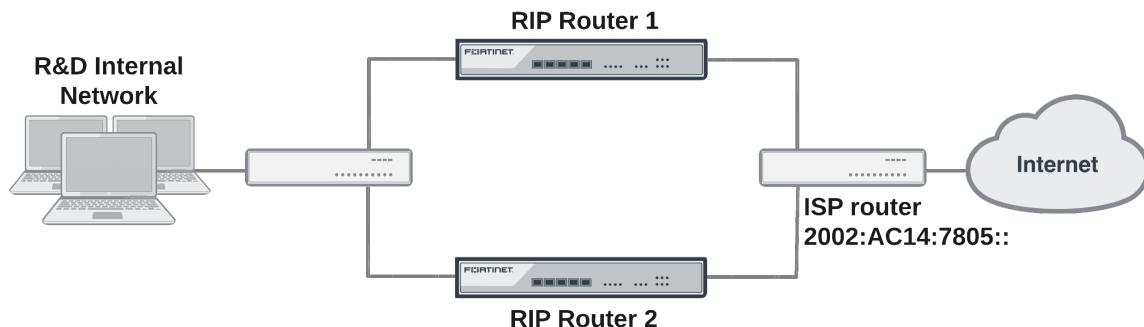
Where possible in this example, the default values will be used or the most general settings. This is intended to provide an easier configuration that will require less troubleshooting.

In this example the routers, networks, interfaces used, and IP addresses are as follows:

Rip example network topology

Network	Router	Interface & Alias	IPv6 address
R&D	Router1	port1 (internal)	2002:A0B:6565:0:0:0:0:0
		port2 (ISP)	2002:AC14:7865:0:0:0:0:0
	Router2	port1 (internal)	2002:A0B:6566:0:0:0:0:0
		port2 (ISP)	2002:AC14:7866:0:0:0:0:0

Network topology for the IPv6 RIPng example



Assumptions

The following assumptions have been made concerning this example:

- All FortiGate units have 5.0+ firmware, and are running factory default settings.
- All CLI and web-based manager navigation assumes the unit is running in NAT operating mode, with VDOMs disabled.
- All FortiGate units have interfaces labeled port1 and port2 as required.
- All firewalls have been configured for each FortiGate unit to allow the required traffic to flow across interfaces.
- All network devices are support IPv6 and are running RIPng.

Configuring the FortiGate units system information

Each FortiGate unit needs IPv6 enabled, a new hostname, and interfaces configured.

To configure system information on Router1 - web-based manager:

1. Go to **Dashboard**.
2. For **Host name**, select **Change**.
3. Enter “Router1”.
4. Go to **System > Feature Visibility**.
5. Enable **IPv6** and click **Apply**.

6. Go to **Network > Interfaces**.
7. Edit port1 (internal) interface.
8. Set the following information, and select **OK**.

Alias	internal
IP/Netmask	2002:A0B:6565::/0
Administrative Access	HTTPS SSH PING
Description	Internal RnD network
Administrative Status	Up

9. Edit port2 (ISP) interface.
10. Set the following information, and select **OK**.

Alias	ISP
IP/Netmask	2002:AC14:7865::/0
Administrative Access	HTTPS SSH PING
Description	ISP and internet
Administrative Status	Up

To configure system information on Router1 - CLI:

```

config system global
    set hostname Router1
    set gui-ipv6 enable
end
config system interface
    edit port1
        set alias internal
        set allowaccess https ping ssh
        set description "Internal RnD network"
        config ipv6
            set ip6-address 2002:a0b:6565::/0
        end
    next
    edit port2
        set alias ISP
        set allowaccess https ping ssh
        set description "ISP and internet"
        config ipv6
            set ip6-address 2002:AC14:7865::
        end
    end
end

```

To configure system information on Router2 - web-based manager:

1. Go to **Dashboard**.
2. For **Host name**, select **Change**.
3. Enter “Router2”.
4. Go to **System > Feature Visibility**.
5. Enable **IPv6** and click **Apply**.
6. Go to **Network > Interfaces**.
7. Edit port1 (internal) interface.
8. Set the following information, and select **OK**.

Alias	internal
IP/Netmask	2002:A0B:6566::/0
Administrative Access	HTTPS SSH PING
Description	Internal RnD network
Administrative Status	Up

9. Edit port2 (ISP) interface.
10. Set the following information, and select **OK**.

Alias	ISP
IP/Netmask	2002:AC14:7866::/0
Administrative Access	HTTPS SSH PING
Description	ISP and internet
Administrative Status	Up

To configure system information on Router2 - CLI:

```

config system global
  set hostname Router2
  set gui-ipv6 enable
end
config system interface
  edit port1
    set alias internal
    set allowaccess https ping ssh
    set description "Internal RnD network"
  config ipv6
    set ip6-address 2002:a0b:6566::/0
  end
next
  edit port2
    set alias ISP
    set allowaccess https ping ssh
    set description "ISP and internet"

```

```
config ipv6
    set ip6-address 2002:AC14:7866::
end
end
```

Configuring RIPng on FortiGate units

Now that the interfaces are configured, you can configure RIPng on the FortiGate units.

There are only two networks and two interfaces to include — the internal network, and the ISP network. There is no redistribution, and no authentication. In RIPng there is no specific command to include a subnet in the RIP broadcasts. There is also no information required for the interfaces beyond including their name.

As this is a CLI only configuration, configure the ISP router and the other FortiGate unit as neighbors. This was not part of the previous example as this feature is not offered in the web-based manager. Declaring neighbors in the configuration like this will reduce the discovery traffic when the routers start up.

Since RIPng is not supported in the web-based manager, this section will only be entered in the CLI.

To configure RIPng on Router1 - CLI:

```
config router ripng
    config interface
        edit port1
        next
        edit port2
        end
    config neighbor
        edit 1
        set interface port1
        set ipv6 2002:a0b:6566::/0
    next
        edit 2
        set interface port2
        set ipv6 2002:AC14:7805::/0
    end
```

To configure RIPng on Router2 - CLI:

```
config router ripng
    config interface
        edit port1
        next
        edit port2
        end
    config neighbor
        edit 1
        set interface port1
        set ipv6 2002:a0b:6565::/0
    next
        edit 2
        set interface port2
        set ipv6 2002:AC14:7805::/0
    end
```

Configuring other network devices

The other devices on the internal network all support IPv6, and are running RIPng where applicable. They only need to know the internal interface network addresses of the FortiGate units.

The ISP routers need to know the FortiGate unit information such as IPv6 addresses.

Testing the configuration

In addition to normal testing of your network configuration, you must also test the IPv6 part of this example.

For troubleshooting problems with your network, see the [FortiOS Handbook Troubleshooting](#) chapter.

Testing the IPv6 RIPng information

There are some commands to use when checking that your RIPng information is correct on your network. These are useful to check on your RIPng FortiGate units on your network. Comparing the output between devices will help you understand your network better, and also track down any problems.

```
diagnose ipv6 address list
```

View the local scope IPv6 addresses used as next-hops by RIPng on the FortiGate unit.

```
diagnose ipv6 route list
```

View ipv6 addresses that are installed in the routing table.

```
get router info6 routing-table
```

View the routing table. This information is almost the same as the previous command (diagnose ipv6 route list) however it is presented in an easier to read format.

```
get router info6 rip interface external
```

View brief output on the RIP information for the interface listed. The information includes if the interface is up or down, what routing protocol is being used, and whether passive interface or split horizon are enabled.

```
get router info6 neighbor-cache list
```

View the IPv6/MAC address mapping. This also displays the interface index and name associated with the address.

Debugging IPv6 on RIPng

The debug commands are very useful to see what is happening on the network at the packet level. There are a few changes to debugging the packet flow when debugging IPv6.

The following CLI commands specify both IPv6 and RIP, so only RIPng packets will be reported. The output from these commands will show you the RIPng traffic on your FortiGate unit including RECV, SEND, and UPDATE actions.

The addresses are in IPv6 format.

```
diagnose debug enable
diagnose ipv6 router rip level info
diagnose ipv6 router rip all enable
```

These three commands will:

Turn on debugging in general

Set the debug level to information, a verbose reporting level

Turn on all RIP router settings

Part of the information displayed from the debugging is the metric (hop count). If the metric is 16, then that destination is unreachable since the maximum hop count is 15.

In general, you should see an update announcement, followed by the routing table being sent out, and a received reply in response.

IPv6 RSSO support

RADIUS Single Sign-On (RSSO) is supported in IPv6, but can only be configured in the CLI:

```
config firewall policy6
  edit <id>
    set rsso enable
    set fall-through-unauthenticated enable
  end
```

IPv6 IPS

IPv6 IPS signature scan can be enabled by interface policy. The user can create an normal IPS sensor and assign it to the IPv6 interface policy.

```
config firewall interface-policy6
  edit 1
    set interface "port1"
    set srcaddr6 "all"
    set dstaddr6 "all"
    set service6 "ALL"
    set ips-sensor-status enable
    set ips-sensor "all_default"
  next
end
```

Blocking IPv6 packets by extension headers

FortiOS can now block IPv6 packets based on the extension headers, using the CLI syntax:

```
config firewall ipv6-eh-filter.
```

The following commands are now available:

- `set hop-opt {disable | enable}`: Block packets with Hop-by-Hop Options header.
- `set dest-opt {disable | enable}`: Block packets with Destination Options header.
- `set hdopt-type <integer>`: Block specific Hop-by-Hop and/or Destination Option types (maximum 7 types, each between 0 and 255).
- `set routing {disable | enable}`: Block packets with Routing header.
- `set routing-type <integer>`: Block specific Routing header types (maximum 7 types, each between 0 and 255).
- `set fragment {disable | enable}`: Block packets with Fragment header.
- `set auth {disable | enable}`: Block packets with Authentication header.
- `set no-next {disable | enable}`: Block packets with No Next header.

IPv6 denial of service policies

Denial of Service (DoS) policies can now be configured by going to **Policy & Objects > IPv6 DoS Policy**. For more information, refer to the “Interface Policies” section of the [FortiOS Handbook Firewall](#) chapter.

Configure hosts in an SNMP v1/2c community to send queries or receive traps

When you add a host to an SNMP v1/2c community you can now decide whether the FortiGate unit will accept queries from the host or whether the FortiGate unit will send traps to the host. You can also configure the host for both traps and queries. You can add up to 16 IPv4 hosts and up to 16 IPv6 hosts.

Use the following command to add two hosts to an SNMP community:

```
config system snmp community
  config hosts
    edit 1
      set interface port1
      set ip 172.20.120.1
      set host-type query
    end
  config hosts6
    edit 1
      set interface port6
      set ip 2001:db8:0:2::30
      set host-type trap
    end
```

IPv6 PIM sparse mode multicast routing

FortiOS supports PIM sparse mode multicast routing for IPv6 multicast (multicast6) traffic and is compliant with [RFC 4601](#). You can use the following command to configure IPv6 PIM sparse multicast routing.

```
config router multicast6
  set multicast-routing {enable | disable}
  config interface
    edit <interface-name>
      set hello-interval <1-65535 seconds>
      set hello-holdtime <1-65535 seconds>
    end
  config pim-sm-global
  config rp-address
    edit <index>
      set ipv6-address <ipv6-address>
    end
```

The following diagnose commands for IPv6 PIM sparse mode are also available:

```
diagnose ipv6 multicast status
diagnose ipv6 multicast vif
diagnose ipv6 multicast mroute
```

IPv6 Neighbor Discovery Proxy

The following is an example configuration of a FortiGate using ND Proxy. Some of these configuration steps have been covered elsewhere, but are shown here to demonstrate how they all work together to achieve the desired effect.

Steps:

- Create zone for ND proxy use that includes the upstream and downstream interfaces.
- Create policies to allow ICMPv6 and DHCPv6 traffic.
- Enable ND Proxy on the interfaces.
- Enable "autoconf" on the upstream interface.

1. Add a zone including wan and lan.

It is possible to use firewall and multicast policies that don't use a zone, but using a zone simplifies the configuration, especially if you have more than two interfaces.

```
config system zone
  edit ndproxy_zone
    set interface wan lan
  end
```



On some models the "lan" interface is named "internal".

2. Add forward firewall policy and multicast policy to allow at least ICMPv6 and DHCPv6 traffic.

```
config firewall multicast-policy6
  edit 0
    set srcintf ndproxy_zone
    set dstintf ndproxy_zone
    set srcaddr all
    set dstaddr all
  end
```

and

```
config firewall policy6
  edit 0
    set srcintf ndproxy_zone
    set dstintf ndproxy_zone
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  end
```

3. Enable ND proxy on WAN and LAN.

```
config system nd-proxy
  set status enable
  set member wan lan
end
```

4. Enable autoconf on the upstream interface.

RA received on the other interface(s) will be dropped.

```
config system interface
  edit wan
  ...
  config ipv6
    set autoconf enable
  end
end
```

Network defense

This section describes in general terms the means by which attackers can attempt to compromise your network using attacks at the network level rather than through application vulnerabilities, and steps you can take to protect it. The goal of an attack can be as complex as gaining access to your network and the privileged information it contains, or as simple as preventing customers from accessing your web server.

Because of popular media, many people are aware of viruses and other malware as a threat against their computers and data, but some of the most costly malicious attack in history have been against networks. A 2016 study found that a single DDoS attack could cost a company over \$1.6 million. Depending on the size and type of company the areas of expense can be:

- Changes in credit and insurance ratings
- Overtime payment to employees
- Hiring new employees to increase IT staff
- PR expenses to restore a company's reputation
- Upgrading infrastructure and software
- Customer compensation

The following topics are included in this section:

- Monitoring
- Blocking external probes
- Defending against DoS attacks

Monitoring

Monitoring, in the form of logging, alert email, and SNMP, does not directly protect your network. But monitoring allows you to review the progress of an attack, whether afterwards or while in progress. How the attack unfolds may reveal weaknesses in your preparations. The packet archive and sniffer policy logs can reveal more details about the attack. Depending on the detail in your logs, you may be able to determine the attacker's location and identity.

While log information is valuable, you must balance the log information with the resources required to collect and store it.

Blocking external probes

Protection against attacks is important, but attackers often use vulnerabilities and network tools to gather information about your network to plan an attack. It is often easier to prevent an attacker from learning important details about your network than to defend against an attack designed to exploit your particular network.

Attacks are often tailored to the hardware or operating system of the target, so reconnaissance is often the first step. The IP addresses of the hosts, the open ports, and the operating systems the hosts are running is invaluable information to an attacker. Probing your network can be as simple as an attacker performing an

address sweep or port scan to a more involved operation like sending TCP packets with invalid combinations of flags to see how your firewall reacts.

Address sweeps

An address sweep is a basic network scanning technique to determine which addresses in an address range have active hosts. A typical address sweep involves sending an ICMP ECHO request (a ping) to each address in an address range to attempt to get a response. A response signifies that there is a host at this address that responded to the ping. It then becomes a target for more detailed and potentially invasive attacks.

Address sweeps do not always reveal all the hosts in an address range because some systems may be configured to ignore ECHO requests and not respond, and some firewalls and gateways may be configured to prevent ECHO requests from being transmitted to the destination network. Despite this shortcoming, Address sweeps are still used because they are simple to perform with software tools that automate the process.

Use the `icmp_sweep` anomaly in a DoS policy to protect against address sweeps.

There are a number of IPS signatures to detect the use of ICMP probes that can gather information about your network. These signatures include `AddressMask`, `Traceroute`, `ICMP.Invalid.Packet.Size`, and `ICMP.Oversized.Packet`. Include ICMP protocol signatures in your IPS sensors to protect against these probes/attacks.

Port scans

Potential attackers may run a port scan on one or more of your hosts. This involves trying to establish a communication session to each port on a host. If the connection is successful, a service may be available that the attacker can exploit.

Use the DoS anomaly check for `tcp_port_scan` to limit the number of sessions (complete and incomplete) from a single source IP address to the configured threshold. If the number of sessions exceed the threshold, the configured action is taken.

Use the DoS anomaly check for `udp_scan` to limit UDP sessions in the same way.

Probes using IP traffic options

Every TCP packet has space reserved for eight flags or control bits. They are used for communicating various control messages. Although space in the packet is reserved for all eight, there are various combinations of flags that should never happen in normal network operation. For example, the SYN flag, used to initiate a session, and the FIN flag, used to end a session, should never be set in the same packet.

Attackers may create packets with these invalid combinations to test how a host will react. Various operating systems and hardware react in different ways, giving a potential attackers clues about the components of your network.

The IPS signature `TCP.Bad.Flags` detects these invalid combinations. The default action is pass though you can override the default and set it to **Block** in your IPS sensor.

Configure packet replay and TCP sequence checking

The `anti-replay` command in the CLI allows you to set the level of checking for packet replay and TCP sequence checking (or TCP Sequence (SEQ) number checking). All TCP packets contain a Sequence Number

(SEQ) and an Acknowledgment Number (ACK). The TCP protocol uses these numbers for error free end-to-end communications. TCP sequence checking can also be used to validate individual packets.

FortiGate units use TCP sequence checking to make sure that a packet is part of a TCP session. By default, if a packet is received with sequence numbers that fall out of the expected range, the FortiGate unit drops the packet. This is normally a desired behavior, since it means that the packet is invalid. But in some cases you may want to configure different levels of `anti-replay` checking if some of your network equipment uses non-RFC methods when sending packets.

Configure the anti-replay CLI command:

```
config system global
    set anti-replay {disable | loose | strict}
end
```

You can set `anti-replay` protection to the following settings:

- `disable` — No anti-replay protection.
- `loose` — Perform packet sequence checking and ICMP anti-replay checking with the following criteria:
 - The SYN, FIN, and RST bit can not appear in the same packet.
 - The FortiGate unit does not allow more than one ICMP error packet through before it receives a normal TCP or UDP packet.
 - If the FortiGate unit receives an RST packet, and `check-reset-range` is set to strict, the FortiGate unit checks to determine if its sequence number in the RST is within the un-ACKed data and drops the packet if the sequence number is incorrect.
- `strict` — Performs all of the loose checking but for each new session also checks to determine if the TCP sequence number in a SYN packet has been calculated correctly and started from the correct value for each new session. Strict anti-replay checking can also help prevent SYN flooding.

If any packet fails a check it is dropped.



The anti-replay setting only affects non-accelerated traffic.

Configure ICMP error message verification

Enable ICMP error message verification to ensure an attacker can not send an invalid ICMP error message.

```
config system global
    check-reset-range {disable | strict}
end
```

- `disable` — the FortiGate unit does not validate ICMP error messages.
- `strict` — enable ICMP error message checking.

If the FortiGate unit receives an ICMP error packet that contains an embedded IP(A,B) | TCP(C,D) header, then if FortiOS can locate the A:C->B:D session it checks to make sure that the sequence number in the TCP header is within the range recorded in the session. If the sequence number is not in range then the ICMP packet is dropped. Strict checking also affects how the anti-replay option checks packets.

Protocol header checking

Select the level of checking performed on protocol headers.

```
config system global
```

```
check-protocol-header {loose | strict}
end
```

- **loose** — the FortiGate unit performs basic header checking to verify that a packet is part of a session and should be processed. Basic header checking includes verifying that the layer-4 protocol header length, the IP header length, the IP version, the IP checksum, IP options are correct, etc.
- **strict** — the FortiGate unit does the same checking as above plus it verifies that ESP packets have the correct sequence number, SPI, and data length.

If the packet fails header checking it is dropped by the FortiGate unit.

Evasion techniques

Attackers employ a wide range of tactics to try to disguise their techniques. If an attacker disguises a known attack in such a way that it is not recognized, the attack will evade your security and possibly succeed. FortiGate security recognizes a wide variety of evasion techniques and normalizes data traffic before inspecting it.

Packet fragmentation

Information sent across local networks and the Internet is encapsulated in packets. There is a maximum allowable size for packets and this maximum size varies depending on network configuration and equipment limitations. If a packet arrives at a switch or gateway and it is too large, the data it carries is divided among two or more smaller packets before being forwarded. This is called fragmentation.

When fragmented packets arrive at their destination, they are reassembled and read. If the fragments do not arrive together, they must be held until all of the fragments arrive. Reassembly of a packet requires all of the fragments.

The FortiGate unit automatically reassembles fragmented packets before processing them because fragmented packets can evade security measures. This reassembly of packets affects TCP, UDP and IP packets. There can be some variation though in what process does the reassembling. The IPS engine, nTurbo and the kernel all can do defragmentation.

For example, you have configured the FortiGate unit to block access to the example.org web site. Any checks for example.com will fail if a fragmented packet arrives and one fragment contains `http://www.exa` while the other contains `mple.com/`. Viruses and malware can be fragmented and avoid detection in the same way. The FortiGate unit will reassemble fragmented packets before examining network data to ensure that inadvertent or deliberate packet fragmentation does not hide threats in network traffic.

Non-standard ports

Most traffic is sent on a standard port based on the traffic type. The FortiGate unit recognizes most traffic by packet content rather than the TCP/UDP port and uses the proper IPS signatures to examine it. Protocols recognized regardless of port include DHCP, DNP3, FTP, HTTP, IMAP, MS RPC, NNTP, POP3, RSTP, SIP, SMTP, and SSL, as well as the supported IM/P2P application protocols.

In this way, the FortiGate unit will recognize HTTP traffic being sent on port 25 as HTTP rather than SMTP, for example. Because the protocol is correctly identified, the FortiGate unit will examine the traffic for any enabled HTTP signatures.

Negotiation codes

Telnet and FTP servers and clients support the use of negotiation information to allow the server to report what features it supports. This information has been used to exploit vulnerable servers. To avoid this problem, the

FortiGate unit removes negotiation codes before IPS inspection.

HTTP URL obfuscation

Attackers encode HTML links using various formats to evade detection and bypass security measures. For example, the URL `www.example.com/cgi.bin` could be encoded in a number of ways to avoid detection but still work properly, and be interpreted the same, in a web browser.

The FortiGate prevents the obfuscation by converting the URL to ASCII before inspection.

HTTP URL obfuscation types

Encoding type	Example
No encoding	<code>http://www.example.com/cgi.bin/</code>
Decimal encoding	<code>http://www.example.com/99;103;&#105;&#46;&#98;&#105;&#110;&#47;</code>
URL encoding	<code>http://www.example.com/%43%47%49%2E%42%49%4E%2F</code>
ANSI encoding	<code>http://www.example.com/%u0063%u0067%u0069%u002E%u0062%u0069%u006E/</code>
Directory traversal	<code>http://www.example.com/cgi.bin/test/..</code>

HTTP header obfuscation

The headers of HTTP requests or responses can be modified to make the discovery of patterns and attacks more difficult. To prevent this, the FortiGate unit will:

- remove junk header lines
- reassemble an HTTP header that's been folded onto multiple lines
- move request parameters to HTTP POST body from the URL

The message is scanned for any enabled HTTP IPS signatures once these problems are corrected.

HTTP body obfuscation

The body content of HTTP traffic can be hidden in an attempt to circumvent security scanning. HTTP content can be GZipped or deflated to prevent security inspection. The FortiGate unit will uncompress the traffic before inspecting it.

Another way to hide the contents of HTTP traffic is to send the HTTP body in small pieces, splitting signature matches across two separate pieces of the HTTP body. The FortiGate unit reassembles these 'chunked bodies' before inspection.

Microsoft RPC evasion

Because of its complexity, the Microsoft Remote Procedure Call protocol suite is subject to a number of known evasion techniques, including:

- SMB-level fragmentation
- DCERPC-level fragmentation
- DCERPC multi-part fragmentation
- DCERPC UDP fragmentation
- Multiple DCERPC fragments in one packet

The FortiGate unit reassembles the fragments into their original form before inspection.

Defending against DoS attacks

A denial of service is the result of an attacker sending an abnormally large amount of network traffic to a target system. Having to deal with the traffic flood slows down or disables the target system so that legitimate users can not use it for the duration of the attack.

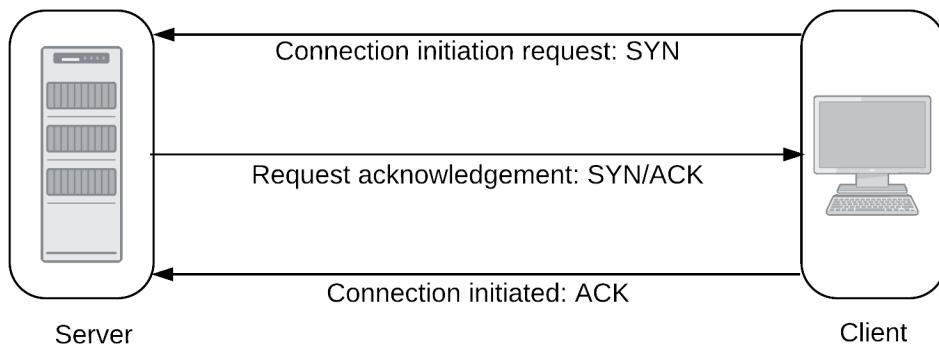
Any network traffic the target system receives has to be examined, and then accepted or rejected. TCP, UDP, and ICMP traffic is most commonly used, but a particular type of TCP traffic is the most effective. TCP packets with the SYN flag are the most efficient DoS attack tool because of how communication sessions are started between systems.

The “three-way handshake”

Communication sessions between systems start with establishing a TCP/IP connection. This is a simple three step process, sometimes called a “three-way handshake,” initiated by the client attempting to open the connection.

1. The client sends a TCP packet with the SYN flag set. With the SYN packet, the client informs the server of its intention to establish a connection.
2. If the server is able to accept the connection to the client, it sends a packet with the SYN and the ACK flags set. This simultaneously acknowledges the SYN packet the server has received, and informs the client that the server intends to establish a connection.
3. To acknowledge receipt of the packet and establish the connection, the client sends an ACK packet.

Establishing a TCP/IP connection



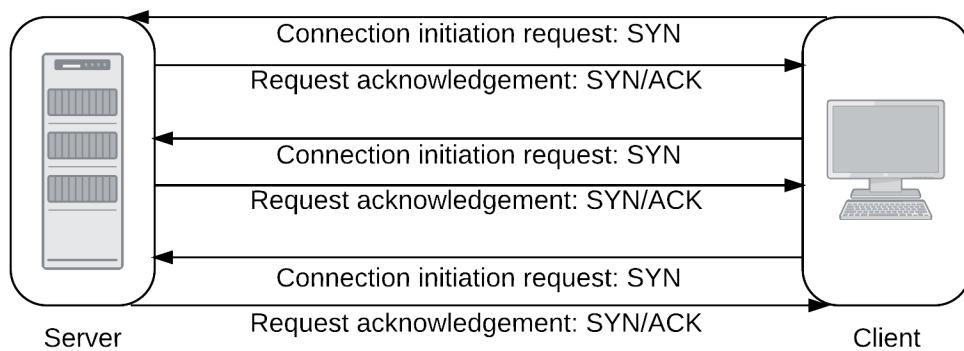
The three-way handshake is a simple way for the server and client to each agree to establish a connection and acknowledge the other party expressing its intent. Unfortunately, the three-way handshake can be used to interfere with communication rather than facilitate it.

SYN flood

When a client sends a SYN packet to a server, the server creates an entry in its session table to keep track of the connection. The server then sends a SYN+ACK packet expecting an ACK reply and the establishment of a connection.

An attacker intending to disrupt a server with a denial of service (DoS) attack can send a flood of SYN packets and not respond to the SYN+ACK packets the server sends in response. Networks can be slow and packets can get lost so the server will continue to send SYN+ACK packets until it gives up, and removes the failed session from the session table. If an attacker sends enough SYN packets to the server, the session table will fill completely, and further connection attempts will be denied until the incomplete sessions time out. Until this happens, the server is unavailable to service legitimate connection requests.

A single client launches a SYN flood attack

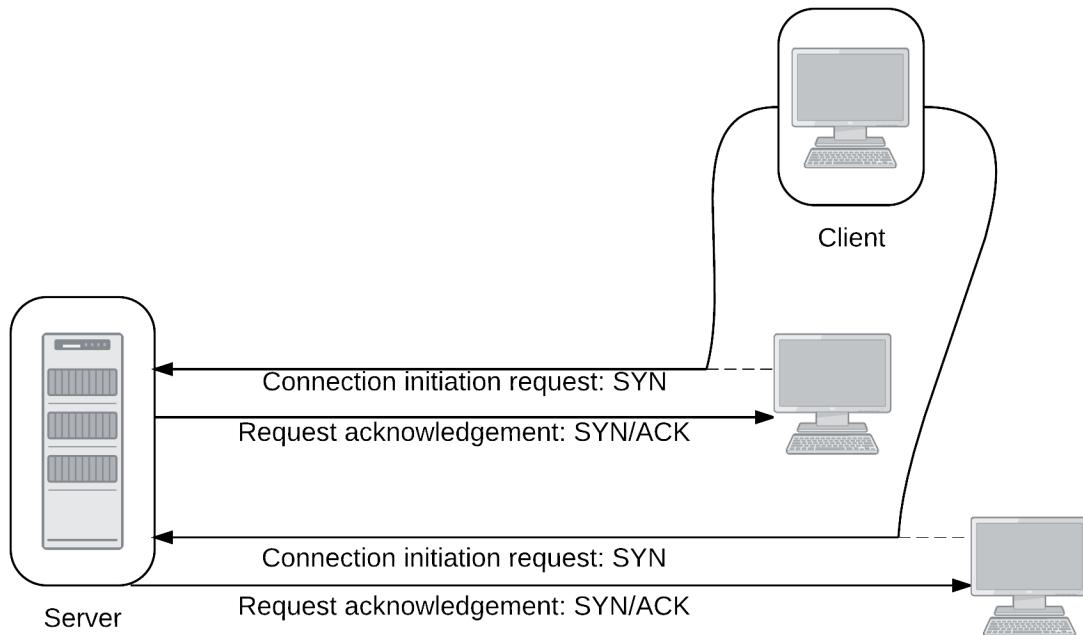


SYN floods are seldom launched from a single address so limiting the number of connection attempts from a single IP address is not usually effective.

SYN spoofing

With a flood of SYN packets coming from a single attacker, you can limit the number of connection attempts from the source IP address or block the attacker entirely. To prevent this simple defense from working, or to disguise the source of the attack, the attacker may spoof the source address and use a number of IP addresses to give the appearance of a distributed denial of service (DDoS) attack. When the server receives the spoofed SYN packets, the SYN+ACK replies will go to the spoofed source IP addresses which will either be invalid, or the system receiving the reply will not know what to do with it.

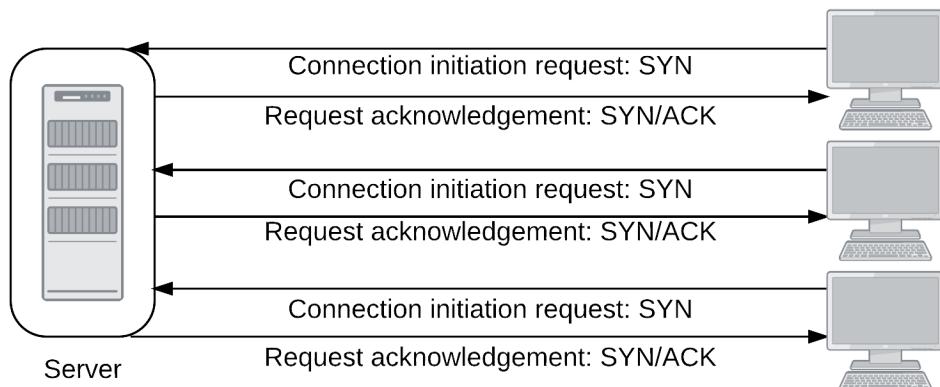
A client launches a SYN spoof attack



DDoS SYN flood

The most severe form of SYN attack is the distributed SYN flood, one variety of distributed denial of service attack (DDoS). Like the SYN flood, the target receives a flood of SYN packets and the ACK+SYN replies are never answered. The attack is distributed across multiple sources sending SYN packets in a coordinated attack.

Multiple attackers launch a distributed SYN flood



The distributed SYN flood is more difficult to defend against because multiple clients are capable of creating a larger volume of SYN packets than a single client. Even if the server can cope, the volume of traffic may

overwhelm a point in the network upstream of the targeted server. The only defense against this is more bandwidth to prevent any choke points.

Configuring the SYN threshold to prevent SYN floods

The preferred primary defense against any type of SYN flood is the DoS anomaly check for `tcp_syn_flood` threshold. The threshold value sets an upper limit on the number of new incomplete TCP connections allowed per second. If the number of incomplete connections exceeds the threshold value, and the action is set to **Pass**, the FortiGate unit will allow the SYN packets that exceed the threshold. If the action is set to **Block**, the FortiGate unit will block the SYN packets that exceed the threshold, but it will allow SYN packets from clients that send another SYN packet.

The tools attackers use to generate network traffic will not send a second SYN packet when a SYN+ACK response is not received from the server. These tools will not “retry.” Legitimate clients will retry when no response is received, and these retries are allowed even if they exceed the threshold with the action set to **Block**.

SYN proxy

FortiGate units with network acceleration hardware, whether built-in or installed in the form of an add-on module, offer a third action for the `tcp_syn_flood` threshold. Instead of **Block** and **Pass**, you can choose to **Proxy** the incomplete connections that exceed the threshold value.

When the `tcp_syn_flood` threshold action is set to **f**, incomplete TCP connections are allowed as normal as long as the configured threshold is not exceeded. If the threshold is exceeded, the FortiGate unit will intercept incoming SYN packets from clients and respond with a SYN+ACK packet. If the FortiGate unit receives an ACK response as expected, it will “replay” this exchange to the server to establish a communication session between the client and the server, and allow the communication to proceed.

Other flood types

UDP and ICMP packets can also be used for DoS attacks, though they are less common. TCP SYN packets are so effective because the target receives them and maintains a session table entry for each until they time out. Attacks using UDP or ICMP packets do not require the same level of attention from a target, rendering them less effective. The target will usually drop the offending packets immediately, closing the session.

Use the `udp_flood` and `icmp_flood` thresholds to defend against these DoS attacks.

DoS policies

DDoS attacks vary in nature and intensity. Attacks aimed at saturating the available bandwidth upstream of your service can only be countered by adding more bandwidth. DoS policies can help protect against DDoS attacks that aim to overwhelm your server resources.

DoS policy recommendations

- Use and configure DoS policies to appropriate levels based on your network traffic and topology. This will help drop traffic if an abnormal amount is received.
- It is important to set a good threshold. The threshold defines the maximum number of sessions/packets per second of normal traffic. If the threshold is exceeded, the action is triggered. Threshold defaults are general recommendations, although your network may require very different values.

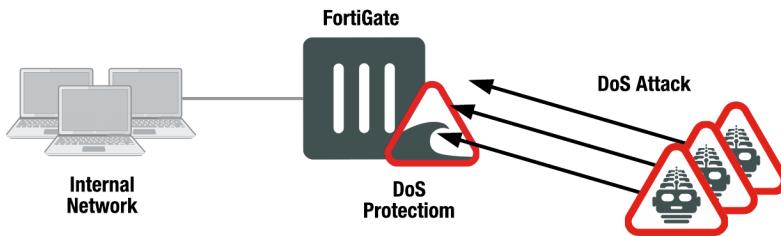
- One way to find the correct values for your environment is to set the action to **Pass** and enable logging. Observe the logs and adjust the threshold values until you can determine the value at which normal traffic begins to generate attack reports. Set the threshold above this value with the margin you want. Note that the smaller the margin, the more protected your system will be from DoS attacks, but your system will also be more likely to generate false alarms.

Inside FortiOS: Denial of Service (DoS) protection

FortiOS DoS protection maintains network integrity and performance by identifying and blocking harmful IPv4 and IPv6-based denial of service (DoS) attacks.

About DoS and DDoS attacks

A denial of service (DoS) occurs when an attacker overwhelms server resources by flooding a target system with anomalous data packets, rendering it unable to service genuine users. A distributed denial of service (DDoS) occurs when an attacker uses a master computer to control a network of compromised systems, otherwise known as a 'botnet', which collectively inundates the target system with excessive anomalous data packets.

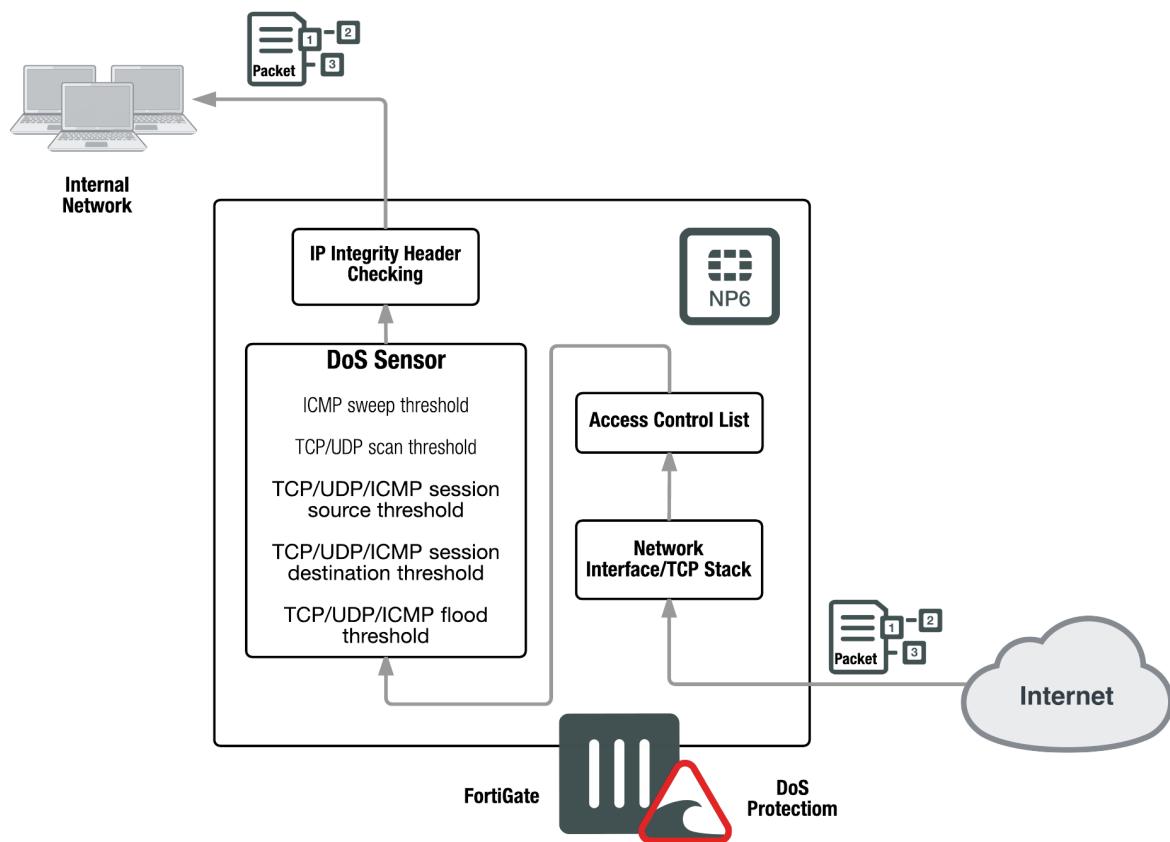


FortiOS DoS and DDoS protection

FortiOS DoS protection identifies potentially harmful traffic that could be part of a DoS or a DDoS attack by looking for specific traffic anomalies. Traffic anomalies that become DoS attacks include: TCP SYN floods, UDP floods, ICMP floods, TCP port scans, TCP session attacks, UDP session attacks, ICMP session attacks, and ICMP sweep attacks. Only traffic identified as part of a DoS attack is blocked; connections from legitimate users are processed normally.

FortiOS applies DoS protection very early in its traffic processing sequence to minimize the effect of a DoS attack on FortiOS system performance. DoS protection is the first step for packets after they are received by a FortiGate interface. Potential DoS attacks are detected and blocked before the packets are sent to other FortiOS systems.

FortiOS also includes an access control list feature that is implemented next. This accelerated ACL technology uses NP6 processors to block traffic (including DoS attacks) by source and destination address and service again before the packets are sent to the FortiGate CPU.



FortiOS DoS protection can operate in a standard configuration or operate out of band in sniffer mode, also known as one-arm mode, similar to intrusion detection systems. When operating in sniffer mode the FortiGate unit detects attacks and logs them without blocking them.

FortiOS DoS policies determine the course of action to take when anomalous traffic reaches a configured packet rate threshold. You can block an attacker, block an interface, block an attacker and interface, or allow traffic to pass through for monitoring purposes. This allows you to maintain network security by gathering information about attacks, monitor potentially offending traffic, or block offenders for the most protection.

FortiGates with NP6 processors also support synproxy DoS protection. An NP6-accelerated TCP SYN proxy offloads the three-way TCP handshake TCP SYN anomaly checking DoS protection to NP6 processors.

FortiOS DDoS prevention

In addition to using DoS protection for protection against DoS attacks, FortiOS includes a number of features that prevent the spread of Botnet and C&C activity. Mobile Malware or Botnet and C&C protection keeps Botnet and C&C code from entering a protected network and compromising protected systems. As a result, systems on the protected network cannot become Botnet clients.

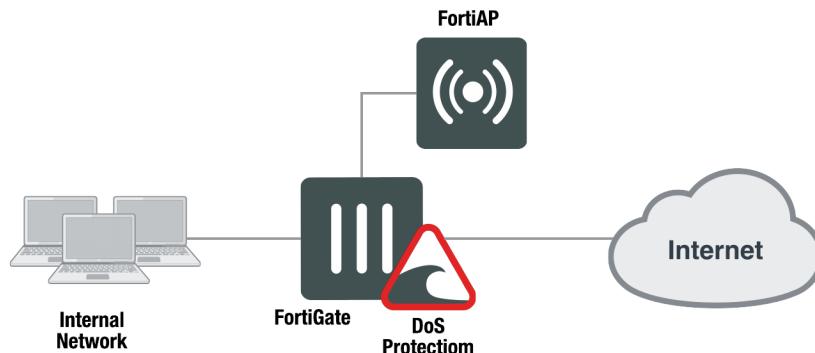
In addition, FortiOS can monitor and block outgoing Botnet connection attempts. Monitoring allows you to find and remove Botnet clients from your network and blocking prevents infected systems from communicating with Botnet sites.

Configuration options

Choose the standard configuration for maximum protection or configure sniffer mode to gather information.

Standard configuration

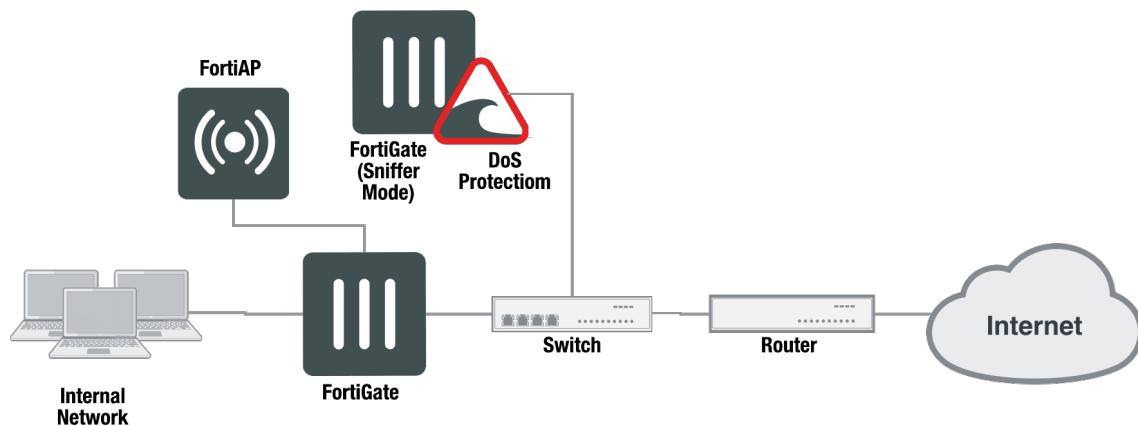
DoS protection is commonly configured on a FortiGate unit that connects a private or DMZ network to the Internet or on a FortiWiFi unit that connects a wireless LAN to an internal network and to the Internet. All Internet traffic or wireless LAN traffic passes through DoS protection in the FortiGate unit or the FortiWiFi unit.



Out of band configuration (sniffer mode)

A FortiGate unit in sniffer mode operates out of band as a one-armed Intrusion Detection System by detecting and reporting attacks. It does not process network traffic nor does it take action against threats. The FortiGate interface operating in sniffer mode is connected to a Test Access Point (TAP) or a Switch Port Analyzer (SPAN) port that processes all of the traffic to be analyzed. The TAP or SPAN sends a copy of the switch traffic to the out of band FortiGate for analysis.

FortiOS records log messages and sends alerts to system administrators when a DoS attack is detected. IDS scanning does not affect network performance or network traffic if the IDS fails or goes offline.



DoS policies

DoS policies provide effective and early DoS detection while remaining light on system resources. They are configured to monitor and to stop traffic with abnormal patterns or attributes. The DoS policy recognizes traffic as a threat when the traffic reaches a user-configured packet rate threshold. The policy then determines the appropriate action. In addition to choosing whether or not to log each type of anomaly, you can choose to pass or block threats.

DoS policy anomaly protection is applied to all incoming traffic to a single FortiGate interface, but you can narrow policies by specifying service, source address, and destination address. The FortiGate unit processes DoS policies in their own respective order first, followed by all other firewall policies.

Hardware acceleration

Hardware acceleration enhances protection and increases the efficiency of your network. FortiOS integrated Content Processors (CPs), Network Processors (NPs), and Security Processors (SPs) accelerate specialized security processing. DoS SYN proxy protection is built in to NP6 processors and many Fortinet Security Processors, like the CE4, XE2, and FE8, to guard against TCP SYN floods. TCP packets with the SYN flag are the most efficient DoS attack tool because of how communication sessions are initiated between systems. NP6 and SP processors can offload TCP SYN flood attack detection and blocking. The SP module increases a FortiGate unit's capacity to protect against TCP SYN flood attacks while minimizing the effect of attacks on the FortiGate unit's overall performance and the network performance. The result is improved capacity and overall system performance.

The FortiGuard Center

The FortiGuard Center shows information on all the most recent FortiGuard news, including information concerning zero-day research and hot intrusion detections. Research papers are also available that concern a

variety of current security issues.

To view recent developments, go to <http://www.fortiguard.com/static/intrusionprevention.html>.

Policy configuration

The firewall policies of the FortiGate are one of the most important aspects of the appliance. There are a lot of building blocks and configurations involved in setting up a firewall and it is within the policies that a lot of these components come together to form a cohesive unit to perform the firewall's main function, analyzing network traffic and responding appropriately to the results of that analysis.

There are a few different kinds of policies and in most cases these are further divided into IPv4 and IPv6 versions:

- [IPv4 policy](#) - used for managing traffic going through the appliance using IPv4 protocols
- [IPv6 policy](#) - used for managing traffic going through the appliance using IPv6 protocols
- [NAT64 policy](#) - used for managing traffic going through the appliance that converts from IPv6 on the incoming interface to IPv4 on the outgoing interface
- [NAT46 policy](#) - used for managing traffic going through the appliance that converts from IPv4 on the incoming interface to IPv6 on the outgoing interface
- [Multicast policy](#) - used to manage traffic sent to multiple destinations
- [IPv4 access control list](#) - used to filter out packets based on specific IPv4 parameters.
- [IPv6 access control list](#) - used to filter out packets based on specific IPv6 parameters.
- [IPv4 DoS policy](#) - used to prevent malicious or flawed packets on an IPv4 interface from denying access to users.
- [IPv6 DoS policy](#) - used to prevent malicious or flawed packets on an IPv6 interface from denying access to users.

Because the policy determines whether or not NAT will be used, it is also important to look at how to configure:

- [Central SNAT](#) - used for granular controlling when NATing is in use.

Viewing firewall policies

To find a Policy window, follow one of these paths in the GUI:

- **Policy & Objects> IPv4 Policy**
- **Policy & Objects> IPv6 Policy**
- **Policy & Objects> NAT64 Policy**
- **Policy & Objects> NAT46 Policy**
- **Policy & Objects> Proxy Policy**
- **Policy & Objects> Multicast Policy**

You may notice other policy options on the left window pane such as:

- **Policy & Objects> IPv4 DoS Policy**
- **Policy & Objects> IPv6 DoS Policy**
- **Policy & Objects> Local InPolicy**

These are different enough that they have their own descriptions in the sections that relate to them.

Menu items

There are some variations, but there are some common elements shared by all of them. There is a menu bar across the top. The menu bar will have the following items going from left to right:

- **Create New** button
- **Edit** button
- **Delete** button
- **Search** field
- **Interface Pair View**- Displays the policies in the order that they are checked for matching traffic, grouped by the pairs of Incoming and Outgoing interfaces. For instance, all of the policies referencing traffic from WAN1 to DMZ will be in one section. The policies referencing traffic from DMZ to WAN1 will be in another section. The sections are collapsible so that you only need to look at the sections with policies you are interested in.
- **By Sequence**- Displays the policies in the order that they are checked for matching traffic without any grouping.

Menu items not shared by all policies

- **Policy Lookup** - (IPv4, IPv6)
- **NAT64 Forwarding** - (NAT64)

The Table of Policies

Columns

The tables that make up the Policy window are based on rows which represent individual policies and the columns that represent the various parameters or status within the policy. The columns are customizable by which columns are included and what order they are in.

The table can be laid out a number of ways to suit the viewer. There is a column for most of the important pieces of information that you might be interested in seeing, but a lot of them are hidden by default. If you had a large enough screen, you might be able to show all of the columns, but even then it might look a bit busy and crammed together. Figure out which pieces of information are most important to you and hide the rest.

To configure which columns are visible and which are hidden, right click on the header row of the table. This will present a drop down menu. The drop down will be divided into sections. At the top will be the **Selected Columns** which are currently visible, and the next section will be **Available Columns** which show which columns are available to add to the table.

To move a column from the **Available** list to the **Selected** list just click on it. To move a column from the **Selected** list to the **Available** list, it also just takes a click of the mouse. To make the changes show up on the table, go to the bottom of the drop down menu and select **Apply**. Any additions to the table will show up on the right side.

One of the more useful ones that can be added is the ID column. The reason for adding this one is that within the configuration file and CLI, the policies are referenced by their ID number. Some policy settings are only available for configuration in the CLI. If you are looking in the CLI you will see that the only designation for a policy is its number and if you wish to edit the policy or change its order in the sequence you will be asked to move it before or after another policy by referencing its number.

How “Any” policy can remove the Interface Pair View

The FortiGate unit will automatically change the view on the policy list page to **By Sequence** whenever there is a policy containing “any” as the Source or Destination interface. If the **Interface Pair View** is grayed out it is likely that one or more of the policies has used the “any” interface.

By using the “any” interface, the policy should go into multiple sections because it could effectively be any of a number of interface pairings. As mentioned, policies are sectioned by using the interface pairings (for example, port1 -> port2) and each section has its own specific policy order. The order in which a policy is checked for matching criteria to a packet’s information is based solely on the position of the policy within its section or within the entire list of policies as a whole but if the policy is in multiple sections at the same time there is no mechanism for placing the policy in a proper order within all of those sections at the same time because it is a manual process and there is no parameter to compare the precedence of one section or policy over the other. Thus a conflict is created. In order to resolve the conflict the FortiGate firewall removes that aspect of the sections so that there is no need to compare and find precedence between the sections and it therefore has only the Global View to work with.

Policy names

Each policy has a name field. Every policy name must be unique for the current VDOM regardless of policy type. Previous to FortiOS 5.4, this field was optional.



On upgrading from an earlier version of FortiOS to 5.4, policy names are not assigned to old policies, but when configuring new policies, a unique name must be assigned to the policy.

Configuring the Name field

GUI

In the GUI, the field for the policy name is the first field on the editing page.

CLI

In the CLI, the syntax for assigning the policy name is:

```
config firewall [policy|policy6]
  edit 0
    set name <policy name>
  end
```

Disabling policy name requirement

While by default the requirement of having a unique name for each policy is the default, it can be enabled or disabled. Oddly enough, if disabling the requirement is a one time thing, doing it in the CLI is more straightforward.



This setting is VDOM based so if you are running multiple VDOMs, you will have to enter the correct VDOM before entering the CLI commands or turning the feature on or off in the GUI.

GUI

To edit the requirement in the GUI, the ability to do so must be enabled in the CLI. The syntax is:

```
config system settings
  set gui-allow-unnamed-policy [enable|disable]
end
```

Once it has been enabled, the requirement for named policies can be relaxed by going to **System > Feature Visibility**. Allow **Unnamed Policies** can be found under **Additional Features**. Here you can toggle the requirement on and off.

CLI

To change the requirement in the CLI, use the following syntax:

```
config system settings
  set gui-advance policy [enable|disable]
end
```

IPv4 policy

To configure a IPv4 policy in the GUI

1. Go to **Policy & Objects > IPv4 Policy**

The right side window will display a table of the existing IPv4 Policies.

- To edit an existing policy, double click on the policy you wish to edit
- To create a new policy, select the **Create New** icon in the top left side of the right window.

2. Make sure the policy has a name in the **Name** field



By default, a policy is required to have a name, but it is possible to toggle this requirement on or off in the CLI, or in the GUI if you have first enabled the GUI option in the CLI.

3. Set the **Incoming Interface** parameter by selecting the field with the "+" next to the field label. Selecting the field will slide out a window from the right where you can select from the available interfaces. You can select one or more specific interfaces or you can select the **any** option. Choosing the any option will remove any other interfaces. For more information on interfaces, check the Concepts section called **Interfaces and Zones**.
4. Set the **Outgoing Interface** parameter by selecting the field with the "+" next to the field label. (Same rules apply as with the above step.)



Alias names for interfaces, if used, appear in the headings for the Interface Pair View or what used to be called the Section View.

Multiple interfaces or ANY interface can be added to a firewall policy. This feature can be enabled or disabled in the GUI by going to the **System > Feature Select** page and toggling **Multiple Interface Policies**.

When selecting the Incoming or Outgoing interface of a policy, there are a few choices:

- The **ANY** interface (choosing this will remove all other interfaces)
- 1 A single specific interface
- 1 multiple specific interfaces (can be added at the same time or one at a time)



The GUI is intuitive and straightforward on how to do this. Click on the "+" symbol in the interface field and then select the desired interfaces from the side menu. There are a couple of ways to do it in the CLI:

1. Set the interfaces all at once:

```
config firewall policy
  edit 0
    set srcintf wan1 wan2
  end
```

2. Set the first interface and append additional ones:

```
config firewall policy
  edit 0
    set srcintf wan1
    append srcintf wan2
  end
```

5. Set the **Source** parameter by selecting the field with the "+" next to the field label. The source in this case is either the source address, source user or source device of the initiating traffic. When the field is selected a window will slide out from the right. Tabs indicating **Address**, **User** or **Device** options are there to help categorize the options along with the option to search. In order to be able to select one of these options it needs to be configured as a firewall object before hand. The "+" icon next to the **Search** field is a shortcut for creating a new firewall object based on the tab that is currently selected. For the **Address** and **Device** tabs, single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
6. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. This field is similar to the **Source** field but address objects are the only available options to select. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
7. Set the **Schedule** parameter by using the drop down menu to select a preconfigured schedule. The "+" icon next to the **Search** field is a shortcut for creating a new schedule object. For more information on addresses, check the Firewall Objects section called [Firewall schedules](#).
8. Set the **Service** parameter by selecting the field with the "+" next to the field label. (Same mechanics for selection apply as with the other similar fields in this window.) Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
9. Set the **Action** parameter. Select one of the following options for the action:

- **ACCEPT** - lets the traffic through to the next phase of analysis
- **DENY** - drops the session
- **LEARN** - collects information about the traffic for future analysis
- **IPsec** - for using with IPsec tunnels

Because the choice of **Action** determines the settings and options below this parameter in the window the rest of the step are associated with a specific **Action**.

Settings if the ACCEPT action is selected.

Firewall / Network Options

10. Set the **NAT** parameter by toggling the slider button.(gray means it is disabled)

The NAT setting section is affected by whether or not Central NAT is enabled.

If Central NAT is enabled, the only option in Firewall / Network options will be whether to enable or disable NAT. The rest of the NAT parameters will be set in the Central SNAT page.

If Central NAT is disabled, there are two additional settings in the Policy configuration page.

11. Set the **Fixed Port** parameter by toggling the slider button.(gray means it is disabled)

12. Set the **IP Pool Configuration** by selection one of the options of:

- **Use Outgoing Interface Address**
- **Use Dynamic IP Pool**

If the **Use Dynamic IP Pool** option is selected, an additional field will appear with the **+** icon. Selecting this field will slide out a window from the right where a preexisting IP Pool can be chosen. One or more IP Pools can be chosen and the **"+"** icon next to the **Search** field is a shortcut for creating a new IP Pool.

Security Profiles

13. Enabling the **Use Security Profile Group** option will allow the selection of a profile group instead of selecting the individual profiles for the policy.
14. Disable or enable the various **Security Profiles**. Once a Profile has been toggled into the enabled mode a drop down menu will appear for the purpose of choosing a specific profile. Only one profile can be chosen for each profile type. The **"+"** icon next to the **Search** field in the drop down menu is a shortcut for creating a new profile.

The list of **Security Profiles** available to set includes:

- **AntiVirus**
- **Web Filter**
- **DNS Filter**
- **Application Control**
- **CASI**
- **IPS**
- **Anti-Spam**
- **DLP Sensor**
- **VoIP**
- **ICAP**
- **Web Application Firewall**
- **Proxy Options**
- **SSL/SSH Inspection**

Logging Options

15. Set the **Log Allowed Traffic** parameter by toggling the slider button (gray means it is disabled). If the **Log Allowed Traffic** setting is enabled, choose whether to log just **Security Events** or **All Sessions** and determine whether or not to keep a record of the packets by toggling the **Capture Packets** setting on or off.
16. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
17. Toggle whether or not to **Enable this policy**. The default is enabled.
18. Select the **OK** button to save the policy.

Settings if the DENY action is selected

Enable the **Log Violation Traffic** setting by toggling the slider button.

10. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
11. Toggle whether or not to **Enable this policy**. The default is enabled.
12. Select the **OK** button to save the policy.

Settings if the LEARN action is selected

To get more information on the **LEARN** option, read the Learning mode for Firewall policies topic in [What's new for Firewall in 6.0](#)

Firewall / Network Options

10. Set the **NAT** parameter by toggling the slider button. (gray means it is disabled). Unlike the **ACCEPT** option, whether or not Central NAT is enabled or disabled does not affect this settings options.
11. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
12. Toggle whether or not to **Enable this policy**. The default is enabled.
13. Select the **OK** button to save the policy.

Settings if the IPsec action is selected

VPN Tunnel

10. For the VPN Tunnel field, use the drop down menu to select the VPN tunnel that you want the policy associated with.
11. Toggle the sliding button to enable or disable the option to **Allow traffic to be initiated from the remote site**

Security Profiles

12. Disable or enable the various **Security Profiles**. Once a Profile has been toggled into the enabled mode a drop down menu will appear for the purpose of choosing a specific profile. Only one profile can be chosen for each profile type. The "+" icon next to the **Search** field in the drop down menu is a shortcut for creating a new profile.

The list of **Security Profiles** available to set includes:

- **AntiVirus**
- **Web Filter**
- **DNS Filter**
- **Application Control**
- **CASI**
- **IPS**

- **Anti-Spam**
- **DLP Sensor**
- **VoIP**
- **ICAP**
- **Web Application Firewall**
- **Proxy Options**
- **SSL/SSH Inspection**

Logging Options

13. Set the **Log Allowed Traffic** parameter by toggling the slider button (gray means it is disabled). If the **Log Allowed Traffic** setting is enabled, choose whether to log just **Security Events** or **All Sessions** and determine whether or not to keep a record of the packets by toggling the **Capture Packets** setting on or off.
14. Add a comment to give a detailed description of the policy in the **Comments** field (up to 1023 characters).
15. Toggle whether or not to **Enable this policy**. The default is enabled.
16. Select the **OK** button to save the policy.

ISDB and IRDB in firewall policies

The Internet Service Database (ISDB) and the IP Reputation Database (IRDB) provide similar functionality, so for ease of use, appear together in the GUI.

The screenshot shows the FortiGate 101E GUI interface. The top bar displays 'FortiGate 101E FG101E4Q17001217' and 'interim build0069'. The left sidebar has a tree structure with 'Policy & Objects' expanded, showing 'Internet Service Database' selected. The main content area shows a table with columns: Name, Reputation, Direction, Protocol, Port, and Number of Entries. Two entries are listed: 'Internet Service Database' (262 entries) and 'IP Reputation Database' (6 entries). The bottom left of the main area has a search bar and a 'Q' icon.

Use the contents of both, or either database as criteria for inclusion or exclusion in a firewall policy.

Use CLI to define the objects of the ISDB or IRDB objects as parameters within a policy is done within the CLI.

CLI Syntax

```
config firewall policy
```

```
edit <ID #>
  set internet-service-src {enable|disable}
  set internet-service-src-id <ID #>
  set internet-service-src-custom <name>
  set internet-service-src-negate {enable|disable}
end
```

CLI options

Option	Description
internet-service-src	Enables or disables the use of Internet Services source for this policy. If enabled, destination address and service are not used.
internet-service-src-id	Internet Service ID Examples: <ul style="list-style-type: none"> • 65536 Google-Others • 65537 Google-Web
internet-service-src-custom	Custom Internet Service name This custom name must already be configured.
internet-service-src-negate	Enables or disables the use of Internet Services in source for this policy. If enabled, <code>internet-service-src</code> specifies what the service must NOT be.



Similar settings are also used in Traffic Shaping policies.

IPv6 policy

To configure a IPv6 policy in the GUI

1. Go to **Policy & Objects > IPv6 Policy**

The right side window will display a table of the existing IPv6 Policies.

- To edit an existing policy, double click on the policy you wish to edit
- To create a new policy, select the **Create New** icon in the top left side of the right window.

2. Make sure the policy has a name in the **Name** field



By default, a policy is required to have a name, but it is possible to toggle this requirement on or off in the CLI, or in the GUI if you have first enabled the GUI option in the CLI.

3. Set the **Incoming Interface** parameter by selecting the field with the "+" next to the field label. Selecting the field will slide out a window from the right where you can select from the available interfaces. You can select one or more specific interfaces or you can select the **any** option. Choosing the any option will remove any other interfaces. For more information on interfaces, check the Concepts section called [Interfaces and Zones](#).
4. Set the **Outgoing Interface** parameter by selecting the field with the "+" next to the field label. (Same rules apply as with the above step.)
5. Set the **Source** parameter by selecting the field with the "+" next to the field label. The source in this case is either the source address, source user or source device of the initiating traffic. When the field is selected a window will slide out from the right. Tabs indicating **Address**, **User** or **Device** options are there to help categorize the options along with the option to search. In order to be able to select one of these options it needs to be configured as a firewall object before hand. The "+" icon next to the **Search** field is a shortcut for creating a new firewall object based on the tab that is currently selected. For the **Address** and **Device** tabs, single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
6. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. This field is similar to the **Source** field but address objects are the only available options to select. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
7. Set the **Schedule** parameter by using the drop down menu to select a preconfigured schedule. The "+" icon next to the **Search** field is a shortcut for creating a new schedule object. For more information on addresses, check the Firewall Objects section called [Firewall schedules](#)
8. Set the **Service** parameter by selecting the field with the "+" next to the field label. (Same mechanics for selection apply as with the other similar fields in this window.) Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
9. Set the **Action** parameter. Select one of the following options for the action:
 - **ACCEPT** - lets the traffic through to the next phase of analysis
 - **DENY** - drops the session

While there are not as many Action options as with the IPv4 policy, because the choice of **Action** determines the settings and options below this parameter in the window the rest of the step are associated with a specific **Action**.

Settings if the **ACCEPT** action is selected.

Firewall / Network Options

10. Set the **NAT** parameter by toggling the slider button.(gray means it is disabled)
11. Set the **IP Pool Configuration** by selection one of the options of:
 - **Use Outgoing Interface Address**
 - **Use Dynamic IP Pool**

If the **Use Dynamic IP Pool** option is selected:

- An additional field will appear with the "+" icon. Selecting this field will slide out a window from the right where a preexisting IP Pool can be chosen. One or more IP Pools can be chosen and the "+" icon next to the **Search** field is a shortcut for creating a new IP Pool.
- An additional option to **Preserve the Source Port** will appear as a toggle option. If the slider button is grayed out it is disabled.

Security Profiles

12. Enabling the **Use Security Profile Group** option will allow the selection of a profile group instead of selecting the individual profiles for the policy.
13. Disable or enable the various **Security Profiles**. Once a Profile has been toggled into the enabled mode a drop down menu will appear for the purpose of choosing a specific profile. Only one profile can be chosen for each profile type. The "+" icon next to the **Search** field in the drop down menu is a shortcut for creating a new profile.

The list of **Security Profiles** available to set includes:

- AntiVirus
- Web Filter
- Application Control
- IPS
- Anti-Spam
- DLP Sensor
- VoIP
- ICAP

Logging Options

14. Set the **Log Allowed Traffic** parameter by toggling the slider button (gray means it is disabled). If the **Log Allowed Traffic** setting is enabled, choose whether to log just **Security Events** or **All Sessions** and determine whether or not to keep a record of the packets by toggling the **Capture Packets** setting on or off.
15. Add a comment to give a detailed description of the policy in the **Comments** field (up to 1023 characters).
16. Toggle whether or not to **Enable this policy**. The default is enabled.
17. Select the **OK** button to save the policy.

Settings if the DENY action is selected

Enable the **Log Violation Traffic** setting by toggling the slider button.

10. Add a comment to give a detailed description of the policy in the **Comments** field (up to 1023 characters).
11. Toggle whether or not to **Enable this policy**. The default is enabled.
12. Select the **OK** button to save the policy.

NAT64 policy

To configure a NAT64 policy in the GUI

1. Go to **Policy & Objects > NAT64 Policy**

The right side window will display a table of the existing NAT64 Policies.

- To edit an existing policy, double click on the policy you wish to edit
- To create a new policy, select the **Create New** icon in the top left side of the right window.

2. Set the **Incoming Interface** parameter by selecting the field with the "+" next to the field label. Selecting the field will slide out a window from the right where you can select from the available interfaces. You can select one or more specific interfaces or you can select the **any** option. Choosing the any option will remove any other interfaces. For more information on interfaces, check the Concepts section called [Interfaces and Zones](#).

3. Set the **Outgoing Interface** parameter by selecting the field with the "+" next to the field label. (Same rules apply as with the above step.)
4. Set the **Source Address** parameter by selecting the field with the "+" next to the field label. The source in this case is an IPv6 Address object of the initiating traffic. When the field is selected a window will slide out from the right. In order to be able to select one of these options it needs to be configured as a firewall object before hand. The "+" icon next to the **Search** field is a shortcut for creating a new firewall object based on the tab that is currently selected. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
5. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. This field is similar to the **Source Address** field. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
6. Set the **Schedule** parameter by using the drop down menu to select a preconfigured schedule. The "+" icon next to the **Search** field is a shortcut for creating a new schedule object. For more information on addresses, check the Firewall Objects section called [Firewall schedules](#)
7. Set the **Service** parameter by selecting the field with the "+" next to the field label. (Same mechanics for selection apply as with the other similar fields in this window.) Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
8. Set the **Action** parameter. Select one of the following options for the action:
 - **ACCEPT** - lets the traffic through to the next phase of analysis
 - **DENY** - drops the session

While there are not as many Action options as with the IPv4 policy, because the choice of **Action** determines the settings and options below this parameter in the window the rest of the step are associated with a specific **Action**.

Settings if the ACCEPT action is selected.

Firewall / Network Options

10. Skip the NAT setting. This type of policy is intended only for traffic that is being NATed from IPv6 to IPv4, because without NATing the traffic couldn't reach its destination, so disabling NAT would be pointless.
11. Set the **IP Pool Configuration** by selection one of the options:
 - **Use Outgoing Interface Address**
 - **Use Dynamic IP Pool**

If the **Use Dynamic IP Pool** option is selected, an additional field will appear with the + icon. Selecting this field will slide out a window from the right where a preexisting IP Pool can be chosen. One or more IP Pools can be chosen and the "+" icon next to the **Search** field is a shortcut for creating a new IP Pool.

12. Set the **Log Allowed Traffic** parameter by toggling the slider button (gray means it is disabled).
If the **Log Allowed Traffic** setting is enabled, choose whether to log just **Security Events** or **All Sessions** and determine whether or not to keep a record of the packets by toggling the **Capture Packets** setting on or off.
13. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
14. Toggle whether or not to **Enable this policy**. The default is enabled.
15. Select the **OK** button to save the policy.

Settings if the DENY action is selected

Enable the Log Violation Traffic setting by toggling the slider button.

10. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
11. Toggle whether or not to **Enable this policy**. The default is enabled.
12. Select the **OK** button to save the policy.

NAT46 policy

To configure a NAT46 policy in the GUI

1. Go to Policy & Objects > NAT46 Policy

The right side window will display a table of the existing NAT46 Policies.

- To edit an existing policy, double click on the policy you wish to edit
 - To create a new policy, select the **Create New** icon in the top left side of the right window.
2. Set the **Incoming Interface** parameter by selecting the field with the "+" next to the field label. Selecting the field will slide out a window from the right where you can select from the available interfaces. You can select one or more specific interfaces or you can select the **any** option. Choosing the any option will remove any other interfaces. For more information on interfaces, check the Concepts section called [Interfaces and Zones](#).
 3. Set the **Outgoing Interface** parameter by selecting the field with the "+" next to the field label. (Same rules apply as with the above step.)
 4. Set the **Source** parameter by selecting the field with the "+" next to the field label. The source in this case is either the source address, source user or source device of the initiating traffic. When the field is selected a window will slide out from the right. Tabs indicating **Address**, **User** or **Device** options are there to help categorize the options along with the option to search. In order to be able to select one of these options it needs to be configured as a firewall object before hand. The "+" icon next to the **Search** field is a shortcut for creating a new firewall object based on the tab that is currently selected. For the **Address** and **Device** tabs, single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
 5. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. This field is similar to the **Source** field but address objects are the only available options to select. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
 6. Set the **Schedule** parameter by using the drop down menu to select a preconfigured schedule. The "+" icon next to the **Search** field is a shortcut for creating a new schedule object. For more information on addresses, check the Firewall Objects section called [Firewall schedules](#).
 7. Set the **Service** parameter by selecting the field with the "+" next to the field label. (Same mechanics for selection apply as with the other similar fields in this window.) Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
 8. Set the **Action** parameter. Select one of the following options for the action:
 - **ACCEPT** - lets the traffic through to the next phase of analysis
 - **DENY** - drops the session

While there are not as many Action options as with the IPv4 policy, because the choice of **Action** determines the settings and options below this parameter in the window the rest of the step are associated with a specific **Action**.

Settings if the ACCEPT action is selected.

Firewall / Network Options

10. Skip the NAT setting. This type of policy is intended only for traffic that is being NATed from IPv4 to IPv6, because without NATing the traffic couldn't reach its destination, so disabling NAT would be pointless.
11. Set the **Fixed Port** parameter by toggling the slider button.(gray means it is disabled)
12. Set the **IP Pool Configuration** by selection one of the options of:
 - **Use Outgoing Interface Address**
 - **Use Dynamic IP Pool**If the **Use Dynamic IP Pool** option is selected, an additional field will appear with the **+** icon. Selecting this field will slide out a window from the right where a preexisting IP Pool can be chosen. One or more IP Pools can be chosen and the **+** icon next to the **Search** field is a shortcut for creating a new IP Pool.
14. Set the **Log Allowed Traffic** parameter by toggling the slider button (gray means it is disabled).If the **Log Allowed Traffic** setting is enabled, choose whether to log just **Security Events** or **All Sessions** and determine whether or not to keep a record of the packets by toggling the **Capture Packets** setting on or off.
15. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
16. Toggle whether or not to **Enable this policy**.The default is enabled.
17. Select the **OK** button to save the policy.

Settings if the DENY action is selected

Enable the Log Violation Traffic setting by toggling the slider button.

10. Add a comment to give a detailed description of the policy in the Comments field (up to 1023 characters).
11. Toggle whether or not to **Enable this policy**.The default is enabled.
12. Select the **OK** button to save the policy.

Central SNAT

The Central NAT feature is not enabled by default. When `central-nat` is enabled, `nat` option under IPv4 policies is skipped and SNAT must be done via `central-snat-map`.

- Info messages and redirection links have been added to IPv4 policy list and dialog to indicate the above
- If NGFW mode is policy-based, then it is assumed that `central-nat` (specifically SNAT) is enabled implicitly
- The option to toggle NAT in `central-snat-map` policies has been added (previously it was only shown in NGFW policy-based mode).
- In `central-snat` policy dialog, the port-mapping fields for the original port have been updated to accept ranges.
- Nat will be skipped in firewall policy if per vdom central nat is enabled.
- The Central SNAT window contains a table of all of the Central SNAT policies.

To toggle the feature on or off, use the following commands:

```
config system settings
    set central-nat [enable | disable]
    end
```

When Central NAT is enable the **Central SNAT** section will appear under the Policy & Objects heading in the GUI.

To configure a Central SNAT entry in the GUI

1. Go to Policy & Objects > Central SNAT

The right side window will display a table of the existing Central SNAT entries.

- To edit an existing entry, double click on the policy you wish to edit
 - To create a new entry, select the **Create New** icon in the top left side of the right window.
2. Set the **Incoming Interface(s)** by clicking on the "+" in the field. This will slide out a window from the right. Here, you can select from the available interfaces. Selecting a listed interface will highlight it in the window and add it to the field. Clicking on an object in this window while it's highlighted will remove it from the field. Multiple selections are allowed.
3. Set the **Outgoing Interface(s)** by clicking on the "+" in the field. This will slide out a window from the right. Here, you can select from the available interfaces. Selecting a listed interface will highlight it in the window and add it to the field. Clicking on an object in this window while it's highlighted will remove it from the field. Multiple selections are allowed.
4. Set the **Source Address** by clicking on the "+" in the field. This will slide out a window from the right. Here, you can select from the available address objects. Selecting a listed object will highlight it in the window and add it to the field. Clicking on an object in this window while it's highlighted will remove it from the field. Multiple selections are allowed. For more information on addresses, check the Firewall Objects section called [Addresses](#).
5. Set the **Destination Address** by clicking on the "+" in the field. This will slide out a window from the right. Here, you can select from the available address objects. Selecting a listed object will highlight it in the window and add it to the field. Clicking on an object in this window while it's highlighted will remove it from the field. Multiple selections are allowed.

Under the NAT Heading

6. Set the **IP Pool Configuration** parameter by selecting either **Use Outgoing Interface Address** or **Use Dynamic IP Pool**.

- If Use Dynamic IP Pool is chosen, a field will appear just beneath the option that is used to select which IP Pool object will be used. Set the IP Pool by clicking on the "+" in the field. This will slide out a window from the right. Here, you can select from the available objects.

7. Set the **Protocol** parameter.

There are 5 options for the **Protocol**.

- **ANY** - any protocol traffic
- **TCP** - TCP traffic only. Protocol number set to 6
- **UDP** - UDP traffic only . Protocol number set to 17
- **SCTP** - SCTP traffic only. Protocol number set to 132
- **Specify** - User can specify the traffic filter protocol by setting the protocol number in the field.

6. If the IP Pool is of the type: Overload, **Explicit Port Mapping** can be enabled.

To enable or disable, use the check box. Once enabled, the following additional parameters will appear.

- **Original Source Port** - in the left number field, set the starting number of the source port range.
 - **Translated Port** - in the left number field, set the starting number of the translated port range. If it is a single port range leave the right number field alone. If the right number field is set to a number higher than the left, the right number field for the Original Source Port will change to make sure the 2 number ranges have a matching number of ports.
7. Select the **OK** button to save the entry.

To configure Central SNAT in the CLI

1. Using the CLI interface of your choice, run the following command to get to the correct context.

```
config firewall central-snat-map
```

- To edit an existing entry, run the command `show` or `show full-configuration` to get a listing of all of the entries in the map. Take note of the policy ID for the entry to be edited.
- To create a new entry the next step will use the policy ID 0 which will check for an unused ID number and create an entry with that number.

2. Edit or create an entry with the correct policy ID

```
edit <policyID number>
```

Run the following commands to set the parameters of the entry:

```
set status [enable|disable]
set orig-addr <valid address object preconfigured on the FortiGate>
set srcintf <name of interface on the FortiGate>
set dst-addr <valid address object preconfigured on the FortiGate>
set dstintf <name of interface on the FortiGate>
set protocol <integer for protocol number>
set orig-port <integer for original port number>
set nat-port <integer for translated port number>
set comments <string>
```

3. Save the entry by running the command `end` or `next`.

Example scenarios to showing how CLI treats central-nat

Make nat available regardless of NGFW mode.

```
config firewall central-snat-map
  edit 1
    set orig-addr "192-86-1-86"
    set srcintf "port23"
    set dst-addr "192-96-1-96"
    set dstintf "port22"
    set nat-ippool "pool1"
    set protocol 17
    set orig-port 2896-2897
    set nat enable
  end
```

Hide nat-port if nat-ippool is not set or NAT is disabled.

```
config firewall central-snat-map
  edit 1
```

```
set orig-addr "192-86-1-86"
set srcintf "port23"
set dst-addr "192-96-1-96"
set dstintf "port22"
set nat-ippool "pool1"
set protocol 17
set orig-port 2896-2897
set nat disable
end
```

Change orig-port to accept range

```
config firewall central-snat-map
  edit 1
    set orig-addr "192-86-1-86"
    set srcintf "port23"
    set dst-addr "192-96-1-96"
    set dstintf "port22"
    set nat-ippool "pool1"
    set protocol 17
    set orig-port 2896-2897 (help text changed to: Original port or port range).
    set nat-port 35804-35805
  end
```

IPv4 access control list

The **IPv4 Access Control List** is a specialized policy for denying IPv4 traffic based on:

- the incoming interface
- the source addresses of the traffic
- the destination addresses of the traffic
- the services or ports the traffic is using

The only action available in this policy is **DENY**.

For more information, see [Access control lists](#).

To configure a IPv4 access control list entry in the GUI

1. Go to **Policy & Objects > IPv4 Access Control List**

The right side window will display a table of the existing IPv4 Access Control List entries.

- To edit an existing entry, double click on the policy you wish to edit
- To create a new entry, select the **Create New** icon in the top left side of the right window.

2. Set the **Incoming Interface** parameter by using the drop down menu to select a single interface.

3. Set the **Source Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).

4. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.

5. Set the **Services** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
6. Toggle whether or not to **Enable this policy**. The default is enabled.
7. Select the **OK** button to save the policy.

To configure a IPv4 access control list entry in the CLI

Use the following syntax:

```
config firewall acl
    edit <acl Policy ID #>
        set status enable
        set interface <interface>
        set srcaddr <address object>
        set dstaddr <address object>
        set service <service object>
    end
end
```

IPv6 access control list

The **IPv6 Access Control List** is a specialized policy for denying IPv6 traffic based on:

- the incoming interface
- the source addresses of the traffic
- the destination addresses of the traffic
- the services or ports the traffic is using

The only action available in this policy is **DENY**

To configure a IPv6 access control list entry in the GUI

1. Go to **Policy & Objects > IPv6 Access Control List**

The right side window will display a table of the existing IPv6 Access Control List entries.

- To edit an existing entry, double click on the policy you wish to edit
 - To create a new entry, select the **Create New** icon in the top left side of the right window.
2. Set the **Incoming Interface** parameter by using the drop down menu to select a single interface.
 3. Set the **Source IPv6 Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
 4. Set the **Destination IPv6 Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
 5. Set the **Services** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
 6. Toggle whether or not to **Enable this policy**. The default is enabled.
 7. Select the **OK** button to save the policy.

To configure a IPv6 access control list entry in the CLI

Use the following syntax:

```
config firewall acl6
    edit <acl Policy ID #>
        set status enable
        set interface <interface>
        set srcaddr <address object>
        set dstaddr <address object>
        set service <service object>
    end
end
```

IPv4 DoS policy

To configure a IPv4 DoS policy in the GUI

1. Go to **Policy & Objects > IPv4 DoS Policy**

The right side window will display a table of the existing IPv4 DoS Policies.

- To edit an existing policy, double click on the policy you wish to edit
 - To create a new policy, select the **Create New** icon in the top left side of the right window.
2. Set the **Incoming Interface** parameter by using the drop down menu to select a single interface.
 3. Set the **Source Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
 4. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
 5. Set the **Services** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
 6. Set the parameters for the various traffic anomalies.

All of the anomalies that profiles have been created for are in 2 tables. These tables break up the anomaly profiles into **L3 Anomalies** and **L4 Anomalies**. All of the anomalies have the following parameters that can be set on a per anomaly or per column basis.

- Status - enable or disable the indicated profile
- Logging - enable or disable logging of the indicated profile being triggered
- Action - whether to Pass or Block traffic when the threshold is reached
- Threshold - the number of anomalous packets detected before triggering the action.

The listing of anomaly profiles includes:

L3 Anomalies

- ip_src_session
- ip_dst_session

L4 Anomalies

- tcp_syn_flood
- tcp_port_scan
- tcp_src_session
- tcp_dst_session
- udp_flood
- udp_scan
- udp_src_session
- udp_dst_session
- icmp_flood
- icmp_sweep
- icmp_src_session
- sctp_flood
- sctp_scan
- sctp_src_session
- sctp_dst_session

7. Toggle whether or not to **Enable this policy**. The default is enabled.
8. Select the **OK** button to save the policy.

Example

The company wishes to protect against Denial of Service attack. They have chosen some where they wish to block the attacks of the incidence goes above a certain threshold and for some others they are just trying to get a baseline of activity for those types of attacks so they are letting the traffic pass through without action.

- The interface to the Internet is on WAN1
- There is no requirement to specify which addresses are being protected or protected from.
- The protection is to extend to all services.
- The TCP attacks are to be blocked
- The UDP, ICMP, and IP attacks are to be recorded but not blocked.
- The SCTP attack filters are disabled
- The tcp_syn_flood attack's threshold is to be changed from the default to 1000

Configuring the DoS policy in the GUI

1. Go to **Policy & Objects > Policy > DoS**.
2. Create a new policy
3. Fill out the fields with the following information:

Field	Value
Incoming Interface	wan1
Source Address	all
Destination Addresses	all
Service	ALL

L3 Anomalies

Name	Status	Logging	Action	Threshold
ip_src_session	enabled	enabled	Pass	5000
ip_dst_session	enabled	enabled	Pass	5000

L4 Anomalies

Name	Status	Logging	Action	Threshold
tcp_syn_flood	enabled	enabled	Block	1000
tcp_port_scan	enabled	enabled	Block	<default value>
tcp_src_session	enabled	enabled	Block	<default value>
tcp_dst_session	enabled	enabled	Block	<default value>
udp_flood	enabled	enabled	Pass	<default value>
udp_scan	enabled	enabled	Pass	<default value>
udp_src_session	enabled	enabled	Pass	<default value>
udp_dst_session	enabled	enabled	Pass	<default value>
icmp_flood	enabled	enabled	Pass	<default value>
icmp_sweep	enabled	enabled	Pass	<default value>
icmp_src_session	enabled	enabled	Pass	<default value>
icmp_dst_session	enabled	enabled	Pass	<default value>
sctp_flood	not enabled	not enabled	Pass	<default value>
sctp_scan	not enabled	not enabled	Pass	<default value>
sctp_src_session	not enabled	not enabled	Pass	<default value>
sctp_dst_session	not enabled	not enabled	Pass	<default value>

4. Toggle the button next to **Enable this policy** to **ON**.
5. Select **OK**.

Configuring the IPv4 DoS policy in the GUI

Using the CLI of your choice, enter the following commands:

```
config firewall DoS-policy
  edit 0
    set status enable
    set interface wan1
    set srcaddr all
    set dstaddr all
    set service ALL
    config anomaly
      edit "tcp_syn_flood"
        set status enable
        set log disable
        set action block
        set threshold 1000
      next
      edit "tcp_port_scan"
        set status enable
        set log disable
        set action block
        set threshold 1000
      next
      edit "tcp_src_session"
        set status enable
        set log disable
        set action block
        set threshold 5000
      next
      edit "tcp_dst_session"
        set status enable
        set log disable
        set action block
        set threshold 5000
      next
      edit "udp_flood"
        set status enable
        set log disable
        set action pass
        set threshold 2000
      next
      edit "udp_scan"
        set status enable
        set log disable
        set action pass
        set quarantine none
        set threshold 2000
      next
      edit "udp_src_session"
        set status enable
        set log disable
        set action pass
        set threshold 5000
      next
      edit "udp_dst_session"
        set status enable
        set log disable
```

```
        set action pass
        set threshold 5000
        next
    edit "icmp_flood"
        set status enable
        set log disable
        set action pass
        set threshold 250
        next
    edit "icmp_sweep"
        set status enable
        set log disable
        set action pass
        set threshold 100
        next
    edit "icmp_src_session"
        set status enable
        set log disable
        set action pass
        set threshold 300
        next
    edit "icmp_dst_session"
        set status enable
        set log disable
        set action pass
        set threshold 1000
        next
    edit "ip_src_session"
        set status disable
        set log enable
        set action pass
        set threshold 5000
        next
    edit "ip_dst_session"
        set status disable
        set log enable
        set action pass
        set threshold 5000
        next
    edit "sctp_flood"
        set status disable
        set log disable
        set action pass
        set threshold 2000
        next
    edit "sctp_scan"
        set status disable
        set log disable
        set action pass
        set threshold 1000
        next
    edit "sctp_src_session"
        set status disable
        set log disable
        set action pass
        set threshold 5000
        next
```

```
edit "sctp_dst_session"
  set status disable
  set log disable
  set action pass
  set threshold 5000
  next
end
end
end
```



In this example of the CLI, all of the relevant settings have been left in, but some of them are default settings and would not have to have been specifically set to work. For instance, if the action parameter is not set it automatically defaults to pass.

IPv6 DoS policy

To configure a IPv6 DoS policy in the GUI

1. Go to **Policy & Objects > IPv6 DoS Policy**

The right side window will display a table of the existing IPv6 DoS Policies.

- To edit an existing policy, double click on the policy you wish to edit
 - To create a new policy, select the **Create New** icon in the top left side of the right window.
2. Set the **Incoming Interface** parameter by using the drop down menu to select a single interface.
 3. Set the **Source IPv6 Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option. For more information on addresses, check the Firewall Objects section called [Addresses](#).
 4. Set the **Destination IPv6 Address** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
 5. Set the **Services** parameter by selecting the field with the "+" next to the field label. Single or multiple options can be selected unless the **ALL** option is chosen in which case, it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).
 6. Set the parameters for the various traffic anomalies.

All of the anomalies that profiles have been created for are in 2 tables. These tables break up the anomaly profiles into **L3 Anomalies** and **L4 Anomalies**. All of the anomalies have the following parameters that can be set on a per anomaly or per column basis.

- Status - enable or disable the indicated profile
- Logging - enable or disable logging of the indicated profile being triggered
- Action - whether to Pass or Block traffic when the threshold is reached
- Threshold - the number of anomalous packets detected before triggering the action.

The listing of anomaly profiles includes:

L3 Anomalies

- ip_src_session
- ip_dst_session

L4 Anomalies

- tcp_syn_flood
- tcp_port_scan
- tcp_src_session
- tcp_dst_session
- udp_flood
- udp_scan
- udp_src_session
- udp_dst_session
- icmp_flood
- icmp_sweep
- icmp_src_session
- icmp_dst_session
- sctp_flood
- sctp_scan

7. Toggle whether or not to **Enable this policy**. The default is enabled.
8. Select the **OK** button to save the policy.

Configuring the IPv6 DoS policy in the GUI

The configuring of the IPv6 version of the DoS policy is the same as in the IPv4 version , with the exception of first command.

Using the CLI of your choice, enter the following commands:

```
config firewall DoS-policy6
```

The rest of the settings are the same as in IPv4 Dos Policy.

Multicast policy

The **Multicast Policy** GUI page has been updated from previous versions of the firmware to the new GUI look and feel. Some functionality has also been changed.

The DNAT option has been removed from the GUI but is still in the CLI.

To create/edit a multicast policy go to **Policy & Objects > Multicast Policy**. The Listing window on the right will have buttons along the top that will enable you to

- **Create New**
- **Edit**
- **Delete**

There is also a **Search** field that will allow you to search or filter the available policies if you have a lot of them.

To configure a new policy left click on the **Create New** button. This will reveal the New Policy editing window.

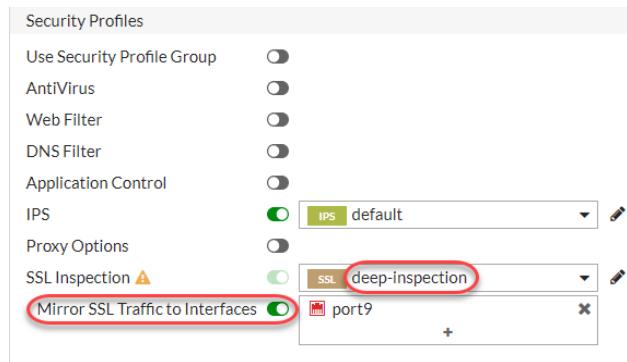
1. Using the drop down menu, fill in the field for **Incoming Interface**. Only one interface can be chosen.
2. Using the drop down menu, fill in the field for **Outgoing Interface**. Only one interface can be chosen.

3. Set the **Source Address** parameter by selecting the field with the "+" next to the field label. When the field is selected a window will slide out from the right. In order for a multicast address to be available for selection, the address object needs to have been created already. Only useable address options will be available for selection. This means only multicast address objects and the more generic **all** and **none** options. The "+" icon next to the **Search** field is a shortcut for creating a new firewall object. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
4. Set the **Destination Address** parameter by selecting the field with the "+" next to the field label. This field is similar to the **Source** field. Single or multiple options can be selected unless the **all** option is chosen in which case, it will be the only option.
5. Set the **Action** parameter. This will be to either **ACCEPT** or **DENY** the traffic through the policy.
6. Toggle the **Enable SNAT** switch to the setting you want. If the slider is gray the option is disabled. If it is colored, it is enabled.
7. Use the drop down menu to select a **Protocol**. The options are:
 - **Any**
 - **ICMP**
 - **IGMP**
 - **TCP** - includes **Port Range** fields
 - **UDP** - includes **Port Range** fields
 - **OSPF**
 - **Other** - includes a field for the protocol number
8. Depending on which Protocol is defined, the some other fields may appear.
 - **Port Range** - The first field is for the starting value for the port and the second for the ending value for the port range used by the protocol. Both of these values are inclusive.
 - Protocol field - This appears when the **Other** option is chosen. Enter the value of the protocol number for the protocol you wish to use.
9. Toggle the **Log Allowed Traffic** switch to the setting you want. If the slider is gray the option is disabled. If it is colored, it is enabled.
10. Toggle the **Enable this policy** switch to the setting you want. If the slider is gray the option is disabled. If it is colored, it is enabled. By default, this should be enabled
11. Click on the **OK** button to save the policy.

SSL mirroring for policies

You can configure the mirroring of SSL inspected traffic for IPv4 and IPv6 policies in the GUI. To use this option, configure the FortiGate to use SSL deep inspection instead of certificate inspection.

SSL inspection is automatically enabled when you enable a security profile in the policy configuration page. Selecting an SSL inspection profile that uses deep inspection reveals the **Mirror SSL Traffic to Interfaces** option. If you enable mirroring, an **SSL Mirror Terms of Use** agreement appears. Once you accept the agreement, you can select interfaces to mirror the traffic to in the Mirror SSL Traffic to Interfaces field.



Object configuration

As was mentioned earlier, the components of the FortiGate firewall go together like interlocking building blocks. The Firewall objects are a prime example of those building blocks. They are something that can be configured once and then used over and over again to build what you need. They can assist in making the administration of the FortiGate unit easier and more intuitive as well as easier to change. By configuring these objects with their future use in mind as well as building in accurate descriptions the firewall will become almost self documenting. That way, months later when a situation changes, you can take a look at a policy that needs to change and use a different firewall object to adapt to the new situation rather than build everything new from the ground up to accommodate the change.

This chapter includes information about the following Firewall objects:

- Addresses
- "Virtual IPs" on page 234
- IP Pools
- "Services" on page 248
- "Firewall schedules" on page 256

UUID support

A Universally Unique Identified (UUID) attribute has been added to some firewall objects, so that the logs can record these UUID to be used by a FortiManager or FortiAnalyzer unit. The objects currently include:

- Addresses, both IPv4 and IPv6
- Address Groups, both IPv4 and IPv6
- Virtual IPs, both IPv4 and IPv6
- Virtual IP groups, both IPv4 and IPv6
- Policies, IPv4,IPv6 and IP64

A UUID is a 16-octet (128-bit) number that is represented by 32 lowercase hexadecimal digits. The digits are displayed in five groups separated by hyphens (-). The pattern is 8-4-4-4-12; 36 digits if you include the hyphens.



Note: UUID is only supported on large-partition platforms (>=128M)

Addresses

Firewall addresses define sources and destinations of network traffic and are used when creating policies. When properly set up these firewall objects can be used with great flexibility to make the configuration of firewall policies simpler and more intuitive. The FortiGate unit compares the IP addresses contained in packet headers with a security policy's source and destination addresses to determine if the security policy matches the traffic.

The address categories and the types within those categories on the FortiGate unit can include:

- IPv4 addresses
 - IP address and Netmask
 - IP address range
 - Geography based address
 - Fully Qualified Domain Name (FQDN) address
 - Wildcard FQDN
 - IPv4 address group
- IPv6 addresses
 - Subnets
 - IP range
 - IPv6 address group
- Multicast addresses
 - Multicast IP range
 - Broadcast subnets
- Proxy addresses
 - URL pattern
 - Host Regex match
 - URL category
 - Http method
 - User agent
 - HTTP header
 - Advanced (source)
 - Advanced (destination)
- IP Pools (IPv4)
 - Overload
 - One-to-one
 - Fixed port range
 - Port block allocation
- IP pools (IPv6)
- Virtual IP addresses
 - IPv4
 - IPv6
 - NAT46
 - NAT64

Interfaces

When setting up an address one of the parameters that is asked for is the interface. This means that the system will expect to see that address only on the interface that you select. You can only select one interface. If you expect that the address may be seen at more than one interface you can choose the “any” interface option. Whenever possible it is best to choose a more specific interface than the “any” option because in the GUI configuration of firewall policies there is a drop down field that will show the possible addresses that can be used. The drop down will only show those addresses that can be on the interface assigned for that interface in the policy.

Example:

- You have an address called "XYZ".
- "XYZ" is set to the WAN1 interface because that is the only interface that will be able to access that address.
- When you are selecting a Source Address in the Web-based Manager for a policy that is using the DMZ the address "XYZ" will not be in the drop-down menu.

When there are only 10 or 20 addresses this is not a concern, but if there are a few hundred addresses configured it can make your life easier.

Addresses, address groups, and virtual IPs must have unique names to avoid confusion in firewall policies. If an address is selected in a policy, the address cannot be deleted until it is deselected from the policy.

Addressing Best Practices Tip

The other reason to assign a specific interface to addresses is that it will prevent you from accidentally assigning an address where it will not work properly. Using the example from earlier, if the "XYZ" address was assigned to the "Any" interface instead of WAN1 and you configure the "XYZ" address.

Addressing Best Practices Tip

Don't specify an interface for VIP objects or other address objects that may need to be moved or approached from a different direction. When configuring a VIP you may think that it will only be associated with a single interface, but you may later find that you need to reference it on another interface.

Example: Some web applications require the use of a FQDN rather than an IP address. If you have a VIP set up that works from the Internet to the Internal LAN you won't be able to use that VIP object to access it from an internal LAN interface.

IPv4 addresses

When creating an IPv4 address there are a number of different types of addresses that can be specified. These include:

- FQDN
- Geography
- IP range
- IP/Netmask
- Wildcard FQDN

Which one chosen will depend on which method most easily yet accurately describes the addresses that you are trying to include with as few entries as possible based on the information that you have. For instance, if you are trying to describe the addresses of a specific company's web server but you have no idea of how extensive their web server farm is you would be more likely to use a Fully Qualified Domain Name (FQDN) rather than a specific IP address. On the other hand some computers don't have FQDNs and a specific IP address must be used.

The following is a more comprehensive description of the different types of addresses.

FQDN addresses

By using Fully Qualified Domain Name (FQDN) addressing you can take advantage of the dynamic ability of DNS to keep up with address changes without having to manually change the addresses on the FortiGate. FQDN addresses are most often used with external web sites but they can be used for internal web sites as well if there is a trusted DNS server that can be accessed. FQDN addressing also comes in handy for large web sites that may use multiple addresses and load balancers for their web sites. The FortiGate firewall automatically maintains a cached record of all the addresses resolved by the DNS for the FQDN addresses used.

For example, if you were doing this manually and you wanted to have a security policy that involved Google you could track down all of the IP addresses that they use across multiple countries. Using the FQDN address is simpler and more convenient.

When representing hosts by an FQDN, the domain name can also be a subdomain, such as mail.example.com.

Valid FQDN formats include:

- <host_name>.<top_level_domain_name> such as example.com
- <host_name>.<second_level_domain_name>.<top_level_domain_name>, such as mail.example.com

When creating FQDN entries it is important to remember that:

- Wildcards are not supported in FQDN address objects
- While there is a level of convention that would imply it, “www.example.com” is not necessarily the same address of “example.com”. they will each have their own records on the DNS server.

The FortiGate firewall keeps track of the DNS TTLs so as the entries change on the DNS servers the IP address will effectively be updated for the FortiGate. As long as the FQDN address is used in a security policy, it stores the address in the DNS cache.



There is a possible security downside to using FQDN addresses. Using a fully qualified domain name in a security policy means that your policies are relying on the DNS server to be accurate and correct. DNS servers in the past were not seen as potential targets because the thinking was that there was little of value on them and therefore are often not as well protected as some other network resources. People are becoming more aware that the value of the DNS server is that in many ways it controls where users and computers go on the Internet. Should the DNS server be compromised, security policies requiring domain name resolution may no longer function properly.

Creating a Fully Qualified Domain Name address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**.
3. In the **Category** field, chose **Address**. (This is for IPv4 addresses.)
4. Input a **Name** for the address object.
5. In the **Type** field, select **FQDN** from the drop down menu.
6. Input the domain name in the **FQDN** field.
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled, the address will appear in drop down menus where it is an option.

9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example: FQDN address

You have to create a policy that will govern traffic that goes to a site that has a number of servers on the Internet. Depending on the traffic or the possibility that one of the servers is down network traffic can go to any one of those sites. The consistent factor is that they all use the same Fully Qualified Domain Name.

- The FQDN of the web site: example.com
- The number of ISP connections off of the FortiGate firewall: 2

Configuring the address in the GUI

1. Go to **Policy & Objects > Objects > Addresses** and select **Create New > Address**.
2. Fill out the fields with the following information:

Category	Address
Name	BigWebsite.com
Type	FQDN
FQDN	bigwebsite.com
Interface	any
Show in Address List	<enable>
Comments	<Input into this field is optional>

3. Select **OK**.

Configuring the address in the CLI

```
config firewall address
    edit BigWebsite.com
        set type fqdn
        set associated-interface any
        set fqdn bigwebsite.com
    end
```

Verification

To verify that the addresses were added correctly:

1. Go to **Firewall Objects > Address > Addresses**. Check that the addresses have been added to the address list and that they are correct.
2. Enter the following CLI command:

```
config firewall address
    edit <the name of the address that you wish to verify>
        Show full-configuration
```

Changing the TTL of a FQDN address

To make sure that the FQDN resolves to the most recent active server you have been asked to make sure that the FortiGate has not cached the address for any longer than 10 minutes.

There is no field for the cached time-to-live in the web-based manager. It is only configurable in the CLI. Enter the following commands:

```
config firewall address
  edit BigWebsite.com
    set cache-ttl 600
  end
```

Geography based addresses

Geography addresses are those determined by country of origin.

This type of address is only available in the IPv4 address category.

Creating a geography address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**.
3. In the **Category** field, chose **Address**. (This is for IPv4 addresses.)
4. Input a **Name** for the address object.
5. In the **Type** field, select **Geography** from the drop down menu.
6. In the **Country** field, select a single country from the drop down menu.
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example: Geography-based address

Configuring the address in the GUI

Your company is US based and has information on its web site that may be considered information that is not allowed to be sent to embargoed countries. In an effort to help reduce the possibility of sensitive information going to those countries you have been asked to set up addresses for those countries so that they can be blocked in the firewall policies.

- One of the countries you have been asked to block is Cuba
 - You have been asked to comment the addresses so that other administrators will know why they have been created
1. Go to **Policy & Objects > Objects > Addresses** and select **Create New > Address**.
 2. Fill out the fields with the following information

Category	Address
Name	Cuba
Type	Geography
Country	Cuba
Interface	any
Visibility	<enable>
Comments	Embargoed

3. Select **OK**.

Configuring the address in the CLI

Enter the following CLI commands:

```
config firewall address
  edit Cuba
    set type geography
    set country CN
    set interface wan1
  end
```

Overrides

It is possible to assign a specific ip address range to a customized country ID. Generally, geographic addressing is done at the VDOM level; it could be considered global if you are using the root VDOM, but the geoip-override setting is a global setting.

```
config system geoip-override
  edit "test"
    set country-id "A0"
    config ip-range
      edit 1
        set start-ip 7.7.7.7
        set end-ip 7.7.7.8
        next
      edit 2
        set start-ip 7.7.10.1
        set end-ip 7.7.10.255
    end
```



- While the setting exists in the configuration file, the system assigns the country-id option automatically.
- While you can use "edit 1" and "edit 2", it is simpler to use "edit 0" and let the system automatically assign an ID number.

After creating a customized Country by using geoip-override command, the New country name has been added automatically to the country list and will be available on the Firewall Address Country field.

Diagnose commands

There are a few diagnose commands used with geographic addresses. The basic syntax is:

```
diagnose firewall ipgeo [country-list | ip-list | ip2country | override | copyright-notice]
```

Diagnose command	Description
country-list	Listing of all the countries.
ip-list	List of the IP addresses associated with the country
ip2country	Used to determine which country a specific IP address is assigned to.
override	Listing of user defined geography data - items configured by using "config system geoip-override" command.
copyright-notice	Shows the copyright notice.



Click on the diagnose command in the table to connect to the Fortinet Diagnose Wiki page that deals with the command option, to get more information.

IP range addresses

Where the subnet address is good a representing a standardized group of addresses that are subnets the IP Range type of address can describe a group of addresses while being specific and granular. It does this by specifying a continuous set of IP addresses between one specific IP address and another. While it is most common that this range is with a subnet it is not a requirement. For instance, 192.168.1.0/24 and 192.168.2.0/24 would be 2 separate subnets but if you wanted to describe the top half of one and the bottom half of the other you could describe the range of 192.168.1.128-192.168.2.127. It's also a lot easier than trying to calculate the correct subnet mask.

The format would be:

x.x.x.x-x.x.x, such as 192.168.110.100-192.168.110.120

There is a notation that is commonly used and accepted by some devices that follows the format:

x.x.x.[x-x], such as 192.168.110.[100-120]

This format is not recognized in FortiOS 5.2 as a valid IP Range.

Creating a IP range address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**
3. In the **Category** field, chose **Address(IPv4 addresses)** or **IPv6 Address**.
4. Input a **Name** for the address object.
5. In the **Type** field, select **IP Range** from the drop down menu.
6. In the **Subnet / IP Range** field, enter the range of addresses in the following format: x.x.x.x-x.x.x.x (no spaces)
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu. (This setting is not available for IPv6 addresses)
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example

Example of a IP Range address for a group of computers set aside for guests on the company network.

Field	Value
Category	Address or IPv6 Address
Name	Guest_users
Type	IP Range
Subnet / IP Range	192.168.100.200-192.168.100.240
Interface	Port1
Show in Address List	[on]
Comments	Computers on the 1st floor used by guests for Internet access.



IP Range addresses can be configured for both IPv4 and IPv6 addresses. The only differences in creating an IPv6 IP Range address is that you would choose IPv6 Address for the Category and the syntax of the address in the Subnet/IP Range field would be in the format of 2001:0db8:0000:0002:0:0:20-2001:0db8:0000:0004:0:0:20

IP / netmask addresses

The subnet type of address is expressed using a host address and a subnet mask. From a strictly mathematical stand point this is the most flexible of the types because the address can refer to as little one individual address or as many as all of the available addresses.

It is usually used when referring to your own internal addresses because you know what they are and they are usually administered in groups that are nicely differentiated along the lines of the old A, B, and C classes of IPv4 addresses. They are also addresses that are not likely to change with the changing of Internet Service Providers (ISP).

When representing hosts by an IP address with a netmask, the IP address can represent one or more hosts. For example, a firewall address can be:

- A single host such as a single computer with the address 192.45.46.45
- A range of hosts such as all of the hosts on the subnet 192.45.46.1 to 192.45.46.255
- All hosts, represented by 0.0.0.0 which matches any IP address

The netmask corresponds to the subnet class of the address being added, and can be represented in either dotted decimal or CIDR format. The FortiGate unit automatically converts CIDR formatted netmasks to dotted decimal format. Example formats:

- Netmask for a class A subnet of 16,777,214 usable addresses: 255.0.0.0, or /8
- Netmask for a class B subnet of 65,534 usable addresses: 255.255.0.0, or /16
- Netmask for a class C subnet of 254 usable addresses: 255.255.255.0, or /24
- Netmask for subnetted class C of 126 usable addresses: 255.255.255.128, or /25
- Netmask for subnetted class C of 62 usable addresses: 255.255.255.128, or /26
- Netmask for subnetted class C of 30 usable addresses: 255.255.255.128, or /27
- Netmask for subnetted class C of 14 usable addresses: 255.255.255.128, or /28
- Netmask for subnetted class C of 6 usable addresses: 255.255.255.128, or /29
- Netmask for subnetted class C of 2 usable addresses: 255.255.255.128, or /30
- Netmask for a single computer: 255.255.255.255, or /32
- Netmask used with 0.0.0.0 to include all IP addresses: 0.0.0.0, or /0

So for a single host or subnet the valid format of IP address and netmask could be either:

x.x.x.x/x.x.x.x, such as 192.168.1.0/255.255.255.0

or

x.x.x.x/x, such as 192.168.1.0/24

Static route configuration

A setting that is found in the IP/Netmask address type that is not found in the other address types is the enabling or disabling of **Static Route Configuration**. Enabling this feature includes the address in the listing of named addresses when setting up a static route.

To use in the GUI

1. Enable the **Static Route Configuration** in the address.
2. Go to **Network > Static Routes** and create a new route.
3. For a **Destination** type, choose **Named Address**.
4. Using the drop down menu, enter the name of the address object in the field just underneath the **Destination** type options.
5. Fill out the other information relevant to the route
6. Select the **OK** button

To enable in the CLI:

```
config firewall address
  edit <address_name>
    set allow-routing enable
  end
```

Creating a subnet address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**.
3. In the **Category** field, chose **Address**. (This is for IPv4 addresses.)
4. Input a **Name** for the address object.
5. In the **Type** field, select **IP/Netmask** from the drop down menu.
6. In the **Subnet/IP Range** field, enter the address and subnet mask according to the format x.x.x.x/x.x.x.x or the short hand format of x.x.x.x/x
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
9. Select the desired on/off toggle setting for **Static Route Configuration**.
10. Input any additional information in the **Comments** field.
11. Press **OK**.

Example

Example of a Subnet address for a database server on the DMZ:

Field	Value
Category	Address
Name	DB_server_1
Type	IP/Netmask
Subnet/IP Range	United States
Interface	any
Show in Address List	[on]
Static Route Configuration	[off]
Comments	

Wildcard addressing

Wildcard addresses are addresses that identify ranges of IP addresses, reducing the amount of firewall addresses and security policies required to match some of the traffic on your network. Wildcard addresses are an advanced

feature, usually required only for complex networks with complex firewall filtering requirements. By using these wildcard addresses in the firewall configuration, administrators can eliminate creating multiple, separate IP based address objects and then grouping them to then apply to multiple security policies.

A wildcard address consists of an IP address and a wildcard netmask, for example, 192.168.0.56 255.255.0.255. In this example, the IP address is 192.168.0.56 and the wildcard netmask is 255.255.0.255. The IP address defines the networks to match and the wildcard netmask defines the specific addresses to match on these networks.

In a wildcard netmask, zero means ignore the value of the octet in the IP address, which means the wildcard firewall address matches any number in this address octet. This also means that the number included in this octet of IP address is ignored and can be any number. Usually, if the octet in the wildcard netmask is zero, the corresponding octet in the IP address is also zero.

In a wildcard netmask, a number means match addresses according to how the numbers translate into binary addresses. For example, the wildcard netmask is 255; the wildcard address will only match addresses with the value for this octet that is in the IP address part of the wildcard address. So, if the first octet of the IP address is 192 and the first octet of the wildcard netmask is 255, the wildcard address will only match addresses with 192 in the first octet.

In the above example, the wildcard address 192.168.0.56 255.255.0.255 would match the following IP addresses:

```
192.168.0.56
192.168.1.56
192.168.2.56
...
192.168.255.56
```

The wildcard addresses 192.168.0.56 255.255.0.255 and 192.168.1.56 255.255.0.255 define the same thing since the 0 in the wildcard mask means to match any address in the third octet.

If we use the wildcard address 172.0.20.10 255.0.255.255, it would match the following IP addresses:

```
172.1.20.10
172.2.20.10
172.3.20.10
...
172.255.20.10
```

If you do not want to use all of the values in the octet, you can select a smaller grouping by using a number other than 255 in the subnet mask. There are some limitations though that users familiar with subnetting an IP range will recognize.

- The range should be a value that is a power of 2, such as 2, 4, 8, 16, etc
- The starting number should be the first number in a number grouping of the same power of 2 divided by the possible range. For instance, if the range is 32 addresses, divide 256 by 32. Start at 0. Therefore the starting IP address can be 0, 33, 65, 97, 129, 161, 193, and 225. If you don't want to do the math yourself there are a number of online subnet calculators that you can use.

You can perform a binary conversion to calculate the addresses that would be matched by a given value. For example, you can create the IP address and wildcard netmask to match the following network addresses:

```
192.168.32.0/24
192.168.33.0/24
192.168.34.0/24
192.168.35.0/24
192.168.36.0/24
192.168.37.0/24
192.168.38.0/24
192.168.39.0/24
```

In this example the range for the numbers in the third octet is 8.

The table shows how to write the third octet for these networks according to the octet bit position and address value for each bit.

Decimal	128	64	32	16	8	4	2	1
32	0	0	1	0	0	0	0	0
33	0	0	1	0	0	0	0	1
34	0	0	1	0	0	0	1	0
35	0	0	1	0	0	0	1	1
36	0	0	1	0	0	1	0	0
37	0	0	1	0	0	1	0	1
38	0	0	1	0	0	1	1	0
39	0	0	1	0	0	1	1	1
	Match	Match	Match	Match	Match	Differ	Differ	Differ

Use some basic math:

$128 + 64 + 32 + 16 + 8 = 248$

248 is the value in the subnet mask for the third octet.

The networks can be summarized into one network (192.168.32.0/21 or 192.168.32.0 255.255.248.0). All eight possible combinations of the three low-order bits are relevant for the network ranges. The wildcard address that would match all of these subnet addresses can be written as 192.168.32.0 255.255.248.0.

Wildcard addresses are similar to routing access list wildcard masks. You add routing access lists containing wildcard masks using the config router access list command. However, router access list wildcard masks use the inverse of the masking system used for firewall wildcard addresses. For the router access list wildcard masks, zero (0) means match all IP addresses and one (1) means ignore all IP addresses. So to match IP addresses

192.168.0.56, 192.268.1.56, 192.168.2.56,... 192.168.255.56 you would use the following router access IP address prefix and wildcard mask: 192.168.0.56 0.0.255.0.

The following is an example of how to configure a wildcard firewall address.

```
config firewall address
    edit example_wildcard_address
        set type wildcard
        set wildcard 192.168.0.56 255.255.0.255
    end
```



Wildcard firewall addresses are initially configured in the CLI. You cannot choose wildcard in the GUI when creating the address, but after the address is created in the CLI, it will show up in the GUI. The **Type** field shows a grayed out value of **Wildcard** and the settings, other than the type, can be edited.

Wildcard FQDN

There are a number of companies that use secondary and even tertiary domain names or FQDNs for their websites. Wildcard FQDN addresses are to ease the administrative overhead in cases where this occurs. Sometimes it's as simple as sites that still use www. as a prefix for their domain name. If you don't know whether or not the www is being used it's simpler to use a wildcard and include all of the possibilities whether it be example.com, www.example.com or even ftp.example.com.

The following wildcard character instances are supported in wildcard FQDN addresses:

- "?" character
- "*" character in the middle of a phrase
- The "?*" combination



Wildcard FQDN addresses do not resolve to a specific set of IP addresses in the same way that a normal FQDN address does. They are intended for use in SSL exemptions and should not be used as source or destination addresses in policies.

Creating a Fully Qualified Domain Name address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**
3. In the **Category** field, chose **Address**. (This is for IPv4 addresses.)
4. Input a **Name** for the address object.
5. In the **Type** field, select **Wildcard FQDN** from the drop down menu.
6. Input the domain name in the **Wildcard FQDN** field.
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example

Example of a FQDN address for a remote FTP server used by Accounting team:

Field	Value
Category	Address
Name	Example.com_servers
Type	Wildcard FQDN
Wildcard FQDN	*.example.com
Interface	any
Show in Address List	[on]
Comments	Secondary and tertiary domain names for example.com

Wildcard FQDNs for SSL deep inspection exemptions

As part of an improvement to SSL deep inspection, wild card FQDN addresses are stored in two tables, one relates to firewall address, historic location for the information, and the second location relates to firewall wildcard-fqdn custom. The wildcard FQDN in firewall address is used by proxy-policy. The wildcard FQDN in firewall wildcard-fqdn custom is used by ssl-exempt in ssl-ssh-profile.



During an upgrade from v5 to v6, all wildcard FQDN in firewall address in the v5 configuration will be moved to firewall wildcard-fqdn custom. If the wildcard FQDN is used in a policy in v5, the upgrade process will leave a copy of the wildcard FQDN in firewall address in addition to the one in firewall wildcard-fqdn custom.

Syntax of the firewall wildcard-fqdn custom object:

```
config firewall wildcard-fqdn custom
  edit <string_value>
    set uuid <string_value>
    set wildcard-fqdn <string_value>
    set color <integer 0-32>
    set comment <string_value>
    set visibility {enable|disable}
    next
  end
```

Syntax of the firewall wildcard-fqdn group object:

```
config firewall wildcard-fqdn group
  edit "test-group"
    set uuid <string_value>
    set member <string_value> [<string_value>]
    set color 0
    set comment ''
    set visibility enable
    next
  end
```



In the CLI, separate group members with a space.

IPv6 addresses

When creating an IPv6 address there are a number of different types of addresses that can be specified. These include:

- Subnet
- IP Range - the details of this type of address are the same as the IPv4 version of this type
- IPv6 FQDN firewall addresses - similar to the IPv4 version.

The IPv6 addresses don't yet have the versatility of the IPv4 address in that they don't have things like geography based addresses, but as IPv6 becomes more mainstream this should change.

Subnet addresses

The Subnet Address type is one that is only used in reference to IPv6 addresses. It represents an IPv6 address subnet. This means that the address will likely be a series of hexadecimal characters followed by a double colon, followed by a "/", and then a number less than 128 to indicate the size of the subnet. An example would be:

fd5e:3c59:35ce:f67e::/64

- The hexadecimal characters represent the IPv6 subnet address.
- The "::" indicates 0's from that point to the left. In an actual address for a computer, the hexadecimal characters that would take the place of these zeros would represent the device address on the subnet.
- /xx, in this case /64 represents the number of bits in the subnet. This will make a range that can potentially include 18,446,744,073,709,551,616 addresses. For those wanting to use English rather than math, that is 18 Quintillion.

Creating a subnet address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**
3. In the **Category** field, chose **IPv6 Address**.
4. Input a **Name** for the address object.
5. In the **Type** field, select **Subnet** from the drop down menu.
6. In the **Subnet / IP Range** field, enter the range of addresses in IPv6 format (no spaces)
7. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
8. Input any additional information in the **Comments** field.
9. Press **OK**.

Example

Example of a IP Range address for a group of computers set aside for guests on the company network.

Field	Value
Category	IPv6 Address
Name	IPv6_Guest_user_range
Type	Subnet
Subnet / IP Range	fd5e:3c59:35ce:f67e::/64
Show in Address List	[on]
Comments	

IPv6 FQDN firewall addresses

FQDN firewall addresses can be configured for IPv6.

Syntax in CLI

```
config firewall address6
  edit <address_name>
    set type fqdn
    set fqdn <domain_name>
    set cache-ttl <integer value from 0 to 86400>
  end
```

Firewall IPv6 address templates

You can use the IPv6 address templates to create new IPv6 addresses that share a prefix. Using templates for addresses reduces the chance of configuring an incorrect address due to a typographical error.

- A standard IPv6 address can be divided into three parts:
[IPv6 network prefix] + [subnet segments] + [host address]
- The subnet segments can be split into multiple 4-bit blocks called nibbles
- Each subnet segments represent different geographical or organizational parts of the network. They are represented by 1 or more nibbles.

Example of a prefix:

2001:db8:1234:0000::/64

Section	Description
The yellow highlighted characters	Prefix (48 bits)
The green highlighted characters (zeros)	Place holder for the subnet segments (16 bits)
The red highlighted characters	Subnet mask

The 16 bits that make up the subnet segments can be more granular.

Example: 0011 1111 0000 1101

Segment	Binary	Hexadecimal
Site	0011	0x3
Subsite	1111	0xf
Subnet	0000 1101	0x0d

The resulting network portion of the address is:

2001:db8:1234:3f0d::/64

By changing the mask, the subnet segment could be increased.

2001:db8:1234:0000:0000::/48

2001:db8:1234:0000:0000:0000::/32

This makes more options available for the configuration of the subnet segments. Below is an example of a very basic template:

Edit IPv6 Address Template

Name	test																												
IPv6 Address Prefix	2001:db8::/32																												
Subnet Segments <small>i</small>																													
<input type="button" value="+ Create New"/> <input type="button" value="Edit Segment"/> <input type="button" value="Edit Values for Segment"/> <input type="button" value="Delete"/> <table border="1"> <thead> <tr> <th>Segment Name</th> <th>Bits</th> <th>Exclusive</th> <th>Defined Values</th> </tr> </thead> <tbody> <tr> <td>country</td> <td>4</td> <td>Disable</td> <td></td> </tr> <tr> <td>state</td> <td>4</td> <td>Disable</td> <td></td> </tr> <tr> <td>city</td> <td>4</td> <td>Disable</td> <td></td> </tr> <tr> <td>site</td> <td>4</td> <td>Disable</td> <td></td> </tr> <tr> <td>lan</td> <td>4</td> <td>Disable</td> <td></td> </tr> <tr> <td>vlan</td> <td>4</td> <td>Disable</td> <td></td> </tr> </tbody> </table>		Segment Name	Bits	Exclusive	Defined Values	country	4	Disable		state	4	Disable		city	4	Disable		site	4	Disable		lan	4	Disable		vlan	4	Disable	
Segment Name	Bits	Exclusive	Defined Values																										
country	4	Disable																											
state	4	Disable																											
city	4	Disable																											
site	4	Disable																											
lan	4	Disable																											
vlan	4	Disable																											
<input type="button" value="OK"/> <input type="button" value="Cancel"/>																													

Using that template, you can see how the GUI could be used to quickly create address objects.

New Address

Category	Address	IPv6 Address	Multicast Address	Proxy Address
Name	example			
Color	<input type="button" value="Change"/>			
Type	IPv6 Template			
IPv6 Address Template	<input type="button" value="test"/>			
Subnet Prefix	2001:db8::/32			
Segment Name: country (4 bits)	Any	Specify	0x5	<input type="button" value=""/>
Segment Name: state (4 bits)	Any	Specify	0x9	<input type="button" value=""/>
Segment Name: city (4 bits)	Any	Specify	0xd	<input type="button" value=""/>
Segment Name: site (4 bits)	Any	Specify	0xf	<input type="button" value=""/>
Segment Name: lan (4 bits)	Any	Specify	0x8	<input type="button" value=""/>
Segment Name: vlan (4 bits)	Any	Specify	0x2	<input type="button" value=""/>
Host Type	Any	Specify		
Host	::12e7			
Show in Address List	<input checked="" type="checkbox"/>			
Comments	<input type="text" value=""/> 0/255			
Tags	<input type="button" value="Add Tag Category"/>			
<input type="button" value="OK"/> <input type="button" value="Cancel"/>				



You can use the template to enter the subnet prefix alone. You don't have to use the segment portion of the template.

Multicast addresses

Multicast addressing defines a specific range of address values set aside for them. Therefore all IPv4 multicast addresses should be between 224.0.0.0 and 239.255.255.255.

More information on the concepts behind Multicast addressing can be found in the Multicast Forwarding section.

Multicast IP range

This type of address will allow multicast broadcasts to a specified range of addresses.

Creating a multicast IP range address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**.
 - If you use the down arrow next to **Create New**, select **Address**.
3. Choose the **Category, Multicast Address**
4. Input a **Name** for the address object.
5. Select the **Type, Multicast IP Range** from the drop-down menu.
6. Enter the value for the **Multicast IP Range**

7. Select the **Interface** from the drop-down menu.
8. Enable the **Show in Address List** function
9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example: Multicast IP range address

The company has a large high tech campus that has monitors in many of its meeting rooms. It is common practice for company wide notifications of importance to be done in a streaming video format with the CEO of the company addressing everyone at once.

The video is High Definition quality so takes up a lot of bandwidth. To minimize the impact on the network the network administrators have set things up to allow the use of multicasting to the monitors for these notifications. Now it has to be set up on the FortiGate firewall to allow the traffic.

- The range being used for the multicast is 239.5.5.10 to 239.5.5.200
- The interface on this FortiGate firewall will be on port 9

1. Go to **Policy & Objects> Objects > Addresses** and select **Create New > Address**.
2. Fill out the fields with the following information

Category	Multicast Address
Name	Meeting_Room_Displays
Type	Multicast IP Range
Multicast IP Range	239.5.5.10-239.5.5.200
Interface	port9
Show in Address List	<enable>
Comments	<Input into this field is optional>

3. Select **OK**.
4. Enter the following CLI command:

```
config firewall multicast-address
    edit "meeting_room_display"
        set type multicastrange
        set associated-interface "port9"
        set start-ip 239.5.5.10
        set end-ip 239.5.5.200
        set visibility enable
    next
end
```

To verify that the address range was added correctly:

1. Go to **Policy & Objects> Objects > Addresses**. Check that the addresses have been added to the address list and that they are correct.
2. Enter the following CLI command:

```
config firewall multicast-address
```

```
edit <the name of the address that you wish to verify>
Show full-configuration
```

Broadcast subnet

This type of address will allow multicast broadcast to every node on a subnet.

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**.
3. In the **Category** field, chose **Multicast Address**.
4. Input a **Name** for the address object.
5. In the **Type** field, select **Broadcast Subnet** from the drop down menu.
6. In the **Broadcast Subnet** field enter the address and subnet mask according to the format x.x.x.x/x.x.x.x or the short hand format of x.x.x.x/x.(Remember, it needs to be within the appropriate IP range 224.0.0.0 to 239.255.255.255)
7. In the **Interface** field, leave as the default **any** or select a specific interface from the drop down menu.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
9. Input any additional information in the **Comments** field.
10. Press **OK**.

Example

Field	Value
Category	Broadcast Subnet
Name	Corpnet-B
Type	Broadcast Subnet
Broadcast Subnet	224.5.5.0/24
Interface	any
Show in Address List	[on]
Comments	Corporate Network devices - Broadcast Group B

Multicast IP addresses

Multicast uses the Class D address space. The 224.0.0.0 to 239.255.255.255 IP address range is reserved for multicast groups. The multicast address range applies to multicast groups, not to the originators of multicast packets. The following table lists the reserved multicast address ranges and describes what they are reserved for:

Reserved Multicast address ranges

Reserved Address Range	Use	Notes
224.0.0.0 to 224.0.0.255	Used for network protocols on local networks. For more information, see RFC 1700.	In this range, packets are not forwarded by the router but remain on the local network. They have a Time to Live (TTL) of 1. These addresses are used for communicating routing information.
224.0.1.0 to 238.255.255.255	Global addresses used for multicasting data between organizations and across the Internet. For more information, see RFC 1700.	Some of these addresses are reserved, for example, 224.0.1.1 is used for Network Time Protocol (NTP).
239.0.0.0 to 239.255.255.255	Limited scope addresses used for local groups and organizations. For more information, see RFC 2365.	Routers are configured with filters to prevent multicasts to these addresses from leaving the local system.

Creating multicast security policies requires multicast firewall addresses. You can add multicast firewall addresses by going to **Firewall Objects > Address > Addresses** and selecting **Create New > Multicast Address**. The factory default configuration includes multicast addresses for Bonjour (224.0.0.251-224.0.0.251), EIGRP (224.0.0.10-224.0.0.100), OSPF (224.0.0.5-224.0.0.60), all_hosts (224.0.0.1-224.0.0.1), and all_routers (224.0.0.2-224.0.0.2).

Proxy addresses

This category of address is different from the other addresses in that it is not designed to be used in the normal firewall policy configuration. It is intended to be used only with explicit web proxies.

In some respects they can be like a FQDN addresses in that they refer to an alpha-numeric string that is assigned to an IP address, but then goes an additional level of granularity by using additional information and criteria to further specify locations or types of traffic within the website itself. In depth information on Explicit Proxy Addressing can be found in [WAN Optimization](#), but it is worth laying out the steps of how to create an address object for this category.

Creating an proxy address

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**. A drop down menu is displayed. Select **Address**.
3. In the **Category** field, chose **Proxy Address**.
4. Input a **Name** for the address object.
5. For the **Type** field, select one of the options from the drop down menu.

Within the Explicit Proxy Address category there are 8 types of addresses. Each of these types will have associated field(s) that also need to have values entered to make the object specific to it's address.

Type = URL Pattern

- In the **Host** field, choose from drop down menu
- In the **URL Path Regex** field, enter the appropriate string

Host Regex Match

- In the **Host Regex Pattern** field, enter the appropriate string

URL Category

- In the **Host** field, choose from drop down menu
- In the **URL Category** field, choose from drop down menu

HTTP Method

- In the **Host** field, choose from drop down menu
- In the **Request Method** field, choose from drop down menu

The options are:

- CONNECT
- DELETE
- GET
- HEAD
- OPTIONS
- POST
- PUT
- TRACE

User Agent

- In the **Host** field, choose from drop down menu
- In the **User Agent** field, choose from drop down menu

The options are:

- Apple Safari
- Google Chrome
- Microsoft Internet Explorer or Spartan
- Mozilla Firefox
- Other browsers

HTTP Header

- In the **Host** field, choose from drop down menu
- In the **Header Name** field, enter the appropriate string value
- In the **Header Regex** field, enter the appropriate string value

Advanced (Source)

- In the **Host** field, choose from drop down menu
- In the **Request Method** field, choose from drop down menu (see **HTTP Method** type for option list)
- In the **User Agent** field, choose from drop down menu (see **User Agent** type for option list)
- In the **Header Group** table, create, edit or delete **Header Name** strings and associated **Header Regex** strings

Advance (Destination)

- In the **Host** field, choose from drop down menu
 - In the **Host Regex Pattern** field, enter the appropriate string
 - In the **URL Category** field, choose from drop down menu
6. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled the address will appear in drop down menus where it is an option.
7. Input any additional information in the **Comments** field.
8. Press **OK**.

Proxy address groups

To create a Proxy address group:

1. Go to **Policy & Objects > Addresses**.
2. Click on **+ Create New** to get the drop down menu. Select **Address Group**.
3. In the **Category** field, choose **Proxy Group**.
4. Fill in a descriptive name in the **Group Name** field.
5. If you wish, use the **Change** link to change the **Color** of icons in the GUI. There are 32 color options.
6. In the **Type** field, select whether the group will be a **Source Group** (composed of source addresses) or a **Destination Group** (composed of destination addresses).
7. Select anywhere in the **Members** field to bring forth the pane of potential members for selection to the group.
8. Select the desired on/off toggle setting for **Show in Address List**. If the setting is enabled, the address will appear in drop down menus where it is an option.
9. Input any additional information in the **Comments** field.
10. Click on **OK**.

New Address Group

Category	<input type="radio"/> IPv4 Group <input type="radio"/> IPv6 Group <input checked="" type="radio"/> Proxy Group
Group Name	<input type="text"/>
Color	[Change]
Type	<input checked="" type="radio"/> Source Group <input type="radio"/> Destination Group
Members	<input type="text"/> +
Show in Address List	<input checked="" type="checkbox"/>
Comments	<input type="text" value="Write a comment..."/> 0/255

OK **Cancel**

Internet services

In FortiOS 5.4, support was added for Internet Service objects which could be used with **FortiView**, **Logging**, **Routing** and **WAN Load Balancing**. Now they can be added to firewall policies as well.



There is an either or relationship between Internet Service objects and destination address and service combinations in firewall policies. This means that a destination address and service can be specified in the policy OR an Internet service, not both.

CLI

The related CLI options/syntax are:

```
config firewall policy
  edit 1
    set internet-service 1 5 10
    set internet-service-custom test
    set internet-service-negate [enable|disable]
  end
```

GUI

In the policy listing page you will notice that an Internet Service object is used, it will be found in both the **Destination** and **Service** column.

In the policy editing page the **Destination Address**, now **Destination** field now has two types, **Address** and **Internet Service**.

New Policy

Name: Citrix access

Incoming Interface: port1

Outgoing Interface: port2

Source: all

Destination:

- Citrix-DNS
- Citrix-FTP(S)
- Citrix-IMAP(S)
- Citrix-NetBIOS.Name.Service
- Citrix-NetBIOS.Session.Service
- Citrix-SMTP(S)
- Citrix-SSH
- Citrix-Web
- CNN-FTP(S)

Select Entries

Address: Internet Service

Search:

- Citrix-FTP(S)
- Citrix-IMAP(S)
- Citrix-NetBIOS.Name.Service
- Citrix-NetBIOS.Session.Service
- Citrix-SMTP(S)
- Citrix-SSH
- Citrix-Web
- CNN-FTP(S)
- CNN-SMTP(S)
- Dropbox-DNS
- Dropbox-NetBIOS.Name.Service

Address groups

Address groups are designed for ease of use in the administration of the device. If you have a number of addresses or address ranges that will commonly be treated the same or require the same security policies, you can put them into address groups, rather than entering multiple individual addresses in each policy refers to them.

The use of groups is not required. If you have a number of different addresses you could add them individually to a policy and the FortiGate firewall will process them just as quickly and efficiently as if they were in a group, but the chances are that if you have used a group once you could need to use it again and depending on the number of addresses involved entering them individually for each policy can become tedious and the likelihood of an address being missed becomes greater. If you have a number of policies using that combination of addresses it is much easier to add or subtract addresses from the group than to try and remember all of the firewall policies that combination of addresses was used in. With the group, you only have to make the one edit and it is used by any firewall policy using that address group.

Because security policies require addresses with homogenous network interfaces, address groups should contain only addresses bound to the same network interface, or to Any.

For example, if address 1.1.1.1 is associated with port1, and address 2.2.2.2 is associated with port2, they cannot be in the same group. However, if 1.1.1.1 and 2.2.2.2 are configured with an interface of Any, they can be grouped, even if the addresses involve different networks.

There are 3 Categories of Address groups to choose from:

- IPv4 Group
- IPv6 Group
- Proxy Group

You cannot mix different categories of addresses within a group, so whether or not it makes sense from an administrative purpose to group certain addresses together, if some are IPv4 and some are IPv6, it cannot be done.

Creating an address group

1. Go to **Policy & Objects > Addresses**.
2. Select the down arrow next to **Create New**, select **Address Group**.
3. Choose the **Category**, that is applicable to the proposed selection of addresses.
4. Input a **Group Name** for the address object.

Depending on which **Category** has been chosen the configurations will differ slightly

IPv4 group

1. Select the "+" in the **Members** field. You can select members of the group from the window that slides out from the left of the screen. It is possible to select more than 1 entry. Select the "X" icon in the field to remove an entry.
2. Select the desired on/off toggle setting for **Show in Address List**.
3. Select the desired on/off toggle setting for **Static Route Configuration**.

IPv6 group

1. Select the "+" in the **Members** field. You can select members of the group from the window that slides out from the left of the screen. It is possible to select more than 1 entry. Select the "X" icon in the field to remove an entry.
2. Select the desired on/off toggle setting for **Show in Address List**.

Proxy group

1. Select which Type, either **Source Group** or **Destination Group**.
2. Select the "+" in the **Members** field. You can select members of the group from the window that slides out from the left of the screen. It is possible to select more than 1 entry. Select the "X" icon in the field to remove an entry.
3. Select the desired on/off toggle setting for **Show in Address List**.

Irrespective of the Category the groups all have the same final configuration options:

1. Input any additional information in the **Comments** field.
2. Press **OK**.

UUID support

Syntax:

```
config firewall {address|address6|addgrp|addgrp6}
    edit 1
        set uuid <example uuid: 8289ef80-f879-51e2-20dd-fa62c5c51f44>
        next
    end
```

Virtual IPs

The mapping of a specific IP address to another specific IP address is usually referred to as Destination NAT. When the Central NAT Table is not being used, FortiOS calls this a Virtual IP Address, sometimes referred to as a VIP. FortiOS uses a DNAT or Virtual IP address to map an External IP address to an IP address. This address

does not have to be an individual host, it can also be an address range. This mapping can include all TCP/UDP ports or if Port Forwarding is enabled it will only refer to the specific ports configured. Because, the Central NAT table is disabled by default the term Virtual IP address or VIP will be used predominantly.

Virtual IP addresses are typically used to NAT external or Public IP addresses to internal or Private IP addresses. Using a Virtual IP address between 2 internal Interfaces made up of Private IP addresses is possible but there is rarely a reason to do so as the 2 networks can just use the IP addresses of the networks without the need for any address translation. Using a Virtual IP address for traffic going from the inside to the Internet is even less likely to be a requirement, but it is supported.

Something that needs to be considered when there are multiple Public IP addresses on the external interface(s) is that when a Virtual IP address is used without Port Forwarding enabled there is a reciprocal effect as far as traffic flow is concerned. Normally, on a firewall policy where NAT is enabled, for outgoing traffic the internal address is translated to the Public address that is assigned to the FortiGate, but if there is a Virtual IP address with no port forwarding enabled, then the Internal IP address in the Mapped field would be translated to the IP address configured as the External Address in the VIP settings.

Example

- The assigned External address (WAN1) of the FortiGate unit is 172.12.96.3 with a subnet mask of 255.255.255.128
- There is a Virtual IP address set up to map the external address 172.12.96.127 on WAN1 to the internal IP address of 192.168.1.127
- Port Forwarding is not enabled because you want all allowed traffic going to the external IP address to go to this server.

In this case any outbound traffic from 192.168.1.127 will go out on WAN1 with the IP address of 172.12.96.127 as the source IP address.

In terms of actually using the Virtual IP address, they would be using in the security policies in the same places that other addresses would be used, usually as a Destination Address.

UUID support for VIP

UUID is now supported in for virtual IPs and virtual IP groups. This includes virtual IPs for IPv4, IPv6, NAT46, and NAT64. To view the UUID for these objects in a FortiGate unit's logs, log-uuid must be set to extended mode, rather than policy-only (which only shows the policy UUID in a traffic log). UUID can only be configured through the CLI

Syntax

```
config sys global
    set log-uuid {disable | policy-only | extended}
end
```



There is another type of address that the term “virtual IP address” commonly refers to which is used in load balancing and other similar configurations. In those cases, a number of devices share a separately created virtual IP address that can be sent to multiple possible devices. In FortiOS these are referred to as Virtual Servers and are configured in the “Load Balance” section.



If Central-NAT is enabled in the CLI the GUI will be different.

Instead of **VIP Type**, the field label will be **DNAT & VIP Type**

Instead of **IPv4** the option will be **IPv4 DNAT**

There will also be the addition setting of **Source Interface Filter**.

Commands to set `central-nat`:

```
config system settings
  set central-nat [enable | disable]
end
```

Creating a virtual IP

1. Go to **Policy & Objects > Virtual IPs**.
2. Select **Create New**. A drop down menu is displayed. Select **Virtual IP**.
3. From the **VIP Type** options, choose an applicable type based on the IP addressing involved. Which is chosen will depend on which of the IP version networks is on the external interface of the FortiGate unit and which is on the internal interface.
The available options are:
 - **IPv4** - IPv4 on both sides of the FortiGate Unit.
 - **IPv6** - IPv6 on both sides of the FortiGate Unit.
 - **NAT46** - Going from an IPv4 Network to an IPv6 Network.
 - **NAT64** - Going from an IPv6 Network to an IPv4 Network.
4. In the **Name** field, input a unique identifier for the Virtual IP.
5. Input any additional information in the **Comments** field.
6. The **Color** of the icons that represent the object in the GUI can be changed by clicking on the **[Change]** link and choosing from the 32 colors.

Because the configuration differs slightly for each type the next steps will be under a separate heading based on the type of the VIP

Configuring a VIP for IPv4

In the **Network** section:

7. If an IPv4 type of Virtual IP, select the **Interface** setting.

Using the drop down menu for the Interface Field, choose the incoming interface for the traffic.

The IPv4 VIP Type is the only one that uses this field. This is a legacy function from previous versions so that they can be upgraded without complicated reconfiguration. The External IP address, which is a required field, tells the unit which interface to use so it is perfectly acceptable to choose **"any"** as the interface. In some configurations, if the Interface field is not set to **"any"** the Virtual IP object will not one of the displayed options when choosing a destination address.

8. Configure the **External IP Address/Range**.

There are two fields. If there is a single IP address, use that address in both fields. This will be the address on the outside of the network that is usually the public address of the server. The format of the address will depend on the **VIP Type** option that was selected.

9. Configure the **Mapped IP Address/Range**. This will be the address that the traffic is being directed to.

There are two fields. If there is a single IP address, use that address in both fields. The format of the address will depend on the **VIP Type** option that was selected.

In the **Optional Filters**

10. Disable/Enable the **Optional Filters**.

If only specific IP addresses and/or services are allowed to be the source for traffic using the VIP, enable the **Optional Filters**.

11. To specify an allowed address enter the value in the field labeled **Source Address**. The value can be formatted in three different ways.

- **Source IP** - Use the standard format for a single IP address

- **Range** - Enter the first and last members of the range

- **Subnet** - Enter the IP address of the broadcast address for the subnet.

To add additional addresses, click on the "+" below the last field with an address. To subtract an address, click on the "X" next to the field you wish to delete.

12. To specify an allowed Service, toggle the **Services** option to enabled. Set the **Services** parameter by selecting the field with the "+" in the field. This will slide a window out from the right. Single or multiple options can be selected by highlighting the services wanted, unless the **ALL** option is chosen, in which case it will be the only option. For more information on services, check the Firewall Objects section called [Services and TCP ports](#).

13. Disable/Enable **Port Forwarding**. If only the traffic for a specific port or port range is being forwarded, enable this setting.

14. Select the **Protocol** from

- **TCP**

- **UDP**

- **SCTP**

- **ICMP**

15. Configure the **External Service Port**. This is the port(s) on the external interface of the FortiGate (the destination port in the header of the packets). The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.

16. Configure the setting **Map to Port**. This will be the listening port on the device located on the internal side of the network. It does not have to be the same as the **External Service Port**. The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.

17. Press **OK**.

Example

This example is for a VIP that is being used to direct traffic from the external IP address to a web server on the internal network. The web server is for company use only. The company's public facing web server already used

port 80 and there is only one IP external IP address so the traffic for this server is being listened for on port 8080 of the external interface and being sent to port 80 on the internal host.

Field	Value
VIP Type	IPv4
Name	Internal_Webserver
Comments	Web server with Collaboration tools for Corporate employees
Interface	Any
External IP Address/Range	172.13.100.27 <this would normally be a public IP address>
Mapped IP Address/Range	192.168.34.150
Optional Filters	enabled
Source Address Filter	<list of IP addresses of remote users>
Services	enabled with HTTP in the list
Port Forwarding	enabled
Map to Port	80 - 80

Configuring a VIP for IPv6

In the **Network** section:

7. Configure the **External IP Address/Range**.

There are two fields. If there is a single IP address, use that address in both fields. This will be the address on the outside of the network that is usually the public address of the server. Enter the address in the standard IPv6 format.

8. Configure the **Mapped IP Address/Range**. This will be the address that the traffic is being directed to.

There are two fields. If there is a single IP address, use that address in both fields. Enter the address in the standard IPv6 format.

In the **Optional Filters**

9. Disable/Enable the **Optional Filters**.

If only specific IP addresses and/or services are allowed to be the source for traffic using the VIP, enable the **Optional Filters**.

10. To specify an allowed address enter the value in the field labeled **Source Address**. The value can be formatted in three different ways.

- **Source IP** - Use the standard format for a single IP address
 - **Range** - Enter the first and last members of the range
 - **Subnet** - Enter the IP address of the broadcast address for the subnet.
To add additional addresses, click on the "+" below the last field with an address. To subtract an address, click on the "X" next to the field you wish to delete.
12. Disable/Enable **Port Forwarding**. If only the traffic for a specific port or port range is being forwarded, enable this setting.
13. Select the **Protocol** from
- **TCP**
 - **UDP**
 - **SCTP**
14. Configure the **External Service Port**. This is the port(s) on the external interface of the FortiGate (the destination port in the header of the packets). The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
15. Configure the setting **Map to Port**. This will be the listening port on the device located on the internal side of the network. It does not have to be the same as the **External Service Port**. The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
16. Press **OK**.

Configuring a VIP for NAT46

In the **Network** section:

7. Configure the **External IP Address/Range**.

There are two fields. If there is a single IP address, use that address in both fields. This will be the address on the outside of the network that is usually the public address of the server. Enter the address in the standard IPv4 format.

8. Configure the **Mapped IP Address/Range**. This will be the address that the traffic is being directed to.

There are two fields. If there is a single IP address, use that address in both fields. Enter the address in the standard IPv6 format.

In the **Optional Filters**

9. Disable/Enable the **Optional Filters**.

If only specific IP addresses and/or services are allowed to be the source for traffic using the VIP, enable the **Optional Filters**.

10. To specify an allowed address enter the value in the field labeled **Source Address**. The value can be formatted in three different ways.
- **Source IP** - Use the standard format for a single IP address
 - **Range** - Enter the first and last members of the range
 - **Subnet** - Enter the IP address of the broadcast address for the subnet.

To add additional addresses, click on the "+" below the last field with an address. To subtract an address, click on the "X" next to the field you wish to delete.

12. Disable/Enable **Port Forwarding**. If only the traffic for a specific port or port range is being forwarded, enable this setting.
13. Select the **Protocol** from
 - **TCP**
 - **UDP**
14. Configure the **External Service Port**. This is the port(s) on the external interface of the FortiGate (the destination port in the header of the packets). The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
15. Configure the setting **Map to Port**. This will be the listening port on the device located on the internal side of the network. It does not have to be the same as the **External Service Port**. The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
16. Press **OK**.

In order for VIP46 to work correctly, NAT64 must be enabled. To enable NAT64, set the following configuration:



```
conf sys nat64
    set status enable
end
```

Configuring a VIP for NAT64

In the **Network** section:

7. Configure the **External IP Address/Range**.

There are two fields. If there is a single IP address, use that address in both fields. This will be the address on the outside of the network that is usually the public address of the server. Enter the address in the standard IPv6 format.

8. Configure the **Mapped IP Address/Range**. This will be the address that the traffic is being directed to.

There are two fields. If there is a single IP address, use that address in both fields. Enter the address in the standard IPv4 format.

In the **Optional Filters**

9. Disable/Enable the **Optional Filters**.

If only specific IP addresses and/or services are allowed to be the source for traffic using the VIP, enable the **Optional Filters**.

10. To specify an allowed address enter the value in the field labeled **Source Address**. The value can be formatted in three different ways.

- **Source IP** - Use the standard format for a single IP address
- **Range** - Enter the first and last members of the range
- **Subnet** - Enter the IP address of the broadcast address for the subnet.

To add additional addresses, click on the "+" below the last field with an address. To subtract an address, click on the "X" next to the field you wish to delete.

12. Disable/Enable **Port Forwarding**. If only the traffic for a specific port or port range is being forwarded, enable this setting.
13. Select the **Protocol** from
 - **TCP**
 - **UDP**
14. Configure the **External Service Port**. This is the port(s) on the external interface of the FortiGate (the destination port in the header of the packets). The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
15. Configure the setting **Map to Port**. This will be the listening port on the device located on the internal side of the network. It does not have to be the same as the **External Service Port**. The first field is for the first port in the range the second is for the last port in the range. As you enter a value in the first field, the second field will auto populate with the same number, working on the premise that a single port is common. Just edit the second field to extend the range.
16. Press **OK**.

FQDN in VIPs

Instead of mapping to an IP address a VIP can use a FQDN(Fully Qualified Domain Name). This has to be configured in the CLI and the FQDN must be an address object that is already configured in the address listing.

The syntax for using a FQDN is:

```
config firewall vip
  edit <VIP id>
    set type fqdn
    set mapped-addr <FQDN address object>
  end
```

Dynamic VIP according to DNS translation

When a dynamic virtual IP is used in a policy, the dynamic DNS translation table is installed along with the dynamic NAT translation table into the kernel. All matched DNS responses will be translated and recorded regardless if they hit the policy. When a client request hits the policy, dynamic NAT translation will occur if it matches a record, otherwise the traffic will be blocked.

Syntax

```
config firewall vip
  edit "1"
    set type dns-translation
    set extip 192.168.0.1-192.168.0.100
    set extintf "dmz"
    set dns-mapping-ttl 604800
    set mappedip "3.3.3.0/24" "4.0.0.0/24"
    end
  end
```

Virtual IP groups

Just like other address, Virtual IP addresses can be organized into groups for ease of administration. If you have multiple virtual IPs that are likely to be associated to common firewall policies rather than add them individually to each of the policies you can add the instead. That way, if the members of the group change then any changes made to the group will propagate to all of the policies using that group.

When using a Virtual IP address group the firewall policy will take into account all of the configured parameters of the Virtual IPs: IP addresses, Ports and port types.

Creating a virtual IP group

1. Go to **Policy & Objects > Virtual IPs**.
2. Select **Create New**. A drop down menu is displayed. Select **Virtual IP Group**.
3. Select the **Type** for VIP group you wish to create.

The options available are:

- **IPv4** - IPv4 on both sides of the FortiGate Unit.
- **IPv6** - IPv6 on both sides of the FortiGate Unit.
- **NAT46** - Going from an IPv4 Network to an IPv6 Network.
- **NAT64** - Going from an IPv6 Network to an IPv4 Network.

Which is chosen will depend on which of the IP version networks is on the external interface of the FortiGate unit and which is on the internal interface. The options will be:

4. Enter a unique identifier for the group in the **Name** field.
5. Enter any additional information in the **Comments** field.
6. If you wish, use the **Change** link to change the **Color** of icons in the GUI. There are 32 color options.
7. If the **Type** is **IPv4**, the **Interface** field will be available. Use the drop-down menu to select the interface if all of the VIPs are on the same interface. If any of the VIPs are on different interfaces or if any of them are associated with the "any" option, choose the any option for the group.
8. Select anywhere in the **Members** field to bring forth the pane of potential members for selection to the group.
9. Press **OK**.

Configuring IP pools

An IP pool is essentially one in which the IP address that is assigned to the sending computer is not known until the session is created, therefore at the very least it will have to be a pool of at least 2 potential addresses. A quick example would be an IP pool for users of a VPN. IP pools are based upon the version of IP determined by the interface that they are associated with so as expected there are two types of IP pools that can be configured:

- "Creating a IPv4 pool" on page 243
- "Creating a IPv6 pool" on page 247

Because of the differences in the configuration for the two types of pools, instructions for configuring them will be done separately.

Creating a IPv4 pool

1. Go to **Policy & Objects > IP Pools**.
2. Select **Create New**.
3. In the **IP Pool Type** field choose **IPv4 Pool**
4. Enter a name in the **Name** field for the new service
5. Include any description you would like in the **Comments** field
6. In the **Type** field choose between:
 - **Overload**
 - **One-to-One**
 - **Fixed Port Range**
 - **Port Block Allocation**

At this point the configurations can start to differ based on the type of type of pool.

For more information on the different types of IP pools, check [IP Pools](#) in the Concepts section.

Overload

7. For the **External IP Range** fields, enter the lowest and highest addresses in the range. If you only want a single address used, enter the same address in both fields.
8. Enable the **ARP Reply** field by making sure there is a check in the box
9. Select **OK**

Overload example for GUI

In this example, the Sales team needs to connect to an Application Service Provider that does the accounting for the company. As a security measure, the ASP only accepts traffic from a white list of IP addresses. There is 1 public IP address of the company on that list. The Sales team consists of 40 people, so they need to share. The external interface is wan1.

Field	Value
IP Pool Type	IPv4 Pool
Name	Sales_Team
Comments	For the Sales team to use to connect to the Accounting ASP
Type	Overload (This is the default)
External IP Range	10.23.56.20 - 10.23.56.20
ARP Reply	enabled

Overload example for CLI

```
config firewall ippool
  edit Sales_Team
    set comments "For the Sales team to use to connect to the Accounting ASP"
```

```

set type overload
set startip 10.23.56.20
set endip 10.23.56.20
set arp-reply enable
set arp-intf wan1
end

```

One-to-one

7. For the **External IP Range** fields, enter the lowest and highest addresses in the range. If you only want a single address used, enter the same address in both fields.
8. Enable the **ARP Reply** field by making sure there is a check in the box.
9. Select **OK**

One-to-one example for GUI

In this example, the external IP address of the mail server is part of a range assigned to the company but not the one that is assigned to the Internet facing interface. A VIP has been set up but in order to properly resolve Reverse DNS lookups the mail server always has to use a specific IP address. The external interface is wan1.

Field	Value
IP Pool Type	IPv4 Pool
Name	Mail-Server
Comments	So the correct IP address is resolved on Reverse DNS look ups of the mail server.
Type	One-to-one
External IP Range	10.23.56.21 - 10.23.56.21
ARP Reply	enabled

One-to-one example for CLI

```

config firewall ippool
  edit Mail-Server
    set comments "So the the correct IP address is resolved on reverse DNS look ups of
      the mail server."
    set type one-to-one
    set startip 10.23.56.21
    set endip 10.23.56.21
    set arp-reply enable
    set arp-intf wan1
  end

```

Fixed port range

7. For the **External IP Range** fields, enter the lowest and highest addresses in the range. If you only want a single address used, enter the same address in both fields.
8. For the **Internal IP Range** fields, enter the lowest and highest addresses in the range.

9. Enable the **ARP Reply** field by making sure there is a check in the box
10. Select **OK**

Fixed port range example for GUI

In this example, the company has a range of 10 IP address that they want to be used by employees on a specific subnet for NATing. The external interface is wan1.

Field	Value
IP Pool Type	IPv4 Pool
Name	IPPool-3
Comments	IP range to be used by outgoing traffic
Type	Fixed Port Range
External IP Range	10.23.56.22 - 10.23.56.31
Internal IP Range	192.168.23.1 - 192.168.23.254
ARP Reply	enabled

Fixed port range example for CLI

```
config firewall ippool
  edit IPPool-3
    set comments "So the the correct IP address is resolved on reverse DNS look ups of
      the mail server."
    set type fixed-port-range
    set startip 10.23.56.22
    set endip 10.23.56.31
    set source-startip 192.168.23.1
    set source-endip 192.168.23.254
    set arp-reply enable
    set arp-intf wan1
  end
```

Port block allocation

7. For the **External IP Range** fields, enter the lowest and highest addresses in the range. If you only want a single address used, enter the same address in both fields.
8. In the **Block Size** field, either type in the value or use the up or down arrows to set the value of the block size.
9. In the **Blocks Per User** field, either type in the value or use the up or down arrows to set the value for the number of blocks per user.
10. Enable the **ARP Reply** field by making sure there is a check in the box
11. Select **OK**

Port block allocation timeout

The port block allocation timeout value is configurable. The setting is found in the CLI.

The option `pba-timeout` has been added to the firewall ip pool configuration. The availability of this option is dependent on the type option being set to port-block-allocation. The timeout value is measured in seconds and is an integer between 3 and 300, with the default being 30.

Syntax:

```
config firewall ippool
    edit <name of PBA pool>
        set type port-block-allocation
        set pba-timeout <integer>
    end
```

Port block allocation example for GUI

In this example, an small ISP is setting up NATing for its clients, but to be fair it is putting some restrictions on the number of connections each client can have so that no one hogs all of the possible ports and addresses. The external interface is port12.

Field	Value
IP Pool Type	IPv4 Pool
Name	Client-IPPool
Comments	IP Pool for clients to access the Internet
Type	Port Block Allocation
External IP Range	10.23.75.5 - 10.23.75.200
Block Size	64
Blocks Per User	8
ARP Reply	enabled

Port block allocation example for CLI

```
config firewall ippool
    edit Client-IPPool
        set comments "IP Pool for clients to access the Internet"
        set type port-block-allocation
        set startip 10.23.75.5
        set endip 10.23.75.200
        set block-size 64
        set num-blocks-per-user 8
        set permit-any-host disable
        set arp-intf wan1
        set arp-reply enable
        arp-intf port12
    end
```

Creating a IPv6 pool

1. Go to **Policy & Objects > IP Pools**.
2. Select **Create New**.
3. In the IP Pool Type field choose **IPv6 Pool**
4. Enter a name in the **Name** field for the new service
5. Include any description you would like in the **Comments** field
6. For the **External IP Range** fields, enter the lowest and highest addresses in the range.

IPv6 example for GUI

In this example, there is a similar situation to the One-to-one example earlier. There is a mail server that needs to be resolved to a specific IP address in Reverse DNS look-ups. The difference in this case is the company is an early adopter of IPv6 connectivity to the Internet.

Field	Value
IP Pool Type	IPv6 Pool
Name	Mail-svr-ipv6
Comments	Registered IPv6 address for mail server
External IP Range	fd2f:50ec:cdea:0663::1025 - fd2f:50ec:cdea:0663::1025

Port block allocation example for CLI

```
config firewall ippool6
  edit Mail-svr-ipv6
    set comments "Registered IPv6 address for mail server"
    set startip fd2f:50ec:cdea:663::102
    set endip fd2f:50ec:cdea:663::1025
  end
```

Creating NAT46 IP pool and multiple (secondary) NAT64 prefixes

Policies that translate between IPv4 and IPv6 can use IPv4 address pools or IPv6 prefixes to be used in the policies, giving more options to the configuration of addresses.

NAT46

For using the ippool in NAT46 policies, first enable the use of ippools and then set the names of the ippool(s).

```
config firewall policy46
  edit 1
    set uuid e9c6ca3e-72ea-51e7-554a-1185693d03eb
    set srcintf "wan1"
    set dstintf "internal7"
    set srcaddr "external-net4"
    set dstaddr "internal-vip46"
    set action accept
    set schedule "always"
```

```
set service "ALL"
set ippool enable
set poolname "intit-pool6"
end
```

NAT64

In order to use these options in the NAT64 firewall policies the new settings `secondary-prefix status` and `secondary-prefix` options have to be configured as in the example below.

```
config system nat64
  set nat64-prefix 2001::/96
  set secondary-prefix enable
    config secondary-prefix
      edit 1
        set nat64-prefix 2002::/94
        next
      edit 2
        set nat64-prefix 2003::/95
        end
    end
end
```



The primary prefix must have a length of 96, but the secondary prefixes can be different lengths

Services

While there are a number of services already configured within FortiOS, the firmware allows for administrators to configure their own. The reasons for doing this usually fall into one or more of the following categories:

- The service is not common enough to have a standard configuration
- The service is not established enough to have a standard configuration
- The service has a standard port number but there is a reason to use a different one:
 - Port is already in use by another service
 - For security reasons, want to avoid standard port

When looking at the list of preconfigured services it may seem like there are a lot, but keep in mind that the theoretical limit for port numbers is 65,535. This gives a fairly good sized range when you are choosing what port to assign a service but there are a few points to keep in mind.

- Most of the well known ports are in the range 0 - 1023
- Most ports assigned by the Internet Corporation for Assigned Names and Numbers (ICANN) will be in the 1024 - 49151 range
- Port numbers between 49,152 and 65,535 are often used for dynamic, private or ephemeral ports.

There are 3 Service objects that can be added and configured:

- Categories
- Services
- Service Groups

Categories

In order to make sorting through the services easier, there is a field to categorize the services. Because selecting a category is part of the process for creating a new service, the configuration of categories will be explained first.

The services can be sorted into the following groups:

- General
- Web Access
- File Access
- Email
- Network Services
- Authentication
- Remote Access
- Tunneling
- VoIP, Messaging and Other Applications
- Web Proxy
- Uncategorized

The categories are for organization purposes so there is not many settings when creating a new one.

Creating a new service category

1. Go to **Policy & Objects > Services**.
2. Select **Create New**. A drop down menu is displayed. Select **Category**
3. Input a **Name** for the category.
4. Input any additional information in the **Comments** field.
5. Press **OK**.

Example

You plan on adding a number of devices such as web cameras that will allow the monitoring of the physical security of your datacenter. A number of non-standard services will have to be created and you would like to keep them grouped together under the heading of "Surveillance"

Example of a new category in the GUI

1. Go to **Policy & Objects > Objects > Services** and select **Create New > Category**.
2. Fill out the fields with the following information

Field	Value
Name	Surveillance
Comments	For DataCenter Surveillance Devices

3. Select **OK**.

Example of a New Category in the CLI

Enter the following CLI command:

```
config firewall service category
  edit Surveillance
    set comment "For DataCenter Surveillance Devices"
  end
```

To verify that the category was added correctly:

1. Go to **Policy & Objects > Objects > Services**. Select the Category Settings icon . A listing of the categories should be displayed.
2. Enter the following CLI command:

```
config firewall service category
  show
```

This should bring up all of the categories. Check to see that the new one is displayed.

Configuring a new service

Occasionally, the preconfigured list of services will not contain the needed service. There are a few variations in the creation of a service depending upon the protocol type, but the first steps in the creation of the service are common to all the variations.

To create a new service:

1. Go to **Policy & Objects > Services**.
2. Select **Create New**. A drop down menu is displayed. Select **Service**
3. Enter a name in the **Name** field for the new service
4. Include any description you would like in the **Comments** field
5. In the **Service Type** field choose between **Firewall** and **Explicit Proxy**.
6. Enable the toggle in the **Show in Service List**. If you can't see the service when you need to select it, it serves very little purpose.
7. For the **Category** field, choose the appropriate category from the **Category** drop down menu. If none is chosen, the **Uncategorized** option will be chosen by default.

Protocol options

This is the section where the configuration options of the service will differ depending on the type of protocol chosen. (The Step numbers will all continue on from the common step sequence).

The protocol options for **Firewall** service type are:

- **TCP/UDP/SCTP**
- **ICMP**
- **ICMP6**
- **IP**

The protocol options for **Proxy** service type are:

- **ALL**
- **CONNECT**

- **FTP**
- **HTTP**
- **SOCKS-TCP**
- **SOCKS-UDP**

TCP/UDP/SCTP

8. For the **Protocol Type** field, choose **TCP/UDP/SCTP** from the drop down menu
9. For the **Address** field, choose IP Range or **FQDN** (Fully Qualified Domain Name) if there is to be a specific destination for the service. Depending on which type of address is selected, the field value needs to be filled with a FQDN string or an IP address in one of the 3 standard IPv4 address formats:
 - x.x.x.x - for a specific address
 - x.x.x.x/x - for a subnet
 - x.x.x.x-x.x.x.x - for a range of specific addresses
10. Configure the **Destination Port** by:
 - Select from the drop down menu, **TCP**, **UDP** or **SCTP**
 - Enter the low end to the port range in the field indicated by grayed out **Low**.
 - Enter the high end of the port range in the field indicated by grayed out **High**. If there is only a single port in the range **High** can be left empty
 - Multiple ports or port ranges can be added by using the "+" at the beginning of the row
 - Rows can be removed by using the trash can symbol at the end of the row
11. If required, you can **Specify Source Ports** for the service by enabling the toggle switch.
 - The **Src Port** will match up with a **Destination Port**
 - **Src Ports** cannot be configured without there being a value for the **Destination Port**
 - The same rules for configuring the **Destination Ports** applies to the **Src Ports**
12. Select **OK** to confirm the configuration

Example

Example settings for a TCP protocol service. In this case, it is for an administrative connection to web servers on the DMZ. The protocol used is HTTPS which would normally use port 443, but that is already in use by another service such as Admin access to the firewall or an SSL-VPN connection.

Field	Value
Name	Example.com_WebAdmin
Comments	Admin connection to Example.com Website
Service Type	Firewall
Show in Service List	enabled
Category	Web Access

Field	Value
Protocol Options	
Protocol Type	TCP/UDP/SCTP
IP/FQDN	<left blank>
Destination Port	<ul style="list-style-type: none"> Protocol: TCP Low: 4300 High: <left blank>
Specify Source Ports	<disabled>

Creating a new TCP/UDP/SCTP service in the CLI

The following is the creation of the same service using the command line.

```
config firewall service custom
  edit Example.com_WebAdmin
    set comment "Admin connection to Example.com Website"
    set category Web Access
    set protocol TCP/UDP/SCTP
    set tcp-portrange 4300
  end
end
```

ICMP / ICMP6

- For the **Protocol Type** field, choose **ICMP** or **ICMP6** from the drop down menu
- In the **Type** field enter the appropriate type number based on the information found in "[ICMP Types and Codes](#)" on page 1 or in "[ICMPv6 Types and Codes](#)" on page 1, depending on whether the **Protocol Type** is **ICMP** or **ICMPv6**
- In the **Code** field enter the appropriate code number for the type, if applicable, based on the information found in "[ICMP Types and Codes](#)" on page 1 or in "[ICMPv6 Types and Codes](#)" on page 1, depending on whether the **Protocol Type** is **ICMP** or **ICMPv6**
- Select **OK** to confirm the configuration

Example

Example settings for an ICMP.service. In this case it has been set up for some special testing of ICMP packets.

Field	Value
Name	ICMP test #4
Comments	For testing of proprietary network scanner
Service Type	Firewall

Field	Value
Show in Service List	enabled
Category	Network Services
Protocol Options	
Protocol Type	ICMP
Type	7
Code	<left blank>

Creating a new ICMP service in the CLI

The following is the creation of the same service using the command line.

```
config firewall service custom
  edit ICMP test4
    set comment "For testing of proprietary network scanner"
    set category Network Services
    set protocol ICMP
    set icmptype 7
  end
end
```

IP

8. For the **Protocol Type** field, choose **IP** from the drop down menu
9. In the **Protocol Number** field enter the numeric value based on the information found in "Protocol Number" on [page 1](#)
10. Select **OK** to confirm the configuration

Example

Example settings for an IP.service. In this case it has been set up to communicate via an old protocol called QNX

Field	Value
Name	QNX
Comments	For QNX communications to the Development Lab
Service Type	Firewall
Show in Service List	enabled
Category	Uncategorized

Field	Value
Protocol Options	
Protocol Type	IP
Protocol Number	106

Creating a new ICMP service in the CLI

The following is the creation of the same service using the command line.

```
config firewall service custom
  edit ICMP test4
    set comment "For QNX communications to the Development Lab "
    set protocol IP
    set icmptype 106
    end
end
```



In the CLI examples, the fields for **Show in Service List**, **Service Type** and in the example for IP, **Category** were not set because the values that would have been set to were the default values and were already correctly set.

ALL/CONNECT/FTP/HTTP/SOCKS-TCP/SOCKS-UDP

These options are available only if the **Service Type** is set to **Explicit Proxy**.

8. For the **Protocol Type** field, choose one of the following from the drop down menu:
 - **ALL**
 - **CONNECT**
 - **FTP**
 - **HTTP**
 - **SOCKS-TCP**
 - **SOCKS-UDP**
9. For the **Address** field, choose IP Range or **FQDN** (Fully Qualified Domain Name) if there is to be a specific destination for the service. Depending on which type of address is selected, the field value needs to be filled with a FQDN string or an IP address in one of the 3 standard IPv4 address formats:
 - x.x.x.x - for a specific address
 - x.x.x.x/x - for a subnet
 - x.x.x.x-x.x.x.x - for a range of specific addresses
10. Configure the **Destination Port** by:
 - Enter the low end to the TCP port range in the field indicated by grayed out **Low**.
 - Enter the high end of the TCP port range in the field indicated by grayed out **High**. If there is only a single port in the range **High** can be left empty
 - Multiple ports or port ranges can be added by using the "+" at the beginning of the row
 - Rows can be removed by using the trash can symbol at the end of the row
11. If required, you can **Specify Source Ports** for the service by enabling the toggle switch.

- The **Src Port** will match up with a **Destination Port**
 - **Src Ports** cannot be configured without there being a value for the **Destination Port**
 - The same rules for configuring the **Destination Ports** applies to the **Src Ports**
12. Select **OK** to confirm the configuration

Specific addresses in TCP/UDP/SCTP

In the TCP/UDP/SCTP services it is also possible to set the parameter for a specific IP or Fully Qualified Domain Name address. The IP/FQDN field refers to the destination address of the traffic, not the source. This means for example, that you can set up a custom service that will describe in a policy the TCP traffic over port 80 going to the web site example.com, but you cannot set up a service that describes the TCP traffic over port 80 that is coming from the computer with the address 192.168.29.59.

Service groups

Just like some of the other firewall components, services can also be bundled into groups for ease of administration.

Creating a service group

1. Go to **Policy & Objects > Services**.
2. Select **Create New**. A drop down menu is displayed. Select **Service Group**
3. Input a **Group Name** to describe the services being grouped
4. Input any additional information in the **Comments** field.
5. Choose a **Type** of group. The options are **Firewall** or **Explicit Proxy**.
6. Add to the list of **Members** from the drop down menu. Using the + sign beside the field will allow the addition of multiple services.
7. Press **OK**.

Example

Example of a New Service Group:

Field	Value
Group Name	Authentication Services
Comments	Services used in Authentication
Type	Firewall

Field	Value
Members	<ul style="list-style-type: none"> • Kerberos • LDAP • LDAP_UDP • RADIUS

Firewall schedules

Firewall schedules control when policies are in effect. When you add a security policy on a FortiGate unit you need to set a schedule to determine the time frame in which that the policy will be functioning. While it is not set by default, the normal schedule would be always. This would mean that the policy that has been created is always function and always policing the traffic going through the FortiGate. The time component of the schedule is based on a 24 hour clock notation or military time as some people would say.

There are two types of schedules: One-time schedules and recurring schedules.

One-time schedule object

One-Time schedules are in effect only once for the period of time specified in the schedule. This can be useful for testing to limit how long a policy will be in effect in case it is not removed, or it can be used for isolated events such as a conference where you will only need a temporary infrastructure change for a few days.

The time frame for a One-time schedule is configured by using a start time which includes, Year | Month | Day | Hour | Minute and a Stop time which includes the same variables. So while the frequency of the schedule is only once it can last anywhere from 1 minute to multiple years.

Configuring a one-time schedule object in the GUI

1. Go to **Policy & Objects > Schedules**.
2. Select **Create New**. A drop down menu is displayed. Select **Schedule**.
3. From the **Type** options, choose **One-time**.
4. Input a **Name** for the schedule object.
5. If you which to add a **Color** to the icon in the GUI, you can click on the **Change** link to choose 1 of 32 color options.
6. Choose a **Start Date**.

Selecting the field with the mouse will bring up a interactive calendar graphic that will allow the user to select the date. The date can also be typed in using the format YYYY/MM/DD.

7. Choose a **Start Time**.

The **Start Time** is composed of two fields, **Hour** and **Minute**. Think of setting the time for a digital clock in 24 hour mode. The **Hour** value can be an integer from 0 and 23. The **Minute** value can be from 0 to 59. 0 and 0 would be midnight at the start of the day and 23 and 59 would be one minute to midnight at the end of the day. The value can be entered by keyboard or by using the up and down arrows in the field to select the value.

6. Choose an **End Date**.

Configuration is the same as **Start Date**.

8. Choose a **Stop Time.**

Configuration is the same as **Start Time**.

9. Enable/Disable **Pre-expiration event log.**

This configures the system to create an event log 1 to 100 days before the **End Date** as a warning in case the schedule needs to be extended.

10. If the **Pre-expiration event log is enabled, set the value for **Number of days before**.****11. Press **OK**.****Example: Firewall schedule - one-time**

The company wants to change over their web site image to reference the new year. They have decided to take this opportunity to do some hardware upgrades as well. Their web site is business oriented so they have determined that over New Year's Eve there will be very limited traffic.

- They are going to need a maintenance window of 2 hours bracketing midnight on New Year's Eve.

Configuration in the GUI**1. Go to **Policy & Objects > Objects > Schedule**.****2. Select **Create New > Schedule**.****3. Fill out the fields with the following information:**

Type	One-time
Name	NewYearsEve_Maintenance
Start Date	2014/12/31 <use the built in calendar>
End Date	2015/01/01 <use the built in calendar>
Start Time	Hour: 23, Minute: 0
Stop Time	Hour: 1Minute: 0
Pre-expiration event log	<disable>

4. Select **OK.****To verify that the schedule was added correctly:****1. Go to **Policy & Objects > Objects > Schedule**.****2. Check that the schedule with the name you used has been added to the list of recurring schedules and that the listed settings are correct.****Configuration in the CLI****1. Enter the following CLI command:**

```
config firewall schedule onetime
    edit maintenance_window
        set start 23:00 2012/12/31
        set end 01:00 2013/01/01
    next
```

end

To verify that the schedule was added correctly:

1. Enter the following CLI command:

```
config firewall schedule onetime
    edit <the name of the schedule you wish to verify>
        show full-configuration
```

Recurring schedule object

Recurring schedules are in effect repeatedly at specified times of specified days of the week. The Recurring schedule is based on a repeating cycle of the days of the week as opposed to every x days or days of the month. This means that you can configure the schedule to be in effect on Tuesday, Thursday, and Saturday but not every 2 days or on odd numbered days of the month.

If a recurring schedule has a stop time that is earlier than the start time, the schedule will take effect at the start time but end at the stop time on the next day. You can use this technique to create recurring schedules that run from one day to the next.

Configuring a recurring schedule object in the GUI

1. Go to **Policy & Objects > Schedules**.
2. Select **Create New**. A drop down menu is displayed. Select **Schedule**.
3. From the **Type** options, choose **Recurring**.
4. Input a **Name** for the schedule object.
5. If you which to add a **Color** to the icon in the GUI, you can click on the **Change** link to choose 1 of 32 color options.
6. From the **Days** options, choose the day of the week that you would like this schedule to apply to. The schedule will be in effect on the days of the week that have a check mark in the checkbox to the left of the name of the weekday.
7. If the scheduled time is the whole day, leave the **All Day** toggle switch enabled. If the schedule is for specific times during the day, disable the **All Day** toggle switch.
8. If the All Day option is disabled, choose a **Start Time**.

The **Start Time** is composed of two fields, **Hour** and **Minute**. Think of setting the time for a digital clock in 24 hour mode. The **Hour** value can be an integer from 0 and 23. The **Minute** value can be from 0 to 59. 0 and 0 would be midnight at the start of the day and 23 and 59 would be one minute to midnight at the end of the day. The value can be entered by keyboard or by using the up and down arrows in the field to select the value.

7. Choose a **Stop Time**.
Configuration is the same as **Start Time**.
8. Press **OK**.



Because recurring schedules do not work with DENY policies, the strategy when designing a schedule should *not* be to determine when users cannot access a policy but to build the schedules around when it *is* possible to access the policy.

Example: Firewall schedule - recurring

The Company wants to allow the use of Facebook by employees, but only during none business hours and the lunch break.

- The business hours are 9:00 p.m. to 6:00 p.m.
- The Lunch break is 12:00 p.m. to 1:00 p.m.
- The plan is to create a schedule to cover the morning business hours and the afternoon business hours and block access to the Facebook web site during that time.

Configuration in the GUI

1. Go to **Policy & Objects > Objects > Schedule**.
2. Select **Create New > Schedule**.
3. Fill out the fields with the following information:

Type	Recurring
Name	Morning_Business_Hours
Days	Monday, Tuesday, Wednesday, Thursday, Friday
Start Time	Hour = 9, Minute = 0
Stop Time	Hour = 12, Minute = 0

4. Select **OK**.
5. Create a second new schedule.

Type	Recurring
Name	Morning_Business_Hours
Days	Monday, Tuesday, Wednesday, Thursday, Friday
Start Time	Hour = 13, Minute = 0
Stop Time	Hour = 18, Minute = 0

6. Select **OK**.

To verify that the schedule was added correctly:

1. Go to **Policy & Objects > Objects > Schedule**.
2. Check that the schedule with the name you used has been added to the list of recurring schedules and that the listed settings are correct.

Configuration in the CLI

1. Enter the following CLI command:

```
config firewall schedule recurring
    edit Morning_Business_Hours
```

```
set day monday tuesday wednesday thursday friday
set start 09:00
set end 12:00
end
```

2. Enter the following CLI command:

```
config firewall schedule recurring
edit Afternoon_Business_Hours
    set day monday tuesday wednesday thursday friday
    set start 13:00
    set end 18:00
end
```

To verify that the schedule was added correctly:

1. Enter the following CLI command:

```
config firewall schedule recurring
edit <the name of the schedule you wish to verify>
    show full-configuration
```

Schedule groups

You can organize multiple firewall schedules into a schedule group to simplify your security policy list. The schedule parameter in the policy configuration does not allow for the entering of multiple schedules into a single policy so if you have a combination of time frames that you want to schedule the policy for then the best approach, rather than making multiple policies is to use a schedule group.

Creating a schedule group object

1. Go to **Policy & Objects > Schedules**.
2. Select **Create New**. A drop down menu is displayed. Select **Schedule Group**
3. Input a **Name** for the schedule object.
4. In the **Members** field, select the "+" to bring forth the panel for selecting entries.
5. Press **OK**.

Example

Your Internet policy allows employees to visit Social Media sites from company computers but not during what is considered working hours. The offices are open a few hours before working hours and the doors are not locked until a few hours after official closing so work hours are from 9 to 5 with a lunch break from Noon to 1:00 p.m.

Your approach is to block the traffic between 9 and noon and between 1:00 p.m. and 5:00 p.m. This means you will need two schedules for a single policy and the schedule group handles this for you. Schedule groups can contain both recurring and one-time schedules. Schedule groups cannot contain other schedule groups.

Schedule expiration

The schedule in a security policy enables certain aspects of network traffic to occur for a specific length of time. What it does not do however, is police that time. That is, the policy is active for a given time frame, and as long as the session is open, traffic can continue to flow.

For example, in an office environment, Skype use is allowed between noon and 1pm. During that hour, any Skype traffic continues. As long as that session is open, after the 1pm end time, the Skype conversations can continue, yet new sessions will be blocked. Ideally, the Skype session should close at 1pm.

Using a CLI command you can set the schedule to terminate all sessions when the end time of the schedule is reached. Within the config firewall command enter the command:

```
set schedule-timeout enable
```

By default, this option is set to disable.

A few further settings are needed to make this work.

```
config firewall policy
  edit ID
    set firewall-session-dirty check-new
    end

config system settings
  set firewall-session-dirty check-policy-option
  end
```



The Policy window will indicate when a policy has become invalid due to its schedule parameters referring only to times in the past.

Firewall-session-dirty setting

The firewall-session-dirty setting has three options

check-all	CPU flushes all current sessions and re-evaluates them. [default]
check-new	CPU keeps existing sessions and applies policy changes to new sessions only. This reduces CPU load and the possibility of packet loss.
check-policy-option	Use the option selected in the firewall-session-dirty field of the firewall policy (check-all or check-new, as above, but per policy).

Secure web gateway, WAN optimization, web caching and WCCP

You can use FortiGate WAN optimization and web caching to improve performance and security of traffic passing between locations on your wide area network (WAN) or from the Internet to your web servers. You can also use the FortiGate unit as an explicit FTP and web proxy server. If your FortiGate unit supports web caching, you can also add web caching to any HTTP sessions including WAN optimization, explicit web proxy and other HTTP sessions.

the next sections of this document describes how FortiGate WAN optimization, web caching, explicit web proxy, explicit FTP proxy and WCCP work and also describes how to configure these features.

Before you begin

Before you begin to configure WAN optimization, Web caching, explicit proxies or WCCP, take a moment to note the following:

- To use WAN optimization and web caching, your FortiGate unit must support these features and not all do. In general your FortiGate unit must include a hard disk to support these features. See "[FortiGate models that support WAN optimization](#)" on page 263. Most FortiGate units support Explicit Web and FTP proxies.
- To be able to configure WAN optimization and web caching from the web manager you should begin by going to **System > Feature Visibility** and turning on **WAN Opt. & Cache**.
- To be able to configure the Web and FTP proxies from the web manager you should begin by going to **System > Feature Visibility** and turning on **Explicit Proxy**.
- If you enable virtual domains (VDOMs) on the FortiGate unit, WAN optimization, web caching, and the explicit web and FTP proxies are available separately for each VDOM.
- This guide is based on the assumption that you are a FortiGate administrator. It is not intended for others who may also use the FortiGate unit, such as FortiClient administrators or end users.
- FortiGate WAN optimization is proprietary to Fortinet. FortiGate WAN optimization will not work with other vendors' WAN optimization or acceleration features.
- FortiGate web caching, explicit web and FTP proxies, and WCCP support known standards for these features. See the appropriate chapters of this document for details.

At this stage, the following installation and configuration conditions are assumed:

- For WAN optimization you have already successfully installed two or more FortiGate units at various locations across your WAN.
- For web caching, the explicit proxies and WCCP you have already successfully installed one or more FortiGate units on your network.
- You have administrative access to the web-based manager and/or CLI.
- The FortiGate units are integrated into your WAN or other networks
- The operation mode has been configured.
- The system time, DNS settings, administrator password, and network interfaces have been configured.
- Firmware, FortiGuard Antivirus and FortiGuard Antispam updates are completed.

- Your Fortinet products have been registered. Register your Fortinet products at the Fortinet Technical Support web site, <https://support.fortinet.com>.

FortiGate models that support WAN optimization

WAN optimization is available on FortiGate models with internal storage that also support SSL acceleration. Internal storage includes high-capacity internal hard disks, AMC hard disk modules, FortiGate Storage Modules (FSMs) or over 4 Gbytes of internal flash storage. All of these storage locations can provide similar web caching and byte caching performance. If you add more than one storage location (for example, by creating multiple partitions on a storage device, by using more than one FSM, or by using an FSM and AMC hard disk in the same FortiGate unit) you can configure different storage locations for web caching and byte caching.

Distributing WAN optimization, explicit proxy, and web caching to multiple CPU cores

By default WAN optimization, explicit proxy and web caching is handled by half of the CPU cores in a FortiGate unit. For example, if your FortiGate unit has 4 CPU cores, by default two will be used for WAN optimization, explicit proxy and web caching. You can use the following command to change the number of CPU cores that are used.

```
config system global
    set wad-worker-count <number>
end
```

The value for <number> can be between 1 and the total number of CPU cores in your FortiGate unit. Adding more cores may enhance WAN optimization, explicit proxy and web caching performance and reduce the performance of other FortiGate systems.

Dispatching traffic to WAD worker based on source affinity

The `wad-worker` balancing algorithm supports a more balanced dispersal of traffic to the wad processes even, if the bulk of the traffic is coming from a small set of, or single source.

By default, dispatching traffic to WAD workers is based on source affinity. This may negatively affect performance when users have another explicit proxy in front of the FortiGate. Source affinity causes the FortiGate to process the traffic as if it originated from the single (or small set of) ip address of the outside proxy. This results in the use of one, or a small number, of WAD processes.

By disabling `wad-source-affinity` the traffic is balanced over all of the WAD processes. When the `wad-source-affinity` is disabled, the WAD dispatcher will not assign the traffic based on the source IP, but will assign the traffic to available workers in a round-robin fashion.

Handling the traffic by different WAD workers results in losing some of the benefits of using source affinity, as is explained by the warning message that appears when it is disabled:



"WARNING: Disabling this option results in some features to be unsupported. IP-based user authentication, disclaimer messages, security profile override, authentication cookies, MAPI scanning, and some video caches such as YouTube are not supported.

Do you want to continue? (y/n)"

CLI

```
config system global
  set wad-source-affinity {enable|disable}
end
```

Toggling disk usage for logging or wan-opt

Both logging and WAN Optimization use hard disk space to save data. In FortiOS, you cannot use the same hard disk for WAN Optimization and logging.

- If the FortiGate has one hard disk, then it can be used for either disk logging or WAN optimization, but not both. By default, the hard disk is used for disk logging.
- If the FortiGate has two hard disks, then one disk is always used for disk logging and the other disk is always used for WAN optimization.

On the FortiGate, go to **System > Advanced > Disk Settings** to switch between **Local Log** and **WAN Optimization**.

You can also change disk usage from the CLI using the following command:

```
configure system global
  set disk-usage {log | wanopt}
end
```



The Toggle Disk Usage feature is supported on all new "E" Series models, while support for "D" Series models may vary.

Please refer to the [Feature Platform Matrix](#) for more information.



Changing the disk setting formats the disk, erases current data stored on the disk and disables either disk logging or WAN Optimization.

You can configure WAN Optimization from the CLI or the GUI. To configure WAN Optimization from the GUI you must go to **System > Feature Visibility** and turn on WAN Optimization.



Remote logging (including logging to FortiAnalyzer and remote Syslog servers) is not affected by using the single local hard disk for WAN Optimization.

Enabling WAN optimization affects more than just disk logging

In addition to affecting WAN Optimization, the following table shows other features affected by the FortiGate disk configuration.

Features affected by Disk Usage as per the number of internal hard disks on the FortiGate

Feature	Logging Only (1 hard disk)	WAN Opt. Only (1 hard disk)	Logging & WAN Opt. (2 hard disks)
Logging	Supported	Not supported	Supported
Report/Historical FortiView	Supported	Not supported	Supported
Firewall Packet Capture (Policy Capture and Interface Capture)	Supported	Not supported	Supported
AV Quarantine	Supported	Not supported	Supported
IPS Packet Capture	Supported.	Not supported	Supported
DLP Archive	Supported	Not supported	Supported
Sandbox DB & Results	FortiSandbox database and results are also stored on disk, but will not be affected by this feature.		

Example topologies relevant to WAN optimization

FortiGate WAN optimization consists of a number of techniques that you can apply to improve the efficiency of communication across your WAN. These techniques include protocol optimization, byte caching, web caching, SSL offloading, and secure tunneling. Protocol optimization can improve the efficiency of traffic that uses the CIFS, FTP, HTTP, or MAPI protocol, as well as general TCP traffic. Byte caching caches files and other data on FortiGate units to reduce the amount of data transmitted across the WAN. Web caching stores web pages on FortiGate units to reduce latency and delays between the WAN and web servers. SSL offloading offloads SSL decryption and encryption from web servers onto FortiGate SSL acceleration hardware. Secure tunneling secures traffic as it crosses the WAN.

You can apply different combinations of these WAN optimization techniques to a single traffic stream depending on the traffic type. For example, you can apply byte caching and secure tunneling to any TCP traffic. For HTTP and HTTPS traffic, you can also apply protocol optimization and web caching.

You can configure a FortiGate unit to be an explicit web proxy server for both IPv4 and IPv6 traffic and an explicit FTP proxy server. Users on your internal network can browse the Internet through the explicit web proxy server or connect to FTP servers through the explicit FTP proxy server. You can also configure these proxies to protect access to web or FTP servers behind the FortiGate unit using a reverse proxy configuration.

Web caching can be applied to any HTTP or HTTPS traffic, this includes normal traffic accepted by a security policy, explicit web proxy traffic, and WAN optimization traffic.

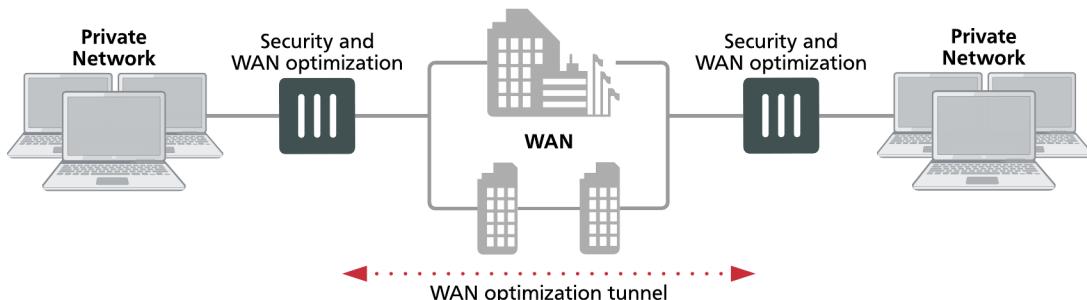
You can also configure a FortiGate unit to operate as a Web Cache Communication Protocol (WCCP) client or server. WCCP provides the ability to offload web caching to one or more redundant web caching servers.

FortiGate units can also apply security profiles to traffic as part of a WAN optimization, explicit web proxy, explicit FTP proxy, web cache and WCCP configuration. Security policies that include any of these options can also include settings to apply all forms of security profiles supported by your FortiGate unit.

Basic WAN optimization topology

The basic FortiGate WAN optimization topology consists of two FortiGate units operating as WAN optimization peers intercepting and optimizing traffic crossing the WAN between the private networks.

Security device and WAN optimization topology



FortiGate units can be deployed as security devices that protect private networks connected to the WAN and also perform WAN optimization. In this configuration, the FortiGate units are configured as typical security devices for the private networks and are also configured for WAN optimization. The WAN optimization configuration intercepts traffic to be optimized as it passes through the FortiGate unit and uses a WAN optimization tunnel with another FortiGate unit to optimize the traffic that crosses the WAN.

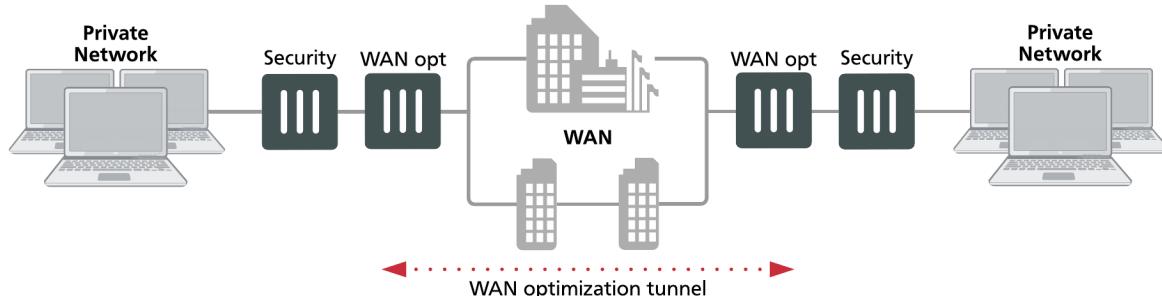
You can also deploy WAN optimization on single-purpose FortiGate units that only perform WAN optimization. In the out of path WAN optimization topology shown below, FortiGate units are located on the WAN outside of the private networks. You can also install the WAN optimization FortiGate units behind the security devices on the private networks.

The WAN optimization configuration is the same for FortiGate units deployed as security devices and for single-purpose WAN optimization FortiGate units. The only differences would result from the different network topologies.

Out-of-path WAN optimization topology

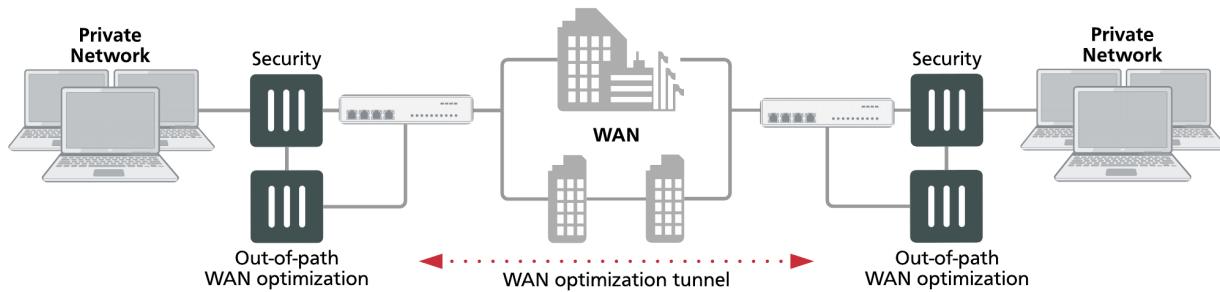
In an out-of-path topology, one or both of the FortiGate units configured for WAN optimization are not directly in the main data path. Instead, the out-of-path FortiGate unit is connected to a device on the data path, and the device is configured to redirect sessions to be optimized to the out-of-path FortiGate unit.

Single-purpose WAN optimization topology



The following out-of-path FortiGate units are configured for WAN optimization and connected directly to FortiGate units in the data path. The FortiGate units in the data path use a method such as policy routing to redirect traffic to be optimized to the out-of-path FortiGate units. The out-of-path FortiGate units establish a WAN optimization tunnel between each other and optimize the redirected traffic.

Out-of-path WAN optimization



One of the benefits of out-of-path WAN optimization is that out-of-path FortiGate units only perform WAN optimization and do not have to process other traffic. An in-path FortiGate unit configured for WAN optimization also has to process other non-optimized traffic on the data path.

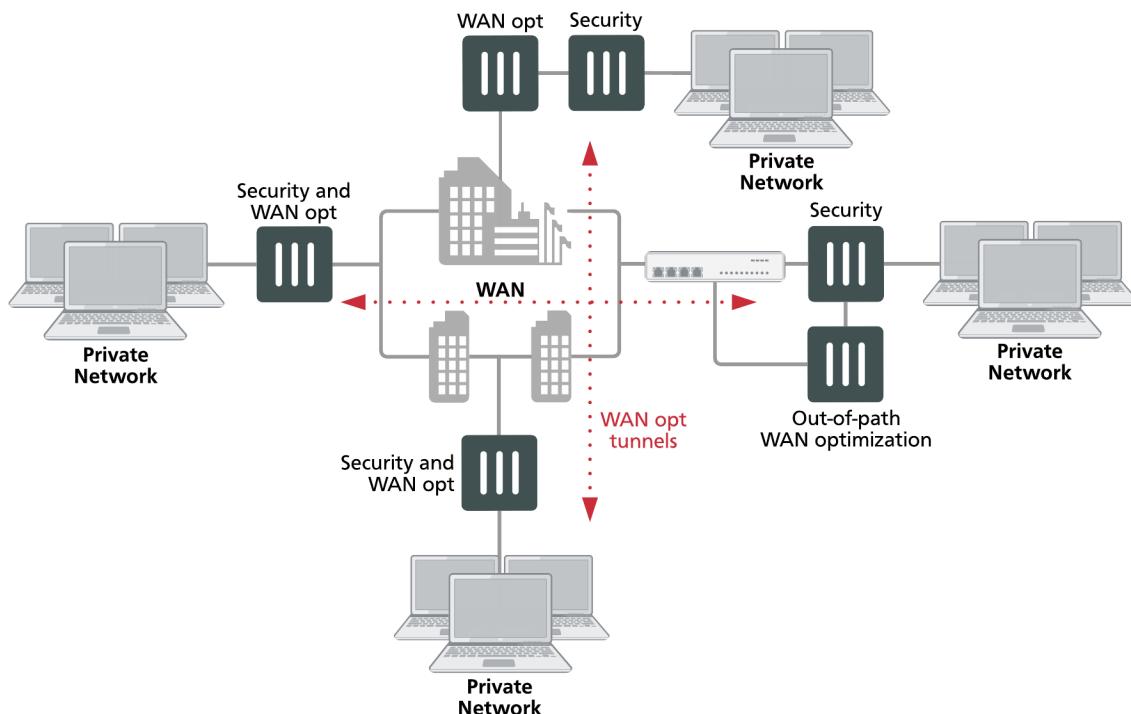
The out-of-path FortiGate units can operate in NAT or transparent mode.

Other out-of-path topologies are also possible. For example, you can install the out-of-path FortiGate units on the private networks instead of on the WAN. Also, the out-of-path FortiGate units can have one connection to the network instead of two. In a one-arm configuration such as this, security policies and routing have to be configured to send the WAN optimization tunnel out the same interface as the one that received the traffic.

Topology for multiple networks

As shown in below, you can create multiple WAN optimization configurations between many private networks. Whenever WAN optimization occurs, it is always between two FortiGate units, but you can configure any FortiGate unit to perform WAN optimization with any of the other FortiGate units that are part of your WAN.

WAN optimization among multiple networks

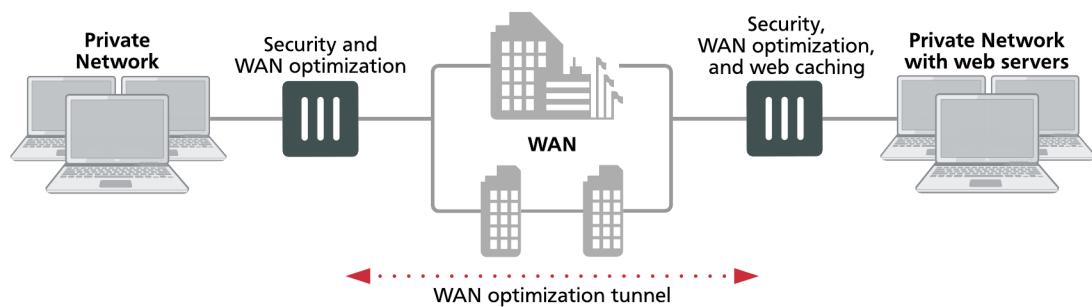


You can also configure WAN optimization between FortiGate units with different roles on the WAN. FortiGate units configured as security devices and for WAN optimization can perform WAN optimization as if they are single-purpose FortiGate units just configured for WAN optimization.

WAN optimization with web caching

You can add web caching to a WAN optimization topology when users on a private network communicate with web servers located across the WAN on another private network.

WAN optimization with web caching topology

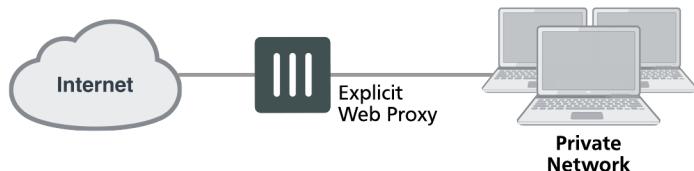


The topology above is the same as that shown in [WAN optimization with web caching on page 269](#) with the addition of web caching to the FortiGate unit in front of the private network that includes the web servers. You can also add web caching to the FortiGate unit that is protecting the private network. In a similar way, you can add web caching to any WAN Optimization topology.

Explicit web proxy topologies

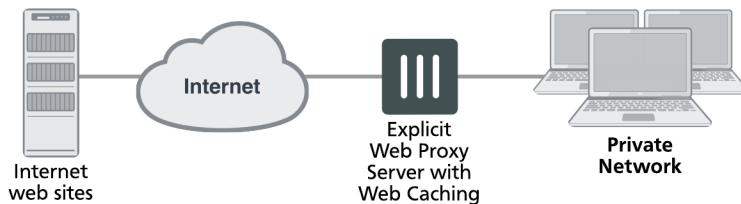
You can configure a FortiGate unit to be an explicit web proxy server for Internet web browsing of IPv4 and IPv6 web traffic. To use the explicit web proxy, users must add the IP address of the FortiGate interface configured for the explicit web proxy to their web browser proxy configuration.

Explicit web proxy topology



If the FortiGate unit supports web caching, you can also add web caching to the security policy that accepts explicit web proxy sessions. The FortiGate unit then caches Internet web pages on a hard disk to improve web browsing performance.

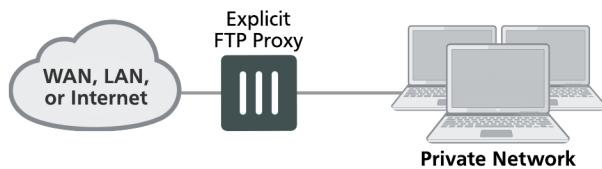
Explicit web proxy with web caching topology



Explicit FTP proxy topologies

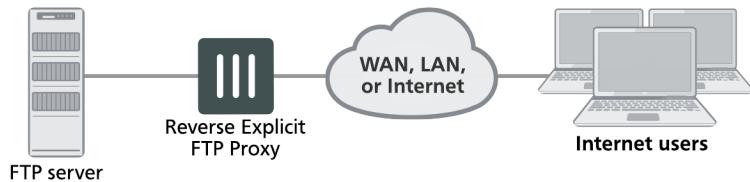
You can configure a FortiGate unit to be an explicit FTP proxy server for FTP users. To use the explicit web proxy, FTP users must connect to and authenticate with the explicit FTP proxy before connecting to an FTP server.

Explicit FTP proxy topology



You can also configure reverse explicit FTP proxy. In this configuration, users on the Internet connect to the explicit web proxy before connecting to an FTP server installed behind a FortiGate unit.

Reverse explicit FTP proxy topology

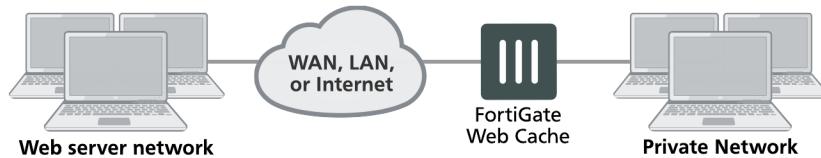


Web caching topologies

FortiGate web caching can be added to any security policy and any HTTP or HTTPS traffic accepted by that security policy can be cached on the FortiGate unit hard disk. This includes WAN optimization and explicit web proxy traffic. The network topologies for these scenarios are very similar. They involved a FortiGate unit installed between users and web servers with web caching enabled.

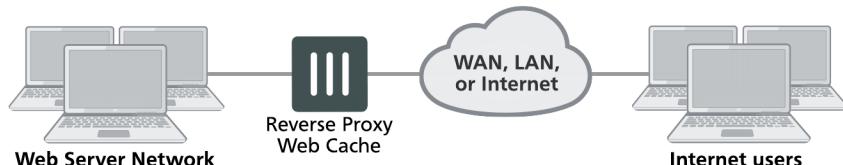
A typical web-caching topology includes one FortiGate unit that acts as a web cache server. Web caching is enabled in a security policy and the FortiGate unit intercepts web page requests accepted by the security policy, requests web pages from the web servers, caches the web page contents, and returns the web page contents to the users. When the FortiGate unit intercepts subsequent requests for cached web pages, the FortiGate unit contacts the destination web server just to check for changes.

Web caching topology



You can also configure reverse proxy web-caching. In this configuration, users on the Internet browse to a web server installed behind a FortiGate unit. The FortiGate unit intercepts the web traffic (HTTP and HTTPS) and caches pages from the web server. Reverse proxy web caching on the FortiGate unit reduces the number of requests that the web server must handle, leaving it free to process new requests that it has not serviced before.

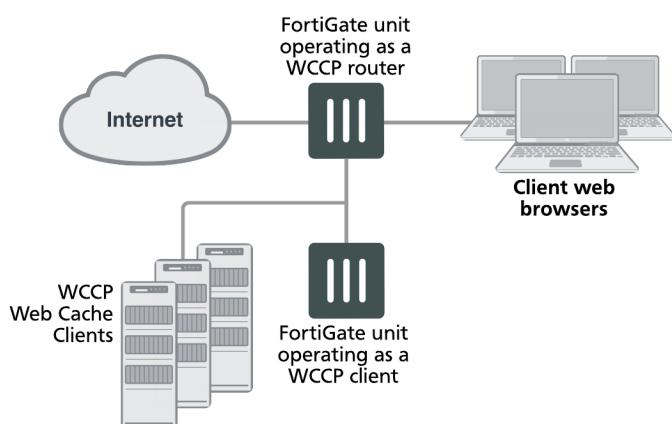
Reverse proxy web caching topology



WCCP topologies

You can operate a FortiGate unit as a Web Cache Communication Protocol (WCCP) router or cache engine. As a router, the FortiGate unit intercepts web browsing requests from client web browsers and forwards them to a WCCP cache engine. The cache engine returns the required cached content to the client web browser. If the cache server does not have the required content it accesses the content, caches it and returns the content to the client web browser.

WCCP topology



FortiGate units can also operate as WCCP cache servers, communicating with WCCP routers, caching web content and providing it to client web browsers as required.

WCCP is transparent to client web browsers. The web browsers do not have to be configured to use a web proxy.

Inside FortiOS: WAN optimization

Enterprises deploying FortiOS can leverage WAN optimization to provide fast and secure application responses between locations on a Wide Area Network (WAN). The web caching component of FortiOS WAN optimization extends this protection and performance boost to cloud services.

Centralize without compromising your WAN performance

Many multi-location enterprise environments reduce costs and consolidate resources by centralizing applications or providing applications in the cloud. Efficient and high-speed communication between applications and their users is critical. Remote sites don't always have access to high bandwidth, but users at all sites expect consistent network performance. Minimizing user impact and improving performance is especially vital when applications designed for local area networks (LANs) are on the cloud.

Even applications that work fine on a local LAN, such as Windows File Sharing (CIFS), email exchange (MAPI), and many others, suffer from bandwidth limitations and latency issues when accessed over a WAN. This results in a loss of productivity and a perceived need for expensive network upgrades. FortiOS's WAN Optimization provides an inexpensive and easy way to deploy a solution to this problem.

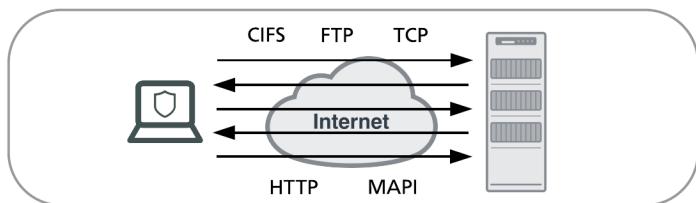
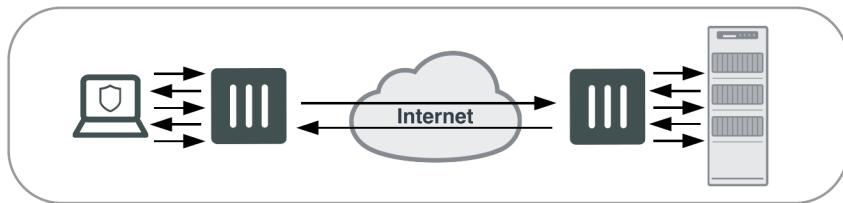
FortiOS is commonly deployed in central offices, satellite offices, and in the cloud to provide secure communications across a WAN using IPsec or SSL VPN. This installed infrastructure can be leveraged to add more value by using WAN Optimization to accelerate WAN traffic and web caching to accelerate cloud services.

FortiOS WAN optimization

FortiOS includes license-free WAN optimization on most current FortiGate devices. WAN optimization is a comprehensive solution that maximizes your WAN performance and provides intelligent bandwidth management and unmatched consolidated security performance. WAN optimization reduces your network overhead and removes unnecessary traffic for a better overall performance experience. Efficient use of bandwidth and better application performance will remove the need for costly WAN link upgrades between data centers and other expensive solutions for your network traffic growth.

Protocol optimization

Protocol optimization is effective for applications designed for the LAN that do not function well on low bandwidth high latency networks. FortiOS protocol optimization improves the efficiency of CIFS, FTP, HTTP, MAPI, and general TCP sessions.

Regular bandwidth usage**Improved bandwidth usage with FortiGate protocol optimization**

For example, CIFS, which is a fairly “chatty” protocol, requires many background transactions to successfully transfer a single file. When transferring the file, CIFS sends small chunks of data and waits sequentially for each chunk’s arrival and acknowledgement before sending the next. This large amount of request/acknowledgement traffic can delay transfers. FortiOS CIFS WAN Optimization removes this chattiness and gets on with the job of transferring the file.

TCP protocol optimization uses techniques such as SACK support, window scaling and window size adjustment, and connection pooling to remove common WAN TCP bottlenecks.

Web caching

In an enterprise environment, multiple users will often want to get the same content (for example, a sales spreadsheet, a corporate presentation or a PDF from a cloud service, or a software update). With FortiOS Web caching, content from the cloud, from the web or from other sites on the WAN is download once and cached on the local FortiGate device. When other users access the same content they download it from the cache. The result is less bandwidth use and reduced latency for the file requester.

FortiOS web caching also recognizes requests for Windows or MS-Office updates and downloads the new update file in the background. Once downloaded to the cache, the new update file is available to all users and all subsequent requests for this update are rapidly downloaded from the cache.

Byte caching

Byte caching improves caching by accelerating the transfer of similar, but not identical content. Byte caching accelerates multiple downloads of different email messages with the same corporate disclaimer by downloading the disclaimer over the WAN once and then downloading all subsequent disclaimers from a local FortiGate unit. Byte caching reduces the amount of data crossing the WAN when multiple different emails with the same or similar attachments or different versions of an attachment are downloaded from a corporate email server to different locations over the WAN.

Dynamic data chunking

Dynamic data chunking detects and optimizes persistent data chunks in changed files or in data embedded in traffic that uses an unknown protocol. For example, dynamic chunking can cache data in Lotus notes traffic and make the data chunks available for email and other protocols.

Data deduplication

Byte caching breaks large units of application data, like an email attachment or a file download, into manageable small chunks of data. Each chunk of data is labeled with a hash, and chunks with their respective hashes are stored in a database on the local FortiGate unit. When a remote user request a file, the WAN Optimization sends the hashes, rather than the actual data. The FortiGate unit at the other end of the WAN tunnel reassembles the data from its own hash database, only downloading chunks that it is missing. Deduplication, or the process of eliminating duplicate data, will reduce space consumption. In addition to reducing the amount of data downloaded across the WAN, byte caching is not application specific and assists by accelerating all of the protocols supported by WAN Optimization.

Server monitoring and management

The health and performance of real servers can be monitored from the FortiGate GUI. Virtual servers and their assigned real servers can be monitored for health status, if there have been any monitor events, number of active sessions, round trip time and number of bytes processed. Should a server become problematic and require administration, it can be gracefully removed from the Real Server pool to enable disruption free maintenance. When a removed real server is able to operate it can gracefully be added back to the virtual server.

SSL acceleration

SSL is used by many organizations to keep WAN communications private. WAN Optimization boosts SSL acceleration properties of FortiGate FortiASIC hardware by accelerating SSL traffic across the WAN. The FortiGate unit handles SSL encryption/decryption for corporate servers providing SSL encrypted connections over the WAN.

VPN replacement

FortiOS WAN optimization supports secure SSL-encrypted tunnels between FortiGate units on the WAN. Employing secure WAN Optimization tunnels can replace IPsec VPNs between sites. The result is a single, relatively simple configuration that supports optimization and privacy of communication across the WAN and uses FortiGate SSL acceleration to provide high performance.

Road warriors and home workers

The drive to give employees greater flexibility and reduce operational costs has led to more remote workers, both at home and on the road. Whether accessing the office from a hotel, public wireless hotspot, or home, the problem is the same: low bandwidth and high latency harming application performance. WAN Optimization is integrated into FortiClient, which can be installed on PCs and wireless devices to optimize communication between remote workers and their offices.

Reduce your...

- **Capital outlay:** Organizations only need to purchase a single device per location.
- **Licensing costs:** WAN Optimization is included with FortiOS. Additional licenses are not needed.
- **Network complexity:** Small offices that may not have the space or power connections for multiple devices do not need to worry: no additional devices are required.

WAN optimization concepts

Client/server architecture

Traffic across a WAN typically consists of clients on a client network communicating across a WAN with a remote server network. The clients do this by starting communication sessions from the client network to the server network. These communication sessions can be open text over the WAN or they can be encrypted by SSL VPN or IPsec VPN.

To optimize these sessions, you can add **WAN optimization security policies** to the **client-side FortiGate unit** to accept sessions from the client network that are destined for the server network. The client-side FortiGate unit is located between the client network and the WAN. WAN optimization security policies include **WAN optimization profiles** that control how the traffic is optimized.

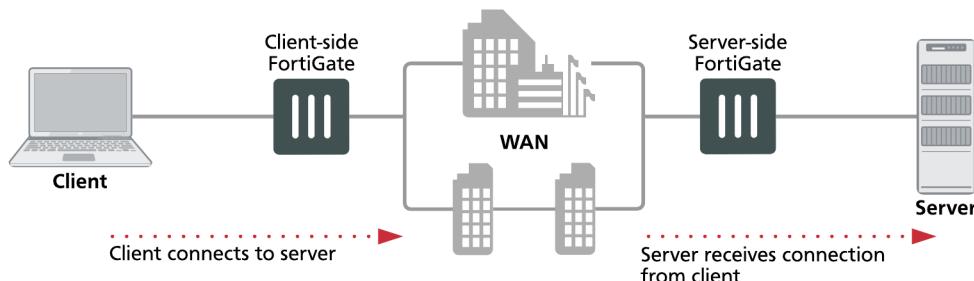
The client-side FortiGate unit must also include the IP address of the **server-side FortiGate unit** in its WAN optimization **peer** configuration. The server-side FortiGate unit is located between the server network and the WAN. The peer configuration allows the client-side FortiGate unit to find the server-side FortiGate unit and attempt to establish a WAN optimization **tunnel** with it.

For the server-side FortiGate unit you must add a security policy with **wanopt** as the **Incoming Interface**. This security policy allows the FortiGate unit to accept WAN optimization sessions from the client-side FortiGate unit. For the server-side FortiGate unit to accept a WAN optimization connection it must have the client-side FortiGate unit in its WAN optimization peer configuration.



WAN optimization profiles are only added to the client-side WAN optimization security policy. The server-side FortiGate unit employs the WAN optimization settings set in the WAN optimization profile on the client-side FortiGate unit.

Client/server architecture



When both peers are identified the FortiGate units attempt to establish a WAN optimization **tunnel** between them. WAN optimization tunnels use port 7810. All optimized data flowing across the WAN between the client-side and server-side FortiGate units use this tunnel. WAN optimization tunnels can be encrypted use SSL encryption to keep the data in the tunnel secure.

Any traffic can be sent through a WAN optimization tunnel. This includes SSL and IPsec VPN traffic. However, instead of configuring SSL or IPsec VPN for this communication you can add SSL encryption using the WAN optimization tunnel.

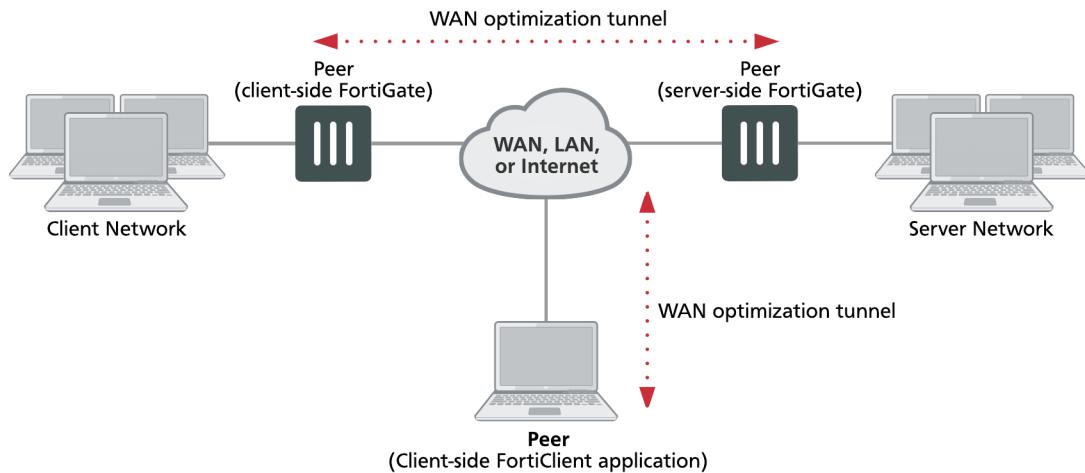
In addition to basic identification by peer host ID and IP address you can configure WAN optimization **authentication** using certificates and pre-shared keys to improve security. You can also configure FortiGate units involved in WAN optimization to accept connections from any identified peer or restrict connections to specific peers.

The FortiClient application can act in the same manner as a client-side FortiGate unit to optimize traffic between a computer running FortiClient and a FortiGate unit.

WAN optimization peers

The client-side and server-side FortiGate units are called WAN optimization peers because all of the FortiGate units in a WAN optimization network have the same peer relationship with each other. The client and server roles just relate to how a session is started. Any FortiGate unit configured for WAN optimization can be a client-side and a server-side FortiGate unit at the same time, depending on the direction of the traffic. Client-side FortiGate units initiate WAN optimization sessions and server-side FortiGate units respond to the session requests. Any FortiGate unit can simultaneously be a client-side FortiGate unit for some sessions and a server-side FortiGate unit for others.

WAN optimization peer and tunnel architecture



To identify all of the WAN optimization peers that a FortiGate unit can perform WAN optimization with, you add host IDs and IP addresses of all of the peers to the FortiGate unit configuration. The peer IP address is actually the IP address of the peer unit interface that communicates with the FortiGate unit.

Protocol optimization

Protocol optimization techniques optimize bandwidth use across the WAN. These techniques can improve the efficiency of communication across the WAN optimization tunnel by reducing the amount of traffic required by

communication protocols. You can apply protocol optimization to Common Internet File System (CIFS), FTP, HTTP, MAPI, and general TCP sessions. You can apply general TCP optimization to MAPI sessions.

For example, CIFS provides file access, record locking, read/write privileges, change notification, server name resolution, request batching, and server authentication. CIFS is a fairly “chatty” protocol, requiring many background transactions to successfully transfer a single file. This is usually not a problem across a LAN. However, across a WAN, latency and bandwidth reduction can slow down CIFS performance.

When you select the CIFS protocol in a WAN optimization profile, the FortiGate units at both ends of the WAN optimization tunnel use a number of techniques to reduce the number of background transactions that occur over the WAN for CIFS traffic.

If a policy accepts a range of different types of traffic, you can set **Protocol** to **TCP** to apply general optimization techniques to TCP traffic. However, applying this TCP optimization is not as effective as applying more protocol-specific optimization to specific types of traffic. TCP protocol optimization uses techniques such as TCP SACK support, TCP window scaling and window size adjustment, and TCP connection pooling to remove TCP bottlenecks.

Protocol optimization and MAPI

By default the MAPI service uses port number 135 for RPC port mapping and may use random ports for MAPI messages. The random ports are negotiated through sessions using port 135. The FortiOS DCE-RPC session helper learns these ports and opens pinholes for the messages. WAN optimization is also aware of these ports and attempts to apply protocol optimization to MAPI messages that use them. However, to configure protocol optimization for MAPI you should set the WAN optimization profile to a single port number (usually port 135). Specifying a range of ports may reduce performance.

Byte caching

Byte caching breaks large units of application data (for example, a file being downloaded from a web page) into small chunks of data, labeling each chunk of data with a hash of the chunk and storing those chunks and their hashes in a database. The database is stored on a WAN optimization storage device. Then, instead of sending the actual data over the WAN tunnel, the FortiGate unit sends the hashes. The FortiGate unit at the other end of the tunnel receives the hashes and compares them with the hashes in its local byte caching database. If any hashes match, that data does not have to be transmitted over the WAN optimization tunnel. The data for any hashes that does not match is transferred over the tunnel and added to that byte caching database. Then the unit of application data (the file being downloaded) is reassembled and sent to its destination.

The stored byte caches are not application specific. Byte caches from a file in an email can be used to optimize downloading that same file or a similar file from a web page.

The result is less data transmitted over the WAN. Initially, byte caching may reduce performance until a large enough byte caching database is built up.

To enable byte caching, you select **Byte Caching** in a WAN optimization profile.

Byte caching cannot determine whether or not a file is compressed (for example a zip file), and caches compressed and non-compressed versions of the same file separately.

Dynamic data chunking for byte caching

Dynamic data chunking can improve byte caching by improving detection of data chunks that are already cached in changed files or in data embedded in traffic using an unknown protocol. Dynamic data chunking is available for HTTP, CIFS and FTP.

Use the following command to enable dynamic data chunking for HTTP in the default WAN optimization profile.

```
config wanopt profile
  edit default
    config http
      set prefer-chunking dynamic
  end
```

By default dynamic data chunking is disabled and `prefer-chunking` is set to `fix`.

WAN optimization transparent mode

WAN optimization is transparent to users. This means that with WAN optimization in place, clients connect to servers in the same way as they would without WAN optimization. However, servers receiving packets after WAN optimization “see” different source addresses depending on whether or not transparent mode is selected for WAN optimization. If transparent mode is selected, WAN optimization keeps the original source address of the packets, so servers appear to receive traffic directly from clients. Routing on the server network should be configured to route traffic with client source IP addresses from the server-side FortiGate unit to the server and back to the server-side FortiGate unit.



Some protocols, for example CIFS, may not function as expected if transparent mode is **not** selected. In most cases, for CIFS WAN optimization you should select transparent mode and make sure the server network can route traffic as described to support transparent mode.

If transparent mode is not selected, the source address of the packets received by servers is changed to the address of the server-side FortiGate unit interface that sends the packets to the servers. So servers appear to receive packets from the server-side FortiGate unit. Routing on the server network is simpler in this case because client addresses are not involved. All traffic appears to come from the server-side FortiGate unit and not from individual clients.



Do not confuse WAN optimization transparent mode with FortiGate transparent mode. WAN optimization transparent mode is similar to source NAT. FortiGate's transparent mode is a system setting that controls how the FortiGate unit (or a VDOM) processes traffic.

Configuring transparent mode

You can configure transparent mode by selecting **Transparent** in a WAN Optimization profile. The profile is added to an active WAN Optimization policy.

When you configure a passive WAN Optimization policy you can accept the active policy transparent setting or you can override the active policy transparent setting. From the GUI you can do this by setting the **Passive Option** as follows:

- **default** use the transparent setting in the WAN Optimization profile added to the active policy (client-side configuration).
- **transparent** impose transparent mode (override the active policy transparent mode setting). Packets exiting the FortiGate keep their original source addresses.
- **non-transparent** impose non-transparent mode (override the active policy transparent mode setting). Packets exiting the FortiGate have their source address changed to the address of the server-side FortiGate unit interface that sends the packets to the servers.

From the CLI you can use the following command:

```
config firewall policy
    set wanopt-passive-opt {default | transparent | non-transparent}
end
```

Operating modes and VDOMs

To use WAN optimization, the FortiGate units can operate in either NAT or transparent mode. The client-side and server-side FortiGate units do not have to be operating in the same mode.

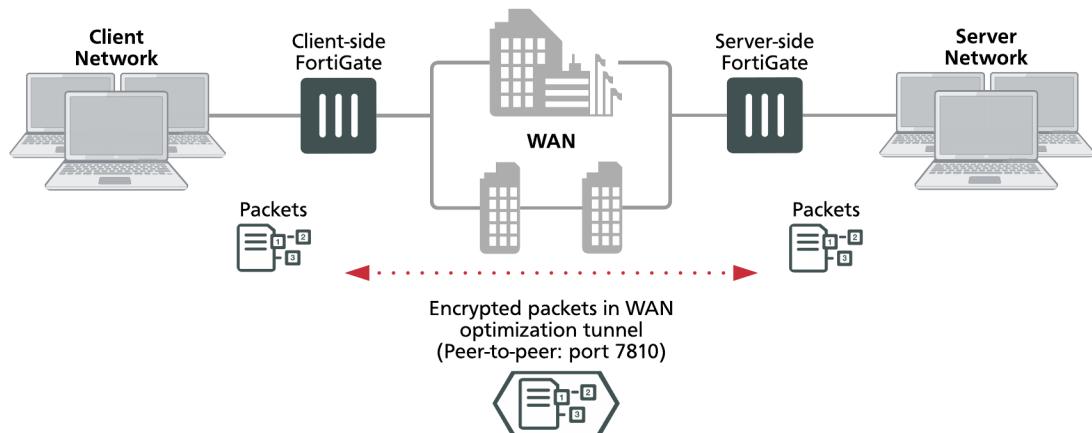
As well, the FortiGate units can be configured for multiple virtual domain (VDOM) operation. You configure WAN optimization for each VDOM and configure one or both of the units to operate with multiple VDOMs enabled.

If a FortiGate unit or VDOM is operating in transparent mode with WAN optimization enabled, WAN optimization uses the management IP address as the peer IP address of the FortiGate unit instead of the address of an interface.

WAN optimization tunnels

All optimized traffic passes between the FortiGate units over a WAN optimization tunnel. Traffic in the tunnel can be sent in plain text or encrypted using AES-128bit-CBC SSL.

WAN optimization tunnels



Both plain text and the encrypted tunnels use TCP destination port 7810.

Before a tunnel can be started, the peers must be configured to authenticate with each other. Then, the client-side peer attempts to start a WAN optimization tunnel with the server-side peer. Once the peers authenticate with each other, they bring up the tunnel and WAN optimization communication over the tunnel starts. After a tunnel has been established, multiple WAN optimization sessions can start and stop between peers without restarting the tunnel.

Tunnel sharing

You can use the `tunnel-sharing` WAN optimization profile CLI keyword to configure tunnel sharing for WAN optimization rules. Tunnel sharing means multiple WAN optimization sessions share the same tunnel. Tunnel sharing can improve performance by reducing the number of WAN optimization tunnels between FortiGate units. Having fewer tunnels means less data to manage. Also, tunnel setup requires more than one exchange of information between the ends of the tunnel. Once the tunnel is set up, each new session that shares the tunnel avoids tunnel setup delays.

Tunnel sharing also uses bandwidth more efficiently by reducing the chances that small packets will be sent down the tunnel. Processing small packets reduces network throughput, so reducing the number of small packets improves performance. A shared tunnel can combine all the data from the sessions being processed by the tunnel and send the data together. For example, suppose a FortiGate unit is processing five WAN optimization sessions and each session has 100 bytes to send. If these sessions use a shared tunnel, WAN optimization combines the packets from all five sessions into one 500-byte packet. If each session uses its own private tunnel, five 100-byte packets will be sent instead. Each packet also requires a TCP ACK reply. The combined packet in the shared tunnel requires one TCP ACK packet. The separate packets in the private tunnels require five.

Use the following command to configure tunnel sharing for HTTP traffic in a WAN optimization profile.

```
config wanopt profile
  edit default
    config http
      set tunnel-sharing {express-shared | private | shared}
  end
```

Tunnel sharing is not always recommended and may not always be the best practice. Aggressive and non-aggressive protocols should not share the same tunnel. An aggressive protocol can be defined as a protocol that is able to get more bandwidth than a non-aggressive protocol. (The aggressive protocols can “starve” the non-

aggressive protocols.) HTTP and FTP are considered aggressive protocols. If aggressive and non-aggressive protocols share the same tunnel, the aggressive protocols may take all of the available bandwidth. As a result, the performance of less aggressive protocols could be reduced. To avoid this problem, rules for HTTP and FTP traffic should have their own tunnel. To do this, set `tunnel-sharing` to `private` for WAN optimization rules that accept HTTP or FTP traffic.

It is also useful to set `tunnel-sharing` to `express-shared` for applications, such as Telnet, that are very interactive but not aggressive. Express sharing optimizes tunnel sharing for Telnet and other interactive applications where latency or delays would seriously affect the user's experience with the protocol.

Set `tunnel-sharing` to `shared` for applications that are not aggressive and are not sensitive to latency or delays. WAN optimization rules set to `sharing` and `express-shared` can share the same tunnel.

WAN optimization and user and device identity policies, load balancing and traffic shaping

Please note the following about WAN optimization and firewall policies:

- WAN optimization is not compatible with firewall load balancing.
- WAN optimization is compatible with source and destination NAT options in firewall policies (including firewall virtual IPs). If a virtual IP is added to a policy the traffic that exits the WAN optimization tunnel has its destination address changed to the virtual IPs mapped to IP address and port.
- WAN optimization is compatible with user identity-based and device identity security policies. If a session is allowed after authentication or device identification the session can be optimized.

Traffic shaping

Traffic shaping works for WAN optimization traffic that is not in a WAN optimization tunnel. So traffic accepted by a WAN optimization security policy on a client-side FortiGate unit can be shaped on ingress. However, when the traffic enters the WAN optimization tunnel, traffic shaping is not applied.

In manual mode:

- Traffic shaping works as expected on the client-side FortiGate unit.
- Traffic shaping cannot be applied to traffic on the server-side FortiGate unit.

In active-passive mode:

- Traffic shaping works as expected on the client-side FortiGate unit.
- If transparent mode is enabled in the WAN optimization profile, traffic shaping also works as expected on the server-side FortiGate unit.
- If transparent mode is not enabled, traffic shaping works partially on the server-side FortiGate unit.

WAN optimization and HA

You can configure WAN optimization on a FortiGate HA cluster. The recommended best practice HA configuration for WAN optimization is active-passive mode. When the cluster is operating, all WAN optimization

sessions are processed by the primary unit only. Even if the cluster is operating in active-active mode, HA does not load-balance WAN optimization sessions.

You can also form a WAN optimization tunnel between a cluster and a standalone FortiGate unit or between two clusters.

In a cluster, only the primary unit stores the byte cache database. This database is not synchronized to the subordinate units. So, after a failover, the new primary unit must rebuild its byte cache. Rebuilding the byte cache can happen relatively quickly because the new primary unit gets byte cache data from the other FortiGate unit that it is participating with in WAN optimization tunnels.

WAN optimization, web caching and memory usage

To accelerate and optimize disk access and to provide better throughput and less latency FortiOS WAN optimization uses provisioned memory to reduce disk I/O and increase disk I/O efficiency. In addition, WAN optimization requires a small amount of additional memory per session for comprehensive flow control logic and efficient traffic forwarding.

When WAN optimization is enabled you will see a reduction in available memory. The reduction increases when more WAN optimization sessions are being processed. If you are thinking of enabling WAN optimization on an operating FortiGate unit, make sure its memory usage is not maxed out during high traffic periods.

In addition to using the system dashboard to see the current memory usage you can use the `get test wad 2` command to see how much memory is currently being used by WAN optimization. See "`get test {wad | wccpd} <test_level>`" for more information.

WAN optimization configuration

This chapter describes FortiGate WAN optimization client server architecture and other concepts you need to understand to be able to configure FortiGate WAN optimization.

Manual (peer-to-peer) and active-passive WAN optimization

You can create **manual** (peer-to-peer) and **active-passive** WAN optimization configurations.



In reality, because WAN optimization traffic can only be processed by one CPU core, it is not recommended to increase the number of manual mode peers on the FortiGate unit per VDOM.

Note that the maximum number of manual peers are restricted to 256 per VDOM. However, in Active-Passive configurations, there is no hard-limit to the maximum number of manual peers per VDOM.

Manual (peer to peer) configurations

Manual configurations allow for WAN optimization between one client-side FortiGate unit and one server-side FortiGate unit. To create a manual configuration you add a **manual mode** WAN optimization security policy to the client-side FortiGate unit. The manual mode policy includes the peer ID of a server-side FortiGate unit.

In a manual mode configuration, the client-side peer can only connect to the named server-side peer. When the client-side peer initiates a tunnel with the server-side peer, the packets that initiate the tunnel include extra information so that the server-side peer can determine that it is a peer-to-peer tunnel request. This extra information is required because the server-side peer does not require a WAN optimization policy; however, you need to add the client peer host ID and IP address to the server-side FortiGate unit peer list.

In addition, from the server-side FortiGate unit CLI you must add an Explicit Proxy security policy with `proxy set` to `wanopt` and the destination interface and network set to the network containing the servers that clients connect to over the WAN optimization tunnel. WAN optimization tunnel requests are accepted by the explicit proxy policy and if the client-side peer is in the server side peer's address list the traffic is forwarded to the servers on the destination network.

Manual mode client-side policy

You must configure manual mode client-side policies from the CLI. From the GUI a manual mode policy has WAN Optimization turned on and includes the following text beside the *WAN optimization* field: *Manual (Profile: <profile-name>. Peer: <peer-name>)*.

Add a manual mode policy to the client-side FortiGate unit from the CLI. The policy enables WAN optimization, sets `wanopt-detection` to `off`, and uses the `wanopt-peer` option to specify the server-side peer. The following example uses the default WAN optimization profile.

```
config firewall policy
  edit 2
    set srcintf internal
```

```

set dstintf wan1
set srcaddr client-subnet
set dstaddr server-subnet
set action accept
set schedule always
set service ALL
set wanopt enable
set wanopt-detection off
set wanopt-profile default
set wanopt-peer server
next
end

```

Manual mode server-side explicit proxy policy

The server-side explicit proxy policy allows connections from the WAN optimization tunnel to the server network by setting the proxy type to `wanopt`. You must add policies that `set proxy` to `wanopt` from the CLI and these policies do not appear on the GUI. The policy should look like the following:

```

configure firewall proxy-policy
edit 3
set proxy wanopt
set dstintf internal
set srcaddr all
set dstaddr server-subnet
set action accept
set schedule always
set service ALL
next
end

```

Active-passive configurations

Active-passive WAN optimization requires an **active** WAN optimization policy on the client-side FortiGate unit and a **passive** WAN optimization policy on the server-side FortiGate unit. The server-side FortiGate unit also requires an explicit proxy policy with `proxy` set to `wanopt`.

You can use the passive policy to control WAN optimization address translation by specifying **transparent mode** or non-transparent mode. See [Manual \(peer-to-peer\) and active-passive WAN optimization on page 284](#). You can also use the passive policy to apply security profiles, web caching, and other FortiGate features at the server-side FortiGate unit. For example, if a server-side FortiGate unit is protecting a web server, the passive policy could enable web caching.

A single passive policy can accept tunnel requests from multiple FortiGate units as long as the server-side FortiGate unit includes their peer IDs and all of the client-side FortiGate units include the server-side peer ID.

Active client-side policy

Add an active policy to the client-side FortiGate unit by turning on **WAN Optimization** and selecting **active**. Then select a **WAN optimization Profile**. From the CLI the policy could look like the following:

```

config firewall policy
edit 2
set srcintf internal
set dstintf wan1
set srcaddr client-subnet
set dstaddr server-subnet

```

```
set action accept
set schedule always
set service ALL
set wanopt enable
set wanopt-detection active
set wanopt-profile default
next
end
```

Server-side tunnel policy

The server-side requires an explicit proxy policy that sets the `proxy` to `wanopt`. You must add this policy from the CLI and policies with `proxy` set to `wanopt` do not appear on the GUI. From the CLI the policy could look like the following:

```
configure firewall proxy-policy
edit 3
  set proxy wanopt
  set dstintf internal
  set srcaddr all
  set dstaddr server-subnet
  set action accept
  set schedule always
  set service ALL
next
end
```

Server-side passive policy

Add a passive policy to the server-side FortiGate unit by selecting **Enable WAN Optimization** and selecting **passive**. Then set the **Passive Option** to **transparent**. From the CLI the policy could look like the following:

```
config firewall policy
edit 2
  set srcintf "wan1"
  set dstintf "internal"
  set srcaddr "all"
  set dstaddr "all"
  set action accept
  set schedule "always"
  set service "ALL"
  set wanopt enable
  set wanopt-detection passive
  set wanopt-passive-opt transparent
next
```

WAN optimization profiles

Use WAN optimization profiles to apply WAN optimization techniques to traffic to be optimized. In a WAN optimization profile you can select the protocols to be optimized and for each protocol you can enable SSL offloading (if supported), secure tunneling, byte caching and set the port or port range the protocol uses. You can also enable transparent mode and optionally select an authentication group. You can edit the default WAN optimization profile or create new ones.

To configure a WAN optimization profile go to **WAN Opt. & Cache > Profiles** and edit a profile or create a new one.

Configuring a WAN optimization profile

Edit WAN Optimization Profile default + ≡

Name	default																														
Comments	default WANopt profile 22/255																														
<input checked="" type="checkbox"/> Transparent Mode																															
<input checked="" type="checkbox"/> Authentication Group	Auth-Grp																														
<table border="1"> <thead> <tr> <th>Protocol</th> <th>SSL Offloading</th> <th>Secure Tunneling</th> <th>Byte Caching</th> <th>Port</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> CIFS</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>445</td> </tr> <tr> <td><input checked="" type="checkbox"/> FTP</td> <td></td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>21</td> </tr> <tr> <td><input checked="" type="checkbox"/> HTTP</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>80</td> </tr> <tr> <td><input checked="" type="checkbox"/> MAPI</td> <td></td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>135</td> </tr> <tr> <td><input checked="" type="checkbox"/> TCP</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>1-65535</td> </tr> </tbody> </table>		Protocol	SSL Offloading	Secure Tunneling	Byte Caching	Port	<input checked="" type="checkbox"/> CIFS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	445	<input checked="" type="checkbox"/> FTP		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	21	<input checked="" type="checkbox"/> HTTP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	80	<input checked="" type="checkbox"/> MAPI		<input type="checkbox"/>	<input checked="" type="checkbox"/>	135	<input checked="" type="checkbox"/> TCP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1-65535
Protocol	SSL Offloading	Secure Tunneling	Byte Caching	Port																											
<input checked="" type="checkbox"/> CIFS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	445																											
<input checked="" type="checkbox"/> FTP		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	21																											
<input checked="" type="checkbox"/> HTTP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	80																											
<input checked="" type="checkbox"/> MAPI		<input type="checkbox"/>	<input checked="" type="checkbox"/>	135																											
<input checked="" type="checkbox"/> TCP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1-65535																											

From the CLI you can use the following command to configure a WAN optimization profile to optimize HTTP traffic.

```
config wanopt profile
  edit new-profile
    config http
      set status enable
  end
```

Transparent Mode

Servers receiving packets after WAN optimization “see” different source addresses depending on whether or not you select **Transparent Mode**.

For more information, see [WAN optimization profiles on page 286](#).

Select this option and select an authentication group so that the client and server-side FortiGate units must authenticate with each other before starting the WAN optimization tunnel. You must also select an authentication group if you select **Secure Tunneling** for any protocol.

Authentication Group

You must add identical authentication groups to both of the FortiGate units that will participate in the WAN optimization tunnel. For more information, see [Configuring authentication groups on page 1](#).

Protocol

Select CIFS, FTP, HTTP or MAPI to apply protocol optimization for the selected protocols. See [WAN optimization profiles on page 286](#).

Select TCP if the WAN optimization tunnel accepts sessions that use more than one protocol or that do not use the CIFS, FTP, HTTP, or MAPI protocol.

SSL Offloading	Select to apply SSL offloading for HTTPS or other SSL traffic. You can use SSL offloading to offload SSL encryption and decryption from one or more HTTP servers to the FortiGate unit. If you enable this option, you must configure the security policy to accept SSL-encrypted traffic.
Secure Tunneling	If you enable SSL offloading, you must also use the CLI command <code>config firewall ssl-server</code> to add an SSL server for each HTTP server that you want to offload SSL encryption/decryption for. For more information, see Turning on web caching for HTTPS traffic on page 1 .
Byte Caching	Select to apply WAN optimization byte caching to the sessions accepted by this rule. For more information, see "Byte caching" .
Port	Enter a single port number or port number range. Only packets whose destination port number matches this port number or port number range will be optimized.

Processing non-HTTP sessions accepted by a WAN optimization profile with HTTP optimization

From the CLI, you can use the following command to configure how to process non-HTTP sessions when a rule configured to accept and optimize HTTP traffic accepts a non-HTTP session. This can occur if an application sends non-HTTP sessions using an HTTP destination port.

```
config wanopt profile
  edit default
    config http
      set status enable
      set tunnel-non-http {disable | enable}
    end
```

To drop non-HTTP sessions accepted by the rule set `tunnel-non-http` to `disable`, or set it to `enable` to pass non-HTTP sessions through the tunnel without applying protocol optimization, byte-caching, or web caching. In this case, the FortiGate unit applies TCP protocol optimization to non-HTTP sessions.

Processing unknown HTTP sessions

Unknown HTTP sessions are HTTP sessions that do not comply with HTTP 0.9, 1.0, or 1.1. From the CLI, use the following command to specify how a rule handles such HTTP sessions.

```
config wanopt profile
  edit default
    config http
      set status enable
      set unknown-http-version {best-effort | reject | tunnel}
    end
```

To assume that all HTTP sessions accepted by the rule comply with HTTP 0.9, 1.0, or 1.1, select **best-effort**. If a session uses a different HTTP version, WAN optimization may not parse it correctly. As a result, the FortiGate unit may stop forwarding the session and the connection may be lost. To reject HTTP sessions that do not use HTTP 0.9, 1.0, or 1.1, select **reject**.

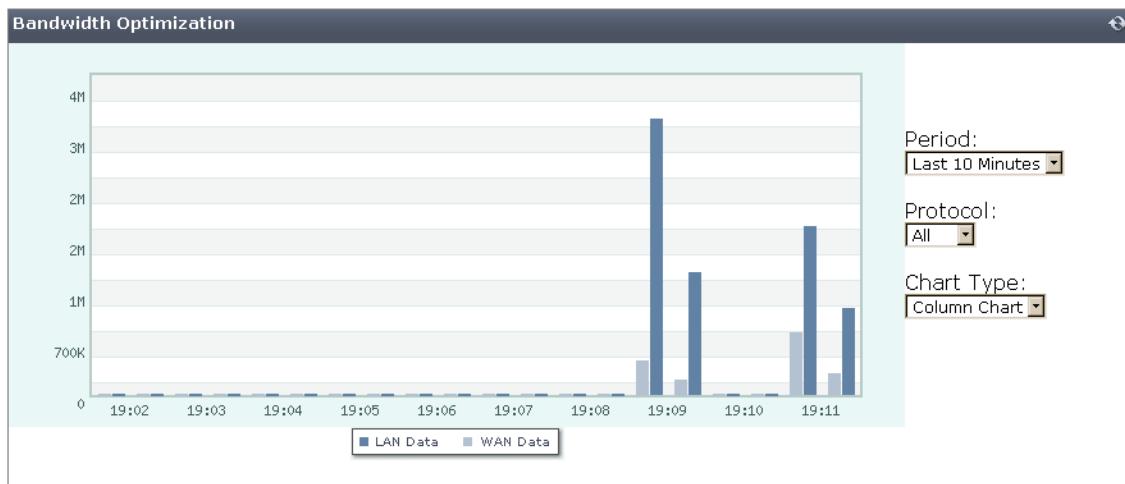
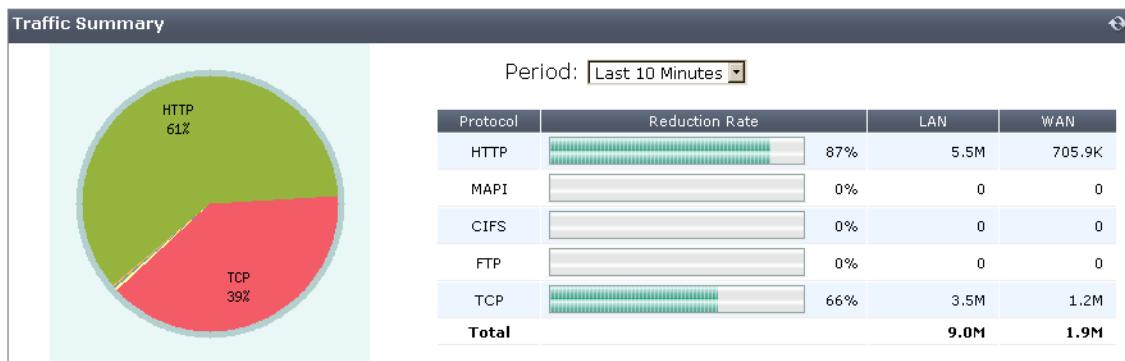
To pass HTTP sessions that do not use HTTP 0.9, 1.0, or 1.1, but without applying HTTP protocol optimization, byte-caching, or web caching, you can also select **tunnel**. TCP protocol optimization is applied to these HTTP sessions.

Monitoring WAN optimization performance

Using WAN optimization monitoring, you can confirm that a FortiGate unit is optimizing traffic and view estimates of the amount of bandwidth saved. The WAN optimization monitor presents collected log information in a graphical format to show network traffic summary and bandwidth optimization information.

To view the WAN optimization monitor, go to **Monitor > WAN Opt. Monitor**.

WAN optimization monitor



Traffic summary

The traffic summary shows how WAN optimization is reducing the amount of traffic on the WAN for each WAN optimization protocol by showing the traffic reduction rate as a percentage of the total traffic. The traffic summary also shows the amount of WAN and LAN traffic. If WAN optimization is being effective the amount of WAN traffic should be lower than the amount of LAN traffic.

You can use the refresh icon to update the traffic summary display at any time. You can also set the amount of time for which the traffic summary shows data. The time period can vary from the last 10 minutes to the last month.

Bandwidth optimization

This section shows network bandwidth optimization per time period. A line or column chart compares an application's pre-optimized size (LAN data) with its optimized size (WAN data). You can select the chart type, the monitoring time period, and the protocol for which to display data. If WAN optimization is being effective the WAN bandwidth should be lower than the LAN bandwidth.

WAN optimization configuration summary

This section includes a client-side and a server-side WAN Optimization configuration summary.:

Client-side configuration summary

WAN optimization profile

Enter the following command to view WAN optimization profile CLI options:

```
tree wanopt profile
-- [profile] --*name (36)
  |- transparent
  |- comments
  |- auth-group (36)
  |- <http> -- status
    |- secure-tunnel
    |- byte-caching
    |- prefer-chunking
    |- tunnel-sharing
    |- log-traffic
    |- port (1,65535)
    |- ssl
    |- ssl-port (1,65535)
    |- unknown-http-version
    +- tunnel-non-http
  |- <cifs> -- status
    |- secure-tunnel
    |- byte-caching
    |- prefer-chunking
    |- tunnel-sharing
    |- log-traffic
    +- port (1,65535)
```

```

|- <mapi> -- status
  |- secure-tunnel
  |- byte-caching
  |- tunnel-sharing
  |- log-traffic
  +- port (1,65535)
|- <ftp> -- status
  |- secure-tunnel
  |- byte-caching
  |- prefer-chunking
  |- tunnel-sharing
  |- log-traffic
  +- port (1,65535)
+- <tcp> -- status
  |- secure-tunnel
  |- byte-caching
  |- byte-caching-opt
  |- tunnel-sharing
  |- log-traffic
  |- port
  |- ssl
  +- ssl-port (1,65535)

```

Local host ID and peer settings

```

config wanopt settings
  set host-id client
end
config wanopt peer
  edit server
  set ip 10.10.2.82
end

```

Security policies

Two client-side WAN optimization security policy configurations are possible. One for active-passive WAN optimization and one for manual WAN optimization.

Active/passive mode on the client-side

```

config firewall policy
  edit 2
    set srcintf internal
    set dstintf wan1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set wanopt enable <<< enable WAN optimization
    set wanopt-detection active <<< set the mode to active/passive
    set wanopt-profile "default" <<< select the wanopt profile
  next
end

```

Manual mode on the client-side

```

config firewall policy
  edit 2
    set srcintf internal
    set dstintf wan1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set wanopt enable <<< enable WAN optimization
    set wanopt-detection off <<< sets the mode to manual
    set wanopt-profile "default" <<< select the wanopt profile
    set wanopt-peer "server" <<< set the only peer to do wanopt
                                         with
                                         (required for manual mode)
  next
end

```

server-side configuration summary

Local host ID and peer settings

```

config wanopt settings
  set host-id server
end
config wanopt peer
  edit client
    set ip 10.10.2.81
end

```

Security policies

Two server-side WAN optimization security policy configurations are possible. One for active-passive WAN optimization and one for manual WAN optimization.

Active/passive mode on server-side

```

config firewall policy
  edit 2 <<< the passive mode policy
    set srcintf wan1
    set dstintf internal
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set wanopt enable
    set wanopt-detection passive
    set wanopt-passive-opt transparent
  end
config firewall proxy-policy
  edit 3 <<< policy that accepts wanopt tunnel connections from the server
    set proxy wanopt <<< wanopt proxy type
    set dstintf internal

```

```

    set srcaddr all
    set dstaddr server-subnet
    set action accept
    set schedule always
    set service ALL
next
end

```

Manual mode on server-side

```

config firewall proxy-policy
    edit 3 <<< policy that accepts wanopt tunnel connections from the           client
        set proxy wanopt <<< wanopt proxy type
        set dstintf internal
        set srcaddr all
        set dstaddr server-subnet
        set action accept
        set schedule always
        set service ALL
    next
end

```

WANopt storage

The `config wanopt storage` option has been combined with `config system storage`.

Setting the disk-usage mode is no longer in `config system global`. It is set through `config system storage`.

Syntax:

```

config system storage
    edit <name-string>
        set status enable
        set media-status
        set order
        set partition
        set device
        set size
        set usage
        set wanopt-mode

```

Option	Description
status	Enable/disable storage
media-status	Enable/disable the physical status of current media
order	Set storage order

Option	Description
partition	Label of underlying partition Example: "MIXEDXXXE2946380"
device	Partition device. Example: "/dev/vdb1"
size	Partition size. Example: 8616
usage	Use hard disk for logging and WAN Optimization.
wanopt-mode	WAN Optimization mode <ul style="list-style-type: none"> mix - default, recommended wanopt - recommended if only wanopt feature is enabled webcache - recommended if only webcache feature is enabled If only one of the two features is being used, using the applicable recommended mode will give a higher cache capacity and improve performance.

WANopt cache service

The `config wanopt cache-service` command is used to configure `cache-service` clusters between multiple FortiGates. The result is that the `cache-service` daemons of the different FortiGates can collaborate together for serving web cache entries.

To configure the wanopt cache-service

```

config wanopt cache-service
  set prefer-scenario
  set collaboration
  set device-id
  set acceptable-connections
  config dst-peer
    edit <dst-peer-name>
      set auth-type
      set encode-type
      set priority
      set ip
  config src-peer
    edit <src-peer>
      set auth-type
      set encode-type
      set priority
      set ip

```

Option	Description
prefer-scenario	Set the preferred cache behavior to the appropriate balance between latency and hit ratio Options: <ul style="list-style-type: none">• balance - Balance between speed and cache hit ratio.• prefer-speed - Prefer response speed at the expense of increased cache bypasses.• prefer-cache - Prefer improving hit-ratio through increasing latency tolerance.
collaboration	enable/disable cache collaboration between cache-service clusters
device-id	Set identifier for this cache device
acceptable-connections	Set strategy when accepting cache collaboration connection Options: <ul style="list-style-type: none">• any - The cache-service can accept any cache collaboration connection.• peers - The cache-service will only accept connections that are already in src-peers.
auth-type	Set authentication type for this peer Value is integer from 0 to 255
encode-type	Set encode type for this peer Value is integer from 0 to 255
priority	Set priority for this peer Value is integer from 0 to 255. Default = 1
ip	Set cluster IP address of this peer

Video caching

This `config wanopt content-delivery-network-rule` command configures web-caching including the video-cache matching rules.

To configure the `wanopt content-delivery-network-rule`

```
config wanopt content-delivery-network-rule
  edit <content_rule_name>
    set comment
    set status
```

```

set host-domain-name-suffix
set category
set request-cache-control
set response-cache-control
set response-expires
set text-response-vcache
set updateserver
config rules
  edit <rule_name>
    set match-mode
    set skip-rule-mode
    config match-entries
      edit <integer>
        set target
        set pattern
    config skip-entries
      set target
      set pattern
    config content id
      set target
      set start-str
      set start-skip
      set start-direction
      set end-str
      set end-skip
      set end-direction
      set range-str

```

Option	Description
comment	Comment about this rule
status	Enable/disable WAN optimization content delivery network rules
host-domain-name-suffix	Suffix portion of the fully qualified domain name (eg. fortinet.com in "www.fortinet.com")
category	Content delivery network rule category
request-cache-control	Enable/disable HTTP request cache control
response-cache-control	Enable/disable HTTP response cache control
response-expires	Enable/disable HTTP response cache expires
updateserver	Enable/disable update server
match-mode	Match criteria for collecting content ID
skip-rule-mode	Skip mode when evaluating skip rules

Option	Description
target	Option in HTTP header or URL parameter to match
pattern	Pattern string for matching target (Referrer or URL pattern, eg. "a", "a*c", "*a*", "a*c*e", and "")
start-str	String from which to start search
start-skip	Number of characters in URL to skip after start-str has been matched
start-direction	Search direction from start-str match
end-str	String from which to end search
end-skip	Number of characters in URL to skip after end-str has been matched
end-direction	Search direction from end-str match
range-str	Name of content ID within the start string and end string

Best practices

This is a short list of WAN optimization and explicit proxy best practices.

- WAN optimization tunnel sharing is recommended for similar types of WAN optimization traffic. However, tunnel sharing for different types of traffic is not recommended. For example, aggressive and non-aggressive protocols should not share the same tunnel. See [Best practices on page 297](#).
- Active-passive HA is the recommended HA configuration for WAN optimization. See [Best practices on page 297](#).
- Configure WAN optimization authentication with specific peers. Accepting any peer is not recommended as this can be less secure. See [Accepting any peers on page 1](#).
- Set the explicit proxy **Default Firewall Policy Action to Deny**. This means that a security policy is required to use the explicit web proxy. See [General explicit web proxy configuration steps on page 1](#).
- Set the explicit FTP proxy **Default Firewall Policy Action to Deny**. This means that a security policy is required to use the explicit FTP proxy. See [General explicit FTP proxy configuration steps on page 1](#).
- Do not enable the explicit web or FTP proxy on an interface connected to the Internet. This is a security risk because anyone on the Internet who finds the proxy could use it to hide their source address. If you must enable the proxy on such an interface make sure authentication is required to use the proxy. See [General explicit web proxy configuration steps on page 1](#).

Example basic manual (peer-to-peer) WAN optimization configuration

In a manual (peer to peer) configuration the WAN optimization tunnel can be set up between one client-side FortiGate unit and one server-side FortiGate unit. The peer ID of the server-side FortiGate unit is added to the client-side WAN optimization policy. When the client-side FortiGate unit initiates a tunnel with the server-side FortiGate unit, the packets that initiate the tunnel include information that allows the server-side FortiGate unit to

determine that it is a manual tunnel request. The server-side FortiGate unit does not require a WAN optimization profile; you just need to add the client peer host ID and IP address to the server-side FortiGate unit peer list and from the CLI an explicit proxy policy to accept WAN optimization tunnel connections.

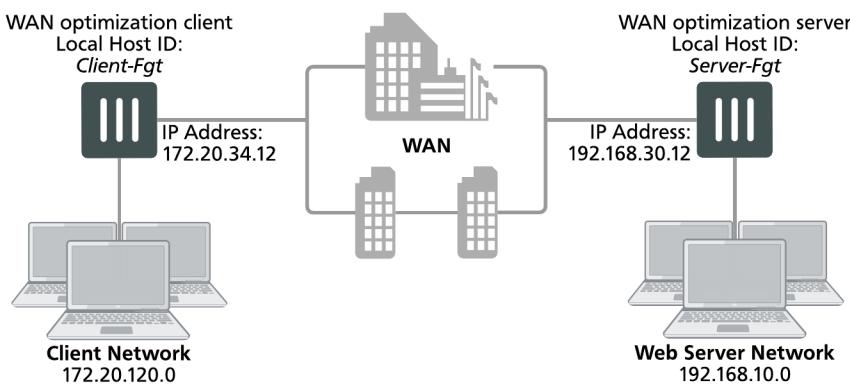
In a manual WAN optimization configuration, you create a manual WAN optimization security policy on the client-side FortiGate unit. To do this you must use the CLI to set `wanopt-detection to off` and to add the peer host ID of the server-side FortiGate unit to the WAN optimization security policy.

Network topology and assumptions

This example configuration includes a client-side FortiGate unit called Client-Fgt with a WAN IP address of 172.20.34.12. This unit is in front of a network with IP address 172.20.120.0. The server-side FortiGate unit is called Server_Fgt with a WAN IP address of 192.168.30.12. This unit is in front of a web server network with IP address 192.168.10.0.

This example customizes the default WAN optimization profile on the client-side FortiGate unit and adds it to the WAN optimization policy. You can also create a new WAN optimization profile.

Example manual (peer-to-peer) topology



General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Configure the client-side FortiGate unit:
 - Add peers.
 - Configure the default WAN optimization profile to optimize HTTP traffic.
 - Add a manual WAN optimization security policy.
2. Configure the server-side FortiGate unit:
 - Add peers.
 - Add a WAN optimization tunnel policy.

Configuring basic peer-to-peer WAN optimization - web-based manager

Use the following steps to configure the example configuration from the web-based manager.

To configure the client-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the client-side FortiGate unit:

Local Host ID	Client-Fgt
----------------------	------------

2. Select **Apply**.
3. Select **Create New** and add the server-side FortiGate unit **Peer Host ID** and **IP Address** for the server-side FortiGate:

Peer Host ID	Server-Fgt
IP Address	192.168.30.12

4. Select **OK**.
5. Go to **Policy & Objects > Addresses** and select **Create New** to add a firewall address for the client network.

Category	Address
Name	Client-Net
Type	Subnet
Subnet / IP Range	172.20.120.0/24
Interface	port1

6. Select **Create New** to add a firewall address for the web server network.

Category	Address
Name	Web-Server-Net
Type	Subnet
Subnet / IP Range	192.168.10.0/24
Interface	port2

7. Go to **WAN Opt. & Cache > Profiles** and edit the default profile.
8. Select **Transparent Mode**.
9. Under Protocol, select **HTTP** and for HTTP select **Byte Caching**. Leave the **HTTP Port** set to 80.
10. Select **Apply** to save your changes.
11. Go to **Policy & Objects > IPv4 Policy** and add a WAN optimization security policy to the client-side FortiGate unit that accepts traffic to be optimized:

Incoming Interface	port1
Source Address	all

Outgoing Interface	port2
Destination Address	all
Schedule	always
Service	ALL
Action	ACCEPT

12. Select **Enable WAN Optimization** and configure the following settings:

Enable WAN Optimization	active
Profile	default

13. Select **OK**.

14. Edit the policy from the CLI to turn off `wanopt-detection`, add the peer ID of the server-side FortiGate unit, and the default WAN optimization profile. The following example assumes the ID of the policy is 5:

```
config firewall policy
  edit 5
    set wanopt-detection off
    set wanopt-peer Server-Fgt
    set wanopt-profile default
  end
```

When you set the detection mode to `off` the policy becomes a manual mode WAN optimization policy. On the web-based manager the WAN optimization part of the policy changes to the following:

Enable WAN Optimization	Manual (Profile: default, Peer: Peer-Fgt-2)
--------------------------------	---

To configure the server-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the server-side FortiGate unit:

Local Host ID	Server-Fgt
----------------------	------------

2. Select **Apply**.

3. Select **Create New** and add a **Peer Host ID** and the **IP Address** for the client-side FortiGate unit:

Peer Host ID	Client-Fgt
IP Address	172.20.34.12

4. Select **OK**.

5. Enter the following CLI command to add an explicit proxy policy to accept WAN optimization tunnel connections.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
```

```
    set action accept
    set schedule always
    set service ALL
next
end
```

Configuring basic peer-to-peer WAN optimization - CLI

Use the following steps to configure the example WAN optimization configuration from the client-side and server-side FortiGate unit CLI.

To configure the client-side FortiGate unit

1. Add the Local Host ID to the client-side FortiGate configuration:

```
config wanopt settings
    set host-id Client-Fgt
end
```

2. Add the server-side Local Host ID to the client-side peer list:

```
config wanopt peer
    edit Server-Fgt
        set ip 192.168.30.12
end
```

3. Add a firewall address for the client network.

```
config firewall address
    edit Client-Net
        set type ipmask
        set subnet 172.20.120.0 255.255.255.0
        set associated-interface port1
end
```

4. Add a firewall address for the web server network.

```
config firewall address
    edit Web-Server-Net
        set type ipmask
        set subnet 192.168.10.0 255.255.255.0
        set associated-interface port2
end
```

5. Edit the default WAN optimization profile, select transparent mode, enable HTTP WAN optimization and enable byte caching for HTTP. Leave the HTTP Port set to 80.

```
config wanopt profile
    edit default
        set transparent enable
        config http
            set status enable
            set byte-caching enable
        end
    end
```

6. Add a WAN optimization security policy to the client-side FortiGate unit to accept the traffic to be optimized:

```
config firewall policy
    edit 0
```

```
set srcintf port1
set dstintf port2
set srcaddr all
set dstaddr all
set action accept
set service ALL
set schedule always
set wanopt enable
set wanopt-profile default
set wanopt-detection off
set wanopt-peer Server-Fgt
end
```

To configure the server-side FortiGate unit

1. Add the Local Host ID to the server-side FortiGate configuration:

```
config wanopt settings
  set host-id Server-Fgt
end
```

2. Add the client-side Local Host ID to the server-side peer list:

```
config wanopt peer
  edit Client-Fgt
    set ip 192.168.30.12
end
```

3. Add a WAN optimization tunnel explicit proxy policy.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  next
end
```

Testing and troubleshooting the configuration

To test the configuration attempt to start a web browsing session between the client network and the web server network. For example, from a PC on the client network browse to the IP address of a web server on the web server network, for example <http://192.168.10.100>. Even though this address is not on the client network you should be able to connect to this web server over the WAN optimization tunnel.

If you can connect, check WAN optimization monitoring. If WAN optimization has been forwarding the traffic the WAN optimization monitor should show the protocol that has been optimized (in this case HTTP) and the reduction rate in WAN bandwidth usage.

If you can't connect you can try the following to diagnose the problem:

- Review your configuration and make sure all details such as address ranges, peer names, and IP addresses are correct.

- Confirm that the security policy on the client-side FortiGate unit is accepting traffic for the 192.168.10.0 network. You can do this by checking the policy monitor (**Monitor > Firewall User Monitor**). Look for sessions that use the policy ID of this policy.
- Check routing on the FortiGate units and on the client and web server networks to make sure packets can be forwarded as required. The FortiGate units must be able to communicate with each other, routing on the client network must allow packets destined for the web server network to be received by the client-side FortiGate unit, and packets from the server-side FortiGate unit must be able to reach the web servers.

You can use the following `get` and `diagnose` commands to display information about how WAN optimization is operating.

Enter the following command to list all of the running WAN optimization tunnels and display information about each one. The command output for the client-side FortiGate unit shows 10 tunnels all created by peer-to-peer WAN optimization rules (auto-detect set to off).

```
diagnose wad tunnel list

Tunnel: id=100 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=100 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=348 bytes_out=384

Tunnel: id=99 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=99 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=348 bytes_out=384

Tunnel: id=98 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=98 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=348 bytes_out=384

Tunnel: id=39 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=39 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=1068 bytes_out=1104

Tunnel: id=7 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=7 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=1228 bytes_out=1264

Tunnel: id=8 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=8 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
    bytes_in=1228 bytes_out=1264

Tunnel: id=5 type=manual
    vd=0 shared=no uses=0 state=3
    peer name=Web-servers id=5 ip=192.168.30.12
    SSL-secured-tunnel=no auth-grp=
```

```
bytes_in=1228 bytes_out=1264

Tunnel: id=4 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web-servers id=4 ip=192.168.30.12
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=1 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web-servers id=1 ip=192.168.30.12
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=2 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web-servers id=2 ip=192.168.30.12
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnels total=10 manual=10 auto=0
```

Example active-passive WAN optimization

In active-passive WAN optimization you add an active WAN optimization policy to the client-side FortiGate unit and you add a WAN optimization tunnel policy and a passive WAN optimization policy to the server-side FortiGate unit.

The active policy accepts the traffic to be optimized and sends it down the WAN optimization tunnel to the server-side FortiGate unit. The active policy can also apply security profiles and other features to traffic before it exits the client-side FortiGate unit.

A tunnel explicit proxy policy on the sever-side FortiGate unit allows the server-side FortiGate unit to form a WAN optimization tunnel with the client-side FortiGate unit. The passive WAN optimization policy is required because of the active policy on the client-side FortiGate unit. You can also use the passive policy to apply WAN optimization transparent mode and features such as security profiles, logging, traffic shaping and web caching to the traffic before it exits the server-side FortiGate unit.

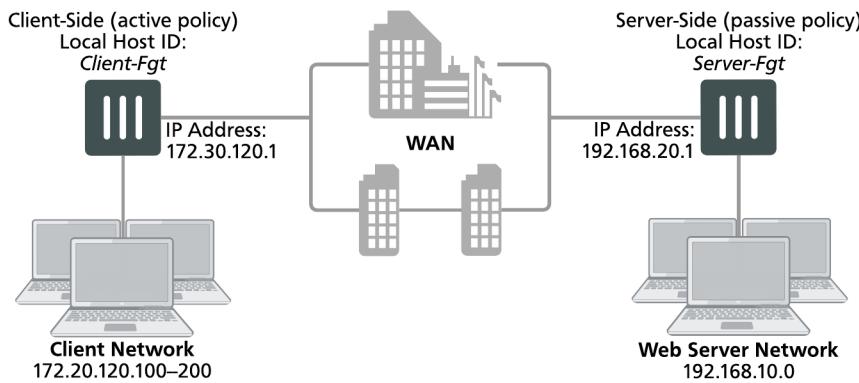
Network topology and assumptions

On the client-side FortiGate unit this example configuration includes a WAN optimization profile that optimizes CIFS, HTTP, and FTP traffic and an active WAN optimization policy. The active policy also applies virus scanning to the WAN optimization traffic.

On the server-side FortiGate unit, the passive policy applies application control to the WAN optimization traffic.

In this example, WAN optimization transparent mode is selected in the WAN optimization profile and the passive WAN optimization policy accepts this transparent mode setting. This means that the optimized packets maintain their original source and destination addresses. As a result, routing on the client network must be configured to route packets for the server network to the client-side FortiGate unit. Also the routing configuration on the server network must be able to route packets for the client network to the server-side FortiGate unit.

Example active-passive WAN optimization topology



General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Configure the client-side FortiGate unit:
 - Add peers.
 - Add a WAN optimization profile to optimize CIFS, FTP, and HTTP traffic.
 - Add firewall addresses for the client and web server networks.
 - Add an active WAN optimization policy.
2. Configure the server-side FortiGate unit by:
 - Add peers.
 - Add firewall addresses for the client and web server networks.
 - Add a passive WAN optimization policy.
 - Add a WAN optimization tunnel policy.

Configuring basic active-passive WAN optimization - web-based manager

Use the following steps to configure the example WAN optimization configuration from the client-side and server-side FortiGate unit web-based manager.

To configure the client-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the client-side FortiGate unit:

Local Host ID	Client-Fgt
----------------------	------------

2. Select **Apply**.

3. Select **Create New** and add a Peer Host ID and the IP Address for the server-side FortiGate unit:

Peer Host ID	Server-Fgt
IP Address	192.168.20.1

4. Select **OK**.
5. Go to **WAN Opt. & Cache > Profiles** and select **Create New** to add a WAN optimization profile to optimize CIFS, HTTP, and FTP traffic:

Name	Custom-wan-opt-pro
Transparent Mode	Select

6. Select the **CIFS** protocol, select **Byte Caching** and set the **Port** to 445.
7. Select the **FTP** protocol, select **Byte Caching** and set the **Port** to 21.
8. Select the **HTTP** protocol, select **Byte Caching** and set the **Port** to 80.
9. Select **OK**.

10. Go to **Policy & Objects > Addresses** and select **Create New** to add an address for the client network.

Category	Address
Address Name	Client-Net
Type	IP Range
Subnet / IP Range	172.20.120.100-172.20.120.200
Interface	port1

11. Select **Create New** to add an address for the web server network.

Category	Address
Address Name	Web-Server-Net
Type	Subnet
Subnet / IP Range	192.168.10.0/24
Interface	port2

12. Go to **Policy & Objects > IPv4 Policy** and select **Create New** to add an active WAN optimization security policy:

Incoming Interface	port1
Source Address	Client-Net
Outgoing Interface	port2
Destination Address	Web-Server-Net
Schedule	always
Service	HTTP FTP SMB

Action	ACCEPT
--------	--------

13. Turn on **WAN Optimization** and configure the following settings:

WAN Optimization	active
Profile	Custom-wan-opt-pro

14. Turn on Antivirus and select the **default** antivirus profile.

15. Select **OK**.

To configure the server-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the server-side FortiGate unit:

Local Host ID	Server-Fgt
----------------------	------------

2. Select **Apply**.

3. Select **Create New** and add a **Peer Host ID** and the **IP Address** for the client-side FortiGate unit:

Peer Host ID	Client-Fgt
IP Address	172.30.120.1

4. Select **OK**.

5. Go to **Policy & Objects > Addresses** and select **Create New** to add an address for the client network.

Category	Address
Address Name	Client-Net
Type	IP Range
Subnet / IP Range	172.20.120.100-172.20.120.200
Interface	port1

6. Select **Create New** to add a firewall address for the web server network.

Category	Address
Address Name	Web-Server-Net
Type	Subnet
Subnet / IP Range	192.168.10.0/24
Interface	port2

7. Select **OK**.
8. Select **Policy & Objects > IPv4 Policy** and select **Create New** to add a passive WAN optimization policy that applies application control.

Incoming Interface	port2
Source Address	Client-Net
Outgoing Interface	port1
Destination Address	Web-Server-Net
Schedule	always
Service	ALL
Action	ACCEPT

9. Turn on **WAN Optimization** and configure the following settings:

WAN Optimization	passive
Passive Option	default

10. Select **OK**.

11. From the CLI enter the following command to add a WAN optimization tunnel explicit proxy policy.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  next
end
```

Configuring basic active-passive WAN optimization - CLI

Use the following steps to configure the example WAN optimization configuration from the client-side and server-side FortiGate unit CLI.

To configure the client-side FortiGate unit

1. Add the Local Host ID to the client-side FortiGate configuration:

```
config wanopt settings
  set host-id Client-Fgt
end
```

2. Add the server-side Local Host ID to the client-side peer list:

```
config wanopt peer
  edit Server-Fgt
  set ip 192.168.20.1
```

```

    end
3. Add a WAN optimization profile to optimize CIFS, HTTP, and FTP traffic.

    config wanopt profile
        edit Custom-wan-opt-pro
            config cifs
                set status enable
                set byte-caching enable
                set port 445
            end
            config http
                set status enable
                set byte-caching enable
                set port 80
            end
            config ftp
                set status enable
                set byte-caching enable
                set port 21
            end
        end
    end

4. Add a firewall address for the client network.

    config firewall address
        edit Client-Net
            set type iprange
            set start-ip 172.20.120.100
            set end-ip 172.20.120.200
            set associated-interface port1
        end

5. Add a firewall address for the web server network.

    config firewall address
        edit Web-Server-Net
            set type ipmask
            set subnet 192.168.10.0 255.255.255.0
            set associated-interface port2
        end

6. Add an active WAN optimization security policy that applies virus scanning:

    config firewall policy
        edit 0
            set srcintf port1
            set dstintf port2
            set srcaddr Client-Net
            set dstaddr Web-Server-Net
            set action accept
            set service HTTP FTP SMB
            set schedule always
            set wanopt enable
            set wanopt-detection active
            set wanopt-profile Custom-wan-opt-pro
        end

```

To configure the server-side FortiGate unit

1. Add the Local Host ID to the server-side FortiGate configuration:

```

    config wanopt settings
        set host-id Server-Fgt
    end

```

2. Add the client-side Local Host ID to the server-side peer list:

```
config wanopt peer
  edit Client-Fgt
    set ip 172.20.120.1
  end
```

3. Add a firewall address for the client network.

```
config firewall address
  edit Client-Net
    set type iprange
    set start-ip 172.20.120.100
    set end-ip 172.20.120.200
    set associated-interface port1
  end
```

4. Add a firewall address for the web server network.

```
config firewall address
  edit Web-Server-Net
    set type ipmask
    set subnet 192.168.10.0 255.255.255.0
    set associated-interface port2
  end
```

5. Add a passive WAN optimization policy.

```
config firewall policy
  edit 0
    set srcintf port1
    set dstintf port2
    set srcaddr Client-Net
    set dstaddr Web-Server-Net
    set action accept
    set service ALL
    set schedule always
    set wanopt enable
    set wanopt-detection passive
    set wanopt-passive-opt default
  end
```

6. Add a WAN optimization tunnel explicit proxy policy.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  next
end
```

Testing and troubleshooting the configuration

To test the configuration attempt to start a web browsing session between the client network and the web server network. For example, from a PC on the client network browse to the IP address of a web server on the web server network, for example <http://192.168.10.100>. Even though this address is not on the client network you should be able to connect to this web server over the WAN optimization tunnel.

If you can connect, check WAN optimization monitoring. If WAN optimization has been forwarding the traffic the WAN optimization monitor should show the protocol that has been optimized (in this case HTTP) and the reduction rate in WAN bandwidth usage.

If you can't connect you can try the following to diagnose the problem:

- Review your configuration and make sure all details such as address ranges, peer names, and IP addresses are correct.
- Confirm that the security policy on the Client-Side FortiGate unit is accepting traffic for the 192.168.10.0 network and that this security policy does not include security profiles. You can do this by checking the FortiGate session table from the dashboard. Look for sessions that use the policy ID of this policy.
- Check routing on the FortiGate units and on the client and web server networks to make sure packets can be forwarded as required. The FortiGate units must be able to communicate with each other, routing on the client network must allow packets destined for the web server network to be received by the client-side FortiGate unit, and packets from the server-side FortiGate unit must be able to reach the web servers etc.

You can use the following `get` and `diagnose` commands to display information about how WAN optimization is operating

Enter the following command to list all of the running WAN optimization tunnels and display information about each one. The command output shows 3 tunnels all created by peer-to-peer WAN optimization rules (auto-detect set to on).

```
diagnose wad tunnel list

Tunnel: id=139 type=auto
    vd=0 shared=no uses=0 state=1
    peer name= id=0 ip=unknown
    SSL-secured-tunnel=no auth-grp=test
    bytes_in=744 bytes_out=76

Tunnel: id=141 type=auto
    vd=0 shared=no uses=0 state=1
    peer name= id=0 ip=unknown
    SSL-secured-tunnel=no auth-grp=test
    bytes_in=727 bytes_out=76

Tunnel: id=142 type=auto
    vd=0 shared=no uses=0 state=1
    peer name= id=0 ip=unknown
    SSL-secured-tunnel=no auth-grp=test
    bytes_in=727 bytes_out=76

Tunnels total=3 manual=0 auto=3
```

Example adding secure tunneling to an active-passive WAN optimization configuration

This example shows how to configure two FortiGate units for active-passive WAN optimization with secure tunneling. The same authentication group is added to both FortiGate units. The authentication group includes a password (or pre-shared key) and has **Peer Acceptance** set to **Accept any Peer**. An active policy is added to

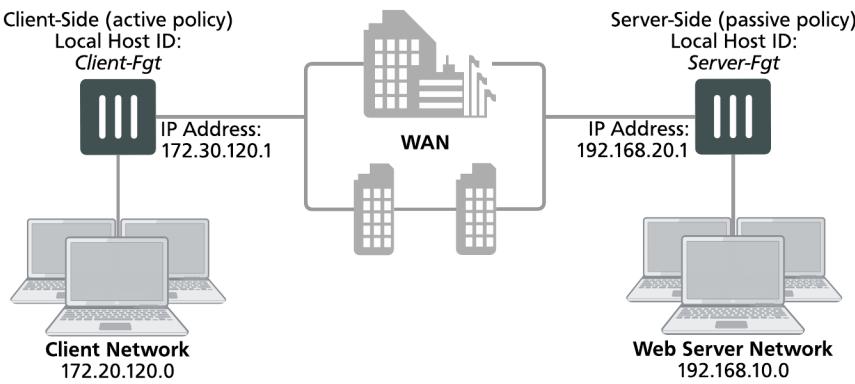
the client-side FortiGate unit and a passive policy to the server-side FortiGate unit. The active policy includes a profile that performs secure tunneling, optimizes HTTP traffic, and uses transparent mode and byte caching.

The authentication group is named **Auth-Secure-Tunnel** and the password for the pre-shared key is **2345678**. The topology for this example is shown below. This example includes web-based manager configuration steps followed by equivalent CLI configuration steps. For information about secure tunneling, see [Secure tunneling on page 1](#).

Network topology and assumptions

This example configuration includes a client-side FortiGate unit called Client-net with a WAN IP address of 172.30.120.1. This unit is in front of a network with IP address 172.20.120.0. The server-side FortiGate unit is called Web-servers and has a WAN IP address of 192.168.20.1. This unit is in front of a web server network with IP address 192.168.10.0.

Example active-passive WAN optimization and secure tunneling topology



General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Configure the client-side FortiGate unit:
 - Add peers.
 - Add an authentication group.
 - Add an active WAN optimization policy.
2. Configure the server-side FortiGate unit.
 - Add peers.
 - Add the same authentication group
 - Add a passive WAN optimization policy that applies application control.
 - Add a WAN optimization tunnel policy.

Also note that if you perform any additional actions between procedures, your configuration may have different results.

Configuring WAN optimization with secure tunneling - web-based manager

Use the following steps to configure the example WAN optimization configuration from the client-side and server-side FortiGate unit web-based manager. (CLI steps follow.)

To configure the client-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the client-side FortiGate unit:

Local Host ID	Client-Fgt
----------------------	------------

2. Select **Apply** to save your setting.
3. Select **Create New** and add a **Peer Host ID** and the **IP Address** for the server-side FortiGate unit:

Peer Host ID	Server-Fgt
IP Address	192.168.20.1

4. Select **OK**.
5. Go to **WAN Opt. & Cache > Authentication Groups** and select **Create New** to add the authentication group to be used for secure tunneling:

Name	Auth-Secure-Tunnel
Authentication Method	Pre-shared key
Password	2345678
Peer Acceptance	Accept Any Peer

6. Select **OK**.
7. Go to **WAN Opt. & Cache > Profiles** and select **Create New** to add a WAN optimization profile that enables secure tunneling and includes the authentication group:

Name	Secure-wan-op-pro
Transparent Mode	Select
Authentication Group	Auth-Secure-tunnel

8. Select the **HTTP** protocol, select **Secure Tunneling** and **Byte Caching** and set the **Port** to 80.
9. Select **OK**.
10. Go to **Policy & Objects > Addresses** and select **Create New** to add a firewall address for the client network.

Category	Address
Name	Client-Net

Type	Subnet
Subnet / IP Range	172.20.120.0/24
Interface	port1

11. Select **Create New** to add a firewall address for the web server network.

Category	Address
Address Name	Web-Server-Net
Type	Subnet
Subnet / IP Range	192.168.10.0/24
Interface	port2

12. Go to **Policy & Objects > IPv4 Policy** and select **Create New** to add an active WAN optimization security policy:

Incoming Interface	port1
Source Address	Client-Net
Outgoing Interface	port2
Destination Address	Web-Server-Net
Schedule	always
Service	HTTP
Action	ACCEPT

13. Turn on **WAN Optimization** and configure the following settings:

WAN Optimization	active
Profile	Secure-wan-opt-pro

14. Select **OK**.

To configure the server-side FortiGate unit

1. Go to **WAN Opt. & Cache > Peers** and enter a **Local Host ID** for the server-side FortiGate unit:

Local Host ID	Server-Fgt
----------------------	------------

2. Select **Apply** to save your setting.

3. Select **Create New** and add a **Peer Host ID** and the **IP Address** for the client-side FortiGate unit:

Peer Host ID	Client-Fgt
IP Address	172.30.120.1

4. Select **OK**.
5. Go to **WAN Opt. & Cache > Authentication Groups** and select **Create New** and add an authentication group to be used for secure tunneling:

Name	Auth-Secure-Tunnel
Authentication Method	Pre-shared key
Password	2345678
Peer Acceptance	Accept Any Peer

6. Select **OK**.
7. Go to **Policy & Objects > Addresses** and select **Create New** to add a firewall address for the client network.

Category	Address
Name	Client-Net
Type	Subnet
Subnet / IP Range	172.20.120.0/24
Interface	port1

8. Select **Create New** to add a firewall address for the web server network.

Category	Address
Address Name	Web-Server-Net
Type	Subnet
Subnet / IP Range	192.168.10.0/24
Interface	port2

9. Select **OK**.
10. Select **Create New** to add a passive WAN optimization policy that applies application control.

Incoming Interface	port2
Source Address	Client-Net
Outgoing Interface	port1
Destination Address	Web-Server-Net

Schedule	always
Service	ALL
Action	ACCEPT

11. Turn on **WAN Optimization** and configure the following settings:

WAN Optimization	passive
Passive Option	default

12. Select **OK**.

13. From the CLI enter the following command to add a WAN optimization tunnel explicit proxy policy.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  next
end
```

Configuring WAN optimization with secure tunneling - CLI

Use the following steps to configure the example WAN optimization configuration from the client-side and server-side FortiGate unit CLI.

To the client-side FortiGate unit

1. Add the Local Host ID to the client-side FortiGate configuration:

```
config wanopt settings
  set host-id Client-Fgt
end
```

2. Add the server-side Local Host ID to the client-side peer list:

```
config wanopt peer
  edit Server-Fgt
  set ip 192.168.20.1
end
```

3. Add a new authentication group to be used for secure tunneling:

```
config wanopt auth-group
  edit Auth-Secure-Tunnel
    set auth-method psk
    set psk 2345678
end
```

Leave **peer-accept** at its default value.

-
4. Add a WAN optimization profile that enables secure tunneling and includes the authentication group, enables HTTP protocol optimization, and enables secure tunneling and byte caching for HTTP traffic:

```
config wanopt profile
  edit Secure-wan-op-pro
    set auth-group Auth-Secure-Tunnel
    config http
      set status enable
      set secure-tunnel enable
      set byte-caching enable
      set port 80
    end
  end
```

5. Add a firewall address for the client network.

```
config firewall address
  edit Client-Net
    set type ipmask
    set subnet 172.20.120.0 255.255.255.0
    set associated-interface port1
  end
```

6. Add a firewall address for the web server network.

```
config firewall address
  edit Web-Server-Net
    set type ipmask
    set subnet 192.168.10.0 255.255.255.0
    set associated-interface port2
  end
```

7. Add an active WAN optimization security policy that includes the WAN optimization profile that enables secure tunneling and that applies virus scanning:

```
config firewall policy
  edit 0
    set srcintf port1
    set dstintf port2
    set srcaddr Client-Net
    set dstaddr Web-Server-Net
    set action accept
    set service HTTP
    set schedule always
    set wanopt enable
    set wanopt-detection active
    set wanopt-profile Secure-wan-opt-pro
  end
```

To configure the server-side FortiGate unit

1. Add the Local Host ID to the server-side FortiGate configuration:

```
config wanopt settings
  set host-id Server-Fgt
end
```

2. Add the client-side Local Host ID to the server-side peer list:

```
config wanopt peer
  edit Client-Fgt
  set ip 172.20.120.1
```

```
end
```

3. Add an authentication group to be used for secure tunneling:

```
config wanopt auth-group
  edit Auth-Secure-Tunnel
    set auth-method psk
    set psk 2345678
  end
```

Leave **peer-accept** at its default value.

4. Add a firewall address for the client network.

```
config firewall address
  edit Client-Net
    set type ipmask
    set subnet 172.20.120.0 255.255.255.0
    set associated-interface port1
  end
```

5. Add a firewall address for the web server network.

```
config firewall address
  edit Web-Server-Net
    set type ipmask
    set subnet 192.168.10.0 255.255.255.0
    set associated-interface port2
  end
```

6. Add a passive WAN optimization policy.

```
config firewall policy
  edit 0
    set srcintf port1
    set dstintf port2
    set srcaddr Client-Net
    set dstaddr Web-Server-Net
    set action accept
    set service ALL
    set schedule always
    set wanopt enable
    set wanopt-detection passive
    set wanopt-passive-opt default
  end
```

7. Add a WAN optimization tunnel explicit proxy policy.

```
configure firewall proxy-policy
  edit 0
    set proxy wanopt
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
  next
end
```

Peers and authentication groups

All communication between WAN optimization peers begins with one WAN optimization peer (or client-side FortiGate unit) sending a WAN optimization tunnel request to another peer (or server-side FortiGate unit). During this process, the WAN optimization peers identify and optionally authenticate each other.

Basic WAN optimization peer requirements

WAN optimization requires the following configuration on each peer. For information about configuring local and peer host IDs, see [Basic WAN optimization peer requirements on page 319](#).

- The peer must have a unique host ID.
- Unless authentication groups are used, peers authenticate each other using host ID values. Do not leave the local host ID at its default value.
- The peer must know the host IDs and IP addresses of all of the other peers that it can start WAN optimization tunnels with. This does not apply if you use authentication groups that accept all peers.
- All peers must have the same local certificate installed on their FortiGate units if the units authenticate by local certificate. Similarly, if the units authenticate by pre-shared key (password), administrators must know the password. The type of authentication is selected in the authentication group. This applies only if you use authentication groups.

Accepting any peers

Strictly speaking, you do not need to add peers. Instead you can configure authentication groups that accept any peer. However, for this to work, both peers must have the same authentication group (with the same name) and both peers must have the same certificate or pre-shared key.

Accepting any peer is useful if you have many peers or if peer IP addresses change. For example, you could have FortiGate units with dynamic external IP addresses (using DHCP or PPPoE). For most other situations, this method is not recommended and is not a best practice as it is less secure than accepting defined peers or a single peer. For more information, see [Basic WAN optimization peer requirements on page 319](#).

How FortiGate units process tunnel requests for peer authentication

When a client-side FortiGate unit attempts to start a WAN optimization tunnel with a peer server-side FortiGate unit, the tunnel request includes the following information:

- the client-side local host ID
- the name of an authentication group, if included in the rule that initiates the tunnel
- if an authentication group is used, the authentication method it specifies: pre-shared key or certificate
- the type of tunnel (secure or not).

For information about configuring the local host ID, peers and authentication groups, see [How FortiGate units process tunnel requests for peer authentication on page 319](#) and [How FortiGate units process tunnel requests for peer authentication on page 319](#).

The authentication group is optional unless the tunnel is a secure tunnel. For more information, see [How FortiGate units process tunnel requests for peer authentication on page 319](#).

If the tunnel request includes an authentication group, the authentication will be based on the settings of this group as follows:

- The server-side FortiGate unit searches its own configuration for the name of the authentication group in the tunnel request. If no match is found, the authentication fails.
- If a match is found, the server-side FortiGate unit compares the authentication method in the client and server authentication groups. If the methods do not match, the authentication fails.
- If the authentication methods match, the server-side FortiGate unit tests the peer acceptance settings in its copy of the authentication group.
- If the setting is **Accept Any Peer**, the authentication is successful.
- If the setting is **Specify Peer**, the server-side FortiGate unit compares the client-side local host ID in the tunnel request with the peer name in the server-side authentication group. If the names match, authentication is successful. If a match is not found, authentication fails.
- If the setting is **Accept Defined Peers**, the server-side FortiGate unit compares the client-side local host ID in the tunnel request with the server-side peer list. If a match is found, authentication is successful. If a match is not found, authentication fails.

If the tunnel request does not include an authentication group, authentication will be based on the client-side local host ID in the tunnel request. The server-side FortiGate unit searches its peer list to match the client-side local host ID in the tunnel request. If a match is found, authentication is successful. If a match is not found, authentication fails.

If the server-side FortiGate unit successfully authenticates the tunnel request, the server-side FortiGate unit sends back a tunnel setup response message. This message includes the server-side local host ID and the authentication group that matches the one in the tunnel request.

The client-side FortiGate unit then performs the same authentication procedure as the server-side FortiGate unit did. If both sides succeed, tunnel setup continues.

Configuring peers

When you configure peers, you first need to add the local host ID that identifies the FortiGate unit for WAN optimization and then add the peer host ID and IP address of each FortiGate unit with which a FortiGate unit can create WAN optimization tunnels.

To configure WAN optimization peers - web-based manager:

1. Go to **WAN Opt. & Cache > Peers**.
2. For **Local Host ID**, enter the local host ID of **this** FortiGate unit and select **Apply**. If you add this FortiGate unit as a peer to another FortiGate unit, use this ID as its **peer** host ID.

The local or host ID can contain up to 25 characters and can include spaces.

3. Select **Create New** to add a new peer.

4. For **Peer Host ID**, enter the peer host ID of the peer FortiGate unit. This is the local host ID added to the peer FortiGate unit.
5. For **IP Address**, add the IP address of the peer FortiGate unit. This is the source IP address of tunnel requests sent by the peer, usually the IP address of the FortiGate interface connected to the WAN.
6. Select **OK**.

To configure WAN optimization peers - CLI:

In this example, the local host ID is named `HQ_Peer` and has an IP address of `172.20.120.100`. Three peers are added, but you can add any number of peers that are on the WAN.

1. Enter the following command to set the local host ID to `HQ_Peer`.

```
config wanopt settings
    set host-id HQ_peer
end
```

2. Enter the following commands to add three peers.

```
config wanopt peer
    edit Wan_opt_peer_1
        set ip 172.20.120.100
    next
    edit Wan_opt_peer_2
        set ip 172.30.120.100
    next
    edit Wan_opt_peer_3
        set ip 172.40.120.100
end
```

Configuring authentication groups

You need to add authentication groups to support authentication and secure tunneling between WAN optimization peers.

To perform authentication, WAN optimization peers use a certificate or a pre-shared key added to an authentication group so they can identify each other before forming a WAN optimization tunnel. Both peers must have an authentication group with the same name and settings. You add the authentication group to a peer-to-peer or active rule on the client-side FortiGate unit. When the server-side FortiGate unit receives a tunnel start request from the client-side FortiGate unit that includes an authentication group, the server-side FortiGate unit finds an authentication group in its configuration with the same name. If both authentication groups have the same certificate or pre-shared key, the peers can authenticate and set up the tunnel.

Authentication groups are also required for secure tunneling.

To add authentication groups, go to **WAN Opt. & Cache > Authentication Groups**.

To add an authentication group - web-based manager:

Use the following steps to add any kind of authentication group. It is assumed that if you are using a local certificate to authenticate, it is already added to the FortiGate unit

1. Go to **WAN Opt. & Cache > Authentication Groups**.
2. Select **Create New**.

3. Add a **Name for the authentication group.**

You will select this name when you add the authentication group to a WAN optimization rule.

4. Select the **Authentication Method.**

Select **Certificate** if you want to use a certificate to authenticate and encrypt WAN optimization tunnels. You must select a local certificate that has been added to this FortiGate unit. (To add a local certificate, go to **System** > **Certificates**.) Other FortiGate units that participate in WAN optimization tunnels with this FortiGate unit must have an authentication group with the same name and certificate.

Select **Pre-shared key** if you want to use a pre-shared key or password to authenticate and encrypt WAN optimization tunnels. You must add the **Password** (or pre-shared key) used by the authentication group. Other FortiGate units that participate in WAN optimization tunnels with this FortiGate unit must have an authentication group with the same name and password. The password must contain at least 6 printable characters and should be known only by network administrators. For optimum protection against currently known attacks, the key should consist of a minimum of 16 randomly chosen alphanumeric characters.

5. Configure **Peer Acceptance for the authentication group.**

Select **Accept Any Peer** if you do not know the peer host IDs or IP addresses of the peers that will use this authentication group. This setting is most often used with FortiGate units that do not have static IP addresses, for example units that use DHCP.

Select **Accept Defined Peers** if you want to authenticate with peers added to the peer list only.

Select **Specify Peer** and select one of the peers added to the peer list to authenticate with the selected peer only.

6. Select **OK.**

7. Add the authentication group to a WAN optimization rule to apply the authentication settings in the authentication group to the rule.

To add an authentication group that uses a certificate- CLI:

Enter the following command to add an authentication group that uses a certificate and can authenticate all peers added to the FortiGate unit configuration.

In this example, the authentication group is named `auth_grp_1` and uses a certificate named `Example_Cert`.

```
config wanopt auth-group
    edit auth_grp_1
        set auth-method cert
        set cert Example_Cert
        set peer-accept defined
    end
```

To add an authentication group that uses a pre-shared key - CLI:

Enter the following command to add an authentication group that uses a pre-shared key and can authenticate only the peer added to the authentication group.

In this example, the authentication group is named `auth_peer`, the peer that the group can authenticate is named `Server_net`, and the authentication group uses `123456` as the pre-shared key. In practice you should use a more secure pre-shared key.

```
config wanopt auth-group
  edit auth_peer
    set auth-method psk
    set psk 123456
    set peer-accept one
    set peer Server_net
  end
```

To add an authentication group that accepts WAN optimization connections from any peer - web-based manager

Add an authentication group that accepts any peer for situations where you do not have the **Peer Host IDs or IP Addresses** of the peers that you want to perform WAN optimization with. This setting is most often used for WAN optimization with FortiGate units that do not have static IP addresses, for example units that use DHCP. An authentication group that accepts any peer is less secure than an authentication group that accepts defined peers or a single peer.

The example below sets the authentication method to **Pre-shared key**. You must add the same password to all FortiGate units using this authentication group.

1. Go to **WAN Opt. & Cache > Authentication Groups**.
2. Select **Create New** to add a new authentication group.
3. Configure the authentication group:

Name	Specify any name.
Authentication Method	Pre-shared key
Password	Enter a pre-shared key.
Peer Acceptance	Accept Any Peer

To add an authentication group that accepts WAN optimization connections from any peer - CLI:

In this example, the authentication group is named `auth_grp_1`. It uses a certificate named `WAN_Cert` and accepts any peer.

```
config wanopt auth-group
  edit auth_grp_1
    set auth-method cert
    set cert WAN_Cert
    set peer-accept any
  end
```

Secure tunneling

You can configure WAN optimization rules to use AES-128bit-CBC SSL to encrypt the traffic in the WAN optimization tunnel. WAN optimization uses FortiASIC acceleration to accelerate SSL decryption and encryption

of the secure tunnel. Peer-to-peer secure tunnels use the same TCP port as non-secure peer-to-peer tunnels (TCP port 7810).

To use secure tunneling, you must select **Enable Secure Tunnel** in a WAN optimization rule and add an authentication group. The authentication group specifies the certificate or pre-shared key used to set up the secure tunnel. The **Peer Acceptance** setting of the authentication group does not affect secure tunneling.

The FortiGate units at each end of the secure tunnel must have the same authentication group with the same name and the same configuration, including the same pre-shared key or certificate. To use certificates you must install the same certificate on both FortiGate units.

For active-passive WAN optimization you can select **Enable Secure Tunnel** only in the active rule. In peer-to-peer WAN optimization you select **Enable Secure Tunnel** in the WAN optimization rule on both FortiGate units. For information about active-passive and peer-to-peer WAN optimization, see [Manual \(peer-to-peer\) and active-passive WAN optimization on page 1](#)

For a secure tunneling configuration example, see [Example: Adding secure tunneling to an active-passive WAN optimization configuration on page 1](#).

Monitoring WAN optimization peer performance

The WAN optimization peer monitor lists all of the WAN optimization peers that a FortiGate unit can perform WAN optimization with. These include peers manually added to the configuration as well as discovered peers.

The monitor lists each peer's name, IP address, and peer type. The peer type indicates whether the peer was manually added or discovered. To show WAN optimization performance, for each peer the monitor lists the percent of traffic reduced by the peer in client-side WAN optimization configurations and in server-side configurations (also called gateway configurations).

To view the peer monitor, go to **WAN Opt. & Cache > Peer Monitor**.

Web cache concepts

FortiGate web caching is a form of object caching that accelerates web applications and web servers by reducing bandwidth usage, server load, and perceived latency. Web caching supports caching of HTTP 1.0 and HTTP 1.1 web sites. See [RFC 2616](#) for information about web caching for HTTP 1.1.



Web caching supports caching of Flash content over HTTP but does not cache audio and video streams including Flash videos and streaming content that use native streaming protocols such as RTMP.

The first time a file is received by web caching it is cached in the format it is received in, whether it be compressed or uncompressed. When the same file is requested by a client but in a different compression format, the cached file is converted to the new compressed format before being sent to the client.

There are three significant advantages to using web caching to improve HTTP and WAN performance:

- reduced bandwidth consumption because fewer requests and responses go over the WAN or Internet.
- reduced web server load because there are fewer requests for web servers to handle.
- reduced latency because responses for cached requests are available from a local FortiGate unit instead of from across the WAN or Internet.

You can use web caching to cache any web traffic that passes through the FortiGate unit, including web pages from web servers on a LAN, WAN or on the Internet. You apply web caching by enabling the web caching option in any security policy. When enabled in a security policy, web caching is applied to all HTTP sessions accepted by the security policy. If the security policy is an explicit web proxy security policy, the FortiGate unit caches explicit web proxy sessions.

Turning on web caching for HTTP and HTTPS traffic

Web caching can be applied to any HTTP or HTTPS traffic by enabling web caching in a security policy that accepts the traffic. This includes IPv4, IPv6, WAN optimization and explicit web proxy traffic. Web caching caches all HTTP traffic accepted by a policy on TCP port 80.

You can add web caching to a policy to:

- Cache Internet HTTP traffic for users on an internal network to reduce Internet bandwidth use. Do this by selecting the web cache option for security policies that allow users on the internal network to browse web sites on the Internet.
- Reduce the load on a public facing web server by caching objects on the FortiGate unit. This is a reverse proxy with web caching configuration. Do this by selecting the web cache option for a security policy that allows users on the Internet to connect to the web server.
- Cache outgoing explicit web proxy traffic when the explicit proxy is used to proxy users in an internal network who are connecting to the web servers on the Internet. Do this by selecting the web cache option for explicit web proxy security policies that allow users on the internal network to browse web sites on the Internet.
- Combine web caching with WAN optimization. You can enable web caching in any WAN optimization security policy. This includes manual, active, and passive WAN optimization policies and WAN optimization tunnel policies.

You can enable web caching on both the client-side and the server-side FortiGate units or on just one or the other. For optimum performance you can enable web caching on both the client-side and server-side FortiGate units. In this way only uncached content is transmitted through the WAN optimization tunnel. All cached content is accessed locally by clients from the client side FortiGate unit.



One important use for web caching is to cache software updates (for example, Windows Updates or iOS updates). When updates occur a large number of users may all be trying to download these updates at the same time. Caching these updates will be a major performance improvement and also have a potentially large impact on reducing Internet bandwidth use. You may want to adjust the maximum cache object size to make sure these updates are cached. See [Turning on web caching for HTTP and HTTPS traffic on page 325](#).

Turning on web caching for HTTPS traffic

Web caching can also cache the content of HTTPS traffic on TCP port 443. With HTTPS web caching, the FortiGate unit receives the HTTPS traffic on behalf of the client, opens up the encrypted traffic and extracts content to be cached. Then FortiGate unit re-encrypts the traffic and sends it on to its intended recipient. It is very similar to a man-in-the-middle attack.

You enable HTTPS web caching from the CLI in a security policy or an explicit proxy policy that accepts the traffic to be cached using `webcache-https`. For a firewall policy:

```
config firewall policy
  edit 0
  .
  .
  .
  set webcache enable
  set webcache-https enable
  .
  .
  .
end
```

For an explicit web proxy policy:

```
config firewall proxy-policy
  edit 0
    set proxy explicit-web
  .
  .
  .
  set webcache enable
  set webcache-https enable
  .
  .
  .
end
```



The `webcache-https` field is available only if `webcache` is enabled.

Web caching for HTTPS traffic is not supported if WAN optimization or FTP proxy is enabled: i.e., if `srcintf` is `ftp-proxy` or `wanopt`.

The `any` setting causes the FortiGate unit to re-encrypt the traffic with the FortiGate unit's certificate rather than the original certificate. This configuration can cause errors for HTTPS clients because the name on the certificate does not match the name on the web site.

You can stop these errors from happening by configuring HTTPS web caching to use the web server's certificate by setting `webcache-https` to `ssl-server`. This option is available for both firewall policies and explicit web proxy policies.

```
config firewall policy
  edit 0
  .
  .
  .
  set webcache enable
  set webcache-https enable
  .
  .
  .
end
```

The `ssl-server` option causes the FortiGate unit to re-encrypt the traffic with a certificate that you imported into the FortiGate unit. You can add certificates using the following command:

```
config firewall ssl-server
  edit corporate-server
    set ip <Web-Server-IP>
    set port 443
    set ssl-mode { full | half}
    set ssl-cert <Web-Server-Cert>
end
```

Where:

`Web-Server-IP` is the web server's IP address.

`Web-Server-Cert` is a web server certificate imported into the FortiGate unit.

The SSL server configuration also determines whether the SSL server is operating in half or full mode and the port used for the HTTPS traffic.

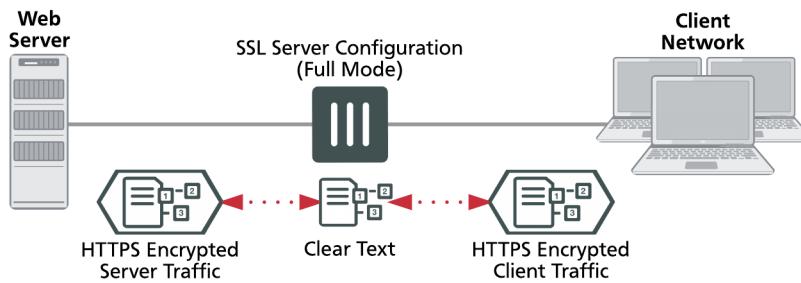
You can add multiple SSL server certificates in this way. When web caching processing an SSL stream if it can find a certificate that matches the web server IP address and port of one of the added SSL servers; that certificate is used to encrypt the SSL traffic before sending it to the client. As a result the client does not generate SSL certificate errors.

Web caching uses the FortiGate unit's FortiASIC to accelerate SSL decryption/encryption performance.

Full mode SSL server configuration

The `ssl-mode` option determines whether the SSL server operates in half or full mode. In full mode the FortiGate unit performs both decryption and encryption of the HTTPS traffic. The full mode sequence is shown below.

Full mode SSL server configuration



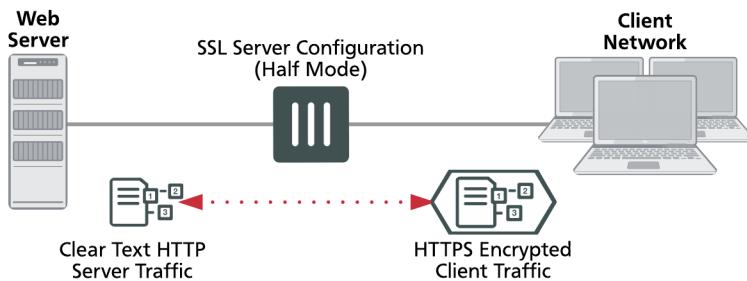
In full mode the FortiGate unit is acting as a man in the middle, decrypting and encrypting the traffic. So both the client and the web server see encrypted packets.

Usually the port of the encrypted HTTPS traffic is always 443. However, in the SSL server configuration you can set the port used for HTTPS traffic. This port is not altered by the SSL Server. So for example, if the SSL Server receives HTTPS traffic on port 443, the re-encrypted traffic forwarded to the FortiGate unit to the server or client will still use port 443.

Half mode SSL server configuration

In half mode, the FortiGate unit only performs one encryption or decryption action. If HTTP packets are received, the half mode SSL server encrypts them and converts them to HTTPS packets. If HTTPS packets are received, the SSL server decrypts them and converts them to HTTP packets.

Half mode SSL server configuration



In half mode, the FortiGate unit is acting like an SSL accelerator, offloading HTTPS decryption from the web server to the FortiGate unit. Since FortiGate units can accelerate SSL processing, the end result could be improved web site performance.

Usually the port of the encrypted traffic is always 443. However, in the SSL server configuration you can set the port used for HTTPS traffic. No matter what port is used for the HTTPS traffic, the decrypted HTTP traffic uses port 80.

Changing the ports on which to look for HTTP and HTTPS traffic to cache

By default FortiOS assumes HTTP traffic uses TCP port 80 and HTTPS traffic uses port 443. So web caching caches all HTTP traffic accepted by a policy on TCP port 80 and all HTTPS traffic on TCP port 443. If you want to cache HTTP or HTTPS traffic on other ports, you can enable security profiles for the security policy and configure a proxy options profile to that looks for HTTP and HTTPS traffic on other TCP ports. To configure a proxy options profile go to **Network > Explicit Proxy**.

Setting the HTTP port to **Any** in a proxy options profile is not compatible with web caching. If you set the HTTP port to any, web caching only caches HTTP traffic on port 80.

Web caching and HA

You can configure web caching on a FortiGate HA cluster. The recommended best practice HA configuration for web caching is active-passive mode. When the cluster is operating, all web caching sessions are processed by the primary unit only. Even if the cluster is operating in active-active mode, HA does not load-balance web caching sessions.

In a cluster, only the primary unit stores the web cache database. The databases is not synchronized to the subordinate units. So, after a failover, the new primary unit must build its web cache.

Web caching and memory usage

To accelerate and optimize disk access and to provide better throughput and less latency, web caching uses provisioned memory to reduce disk I/O and increase disk I/O efficiency. In addition, web caching requires a small amount of additional memory per session for comprehensive flow control logic and efficient traffic forwarding.

When web caching is enabled you will see a reduction in available memory. The reduction increases when more web caching sessions are being processed. If you are thinking of enabling web caching on an operating FortiGate unit, make sure its memory usage is not maxed out during high traffic periods.

In addition to using the system dashboard to see the current memory usage you can use the `get test wad 2` command to see how much memory is currently being used by web caching. See [get test {wad | wccpd} <test_level> on page 1](#) for more information.

Changing web cache settings

In most cases, the default settings for the WAN optimization web cache are acceptable. However, you may want to change them to improve performance or optimize the cache for your configuration. To change these settings, go to **WAN Opt. & Cache > Settings**.

From the FortiGate CLI, you can use the `config wanopt webcache` command to change these WAN optimization web cache settings.



For more information about many of these web cache settings, see [RFC 2616](#).

Always revalidate

Select to always revalidate requested cached objects with content on the server before serving them to the client.

Max cache object size

Set the maximum size of objects (files) that are cached. The default size is 512000 KB and the range is 1 to 4294967 KB. This setting determines the maximum object size to store in the web cache. Objects that are larger than this size are still delivered to the client but are not stored in the FortiGate web cache.

For most web traffic the default maximum cache object size is recommended. However, since web caching can also cache larger objects such as Windows updates, Mac OS updates, iOS updates or other updates delivered using HTTP you might want to increase the object size to make sure these updates are cached. Caching these updates can save a lot of Internet bandwidth and improve performance when major updates are released by these vendors.

Negative response duration

Set how long in minutes that the FortiGate unit caches error responses from web servers. If error responses are cached, then subsequent requests to the web cache from users will receive the error responses regardless of the actual object status.

The default is 0, meaning error responses are not cached. The content server might send a client error code (4xx HTTP response) or a server error code (5xx HTTP response) as a response to some requests. If the web cache is configured to cache these negative responses, it returns that response in subsequent requests for that page or image for the specified number of minutes.

Fresh factor

Set the fresh factor as a percentage. The default is 100, and the range is 1 to 100%. For cached objects that do not have an expiry time, the web cache periodically checks the server to see if the objects have expired. The higher the **Fresh Factor** the less often the checks occur.

For example, if you set the **Max TTL** value and **Default TTL** to 7200 minutes (5 days) and set the **Fresh Factor** to 20, the web cache check the cached objects 5 times before they expire, but if you set the **Fresh Factor** to 100, the web cache will check once.

Max TTL

The maximum amount of time (Time to Live) an object can stay in the web cache without the cache checking to see if it has expired on the server. The default is 7200 minutes (120 hours or 5 days) and the range is 1 to 5256000 minutes (5256000 minutes in a year).

Min TTL

The minimum amount of time an object can stay in the web cache before the web cache checks to see if it has expired on the server. The default is 5 minutes and the range is 1 to 5256000 minutes (5256000 minutes in a year).

Default TTL

The default expiry time for objects that do not have an expiry time set by the web server. The default expiry time is 1440 minutes (24 hours) and the range is 1 to 5256000 minutes (5256000 minutes in a year).

Proxy FQDN

The fully qualified domain name (FQDN) for the proxy server. This is the domain name to enter into browsers to access the proxy server. This field is for information only can be changed from the explicit web proxy configuration.

Max HTTP request length

The maximum length of an HTTP request that can be cached. Larger requests will be rejected. This field is for information only can be changed from the explicit web proxy configuration.

Max HTTP message length

The maximum length of an HTTP message that can be cached. Larger messages will be rejected. This field is for information only can be changed from the explicit web proxy configuration.

Ignore

Select the following options to ignore some web caching features.

If-modified-since	By default, if the time specified by the if-modified-since (IMS) header in the client's conditional request is greater than the last modified time of the object in the cache, it is a strong indication that the copy in the cache is stale. If so, HTTP does a conditional GET to the Overlay Caching Scheme (OCS), based on the last modified time of the cached object. Enable ignoring if-modified-since to override this behavior.
HTTP 1.1 conditionals	HTTP 1.1 provides additional controls to the client over the behavior of caches toward stale objects. Depending on various cache-control headers, the FortiGate unit can be forced to consult the OCS before serving the object from the cache. For more information about the behavior of cache-control header values, see RFC 2616 . Enable ignoring HTTP 1.1 Conditionals to override this behavior.

Pragma-no-cache	Typically, if a client sends an HTTP GET request with a pragma no-cache (PNC) or cache-control no-cache header, a cache must consult the OCS before serving the content. This means that the FortiGate unit always re-fetches the entire object from the OCS, even if the cached copy of the object is fresh. Because of this behavior, PNC requests can degrade performance and increase server-side bandwidth utilization. However, if you enable ignoring Pragma-no-cache, then the PNC header from the client request is ignored. The FortiGate unit treats the request as if the PNC header is not present.
IE Reload	Some versions of Internet Explorer issue Accept / header instead of Pragma no-cache header when you select Refresh . When an Accept header has only the / value, the FortiGate unit treats it as a PNC header if it is a type-N object. Enable ignoring IE reload to cause the FortiGate unit to ignore the PNC interpretation of the Accept / header.

Cache expired objects

Applies only to type-1 objects. When this option is selected, expired type-1 objects are cached (if all other conditions make the object cacheable).

Revalidated pragma-no-cache

The pragma-no-cache (PNC) header in a client's request can affect how efficiently the FortiGate unit uses bandwidth. If you do not want to completely ignore PNC in client requests (which you can do by selecting to ignore Pragma-no-cache, above), you can nonetheless lower the impact on bandwidth usage by selecting **Revalidate Pragma-no-cache**.

When you select **Revalidate Pragma-no-cache**, a client's non-conditional PNC-GET request results in a conditional GET request sent to the OCS if the object is already in the cache. This gives the OCS a chance to return the 304 Not Modified response, which consumes less server-side bandwidth, because the OCS has not been forced to otherwise return full content.

By default, **Revalidate Pragma-no-cache** is disabled and is not affected by changes in the top-level profile.

Most download managers make byte-range requests with a PNC header. To serve such requests from the cache, you should also configure byte-range support when you configure the **Revalidate pragma-no-cache** option.

Web cache configuration

Forwarding URLs to forwarding servers and exempting web sites from web caching

You can go to **Network > Explicit Proxy** and use the URL match list to forward URL patterns to forwarding servers and create a list of URLs that are exempt from web caching.

Forwarding URLs and URL patterns to forwarding servers

As part of configuring the explicit web proxy you can configure proxy chaining by adding web proxy forwarding servers. See [Proxy chaining \(web proxy forwarding servers\)](#).

You can then use the URL match list to always forward explicit web proxy traffic destined for configured URLs or URL patterns to one of these forwarding servers. For example, you might want to forward all traffic for a specific country to a proxy server located in that country.

To forward traffic destined for a URL to a forwarding server that you have already added, go to **Network > Explicit Proxy** and select **Create New**. Add a name for the URL match entry and enter the URL or URL pattern. You can use wildcards such as * and ? and you can use a numeric IP address. Select **Forward to Server** and select a web proxy forwarding server from the list.

You can also exempt the URL or URL pattern from web caching.

Use the following command to forward all .ca traffic to a proxy server and all .com traffic to another proxy server.

```
config web-proxy url-match
  edit "com"
    set forward-server "server-commercial"
    set url-pattern "com"
  next
  edit "ca"
    set forward-server "server-canada"
    set url-pattern "ca"
  next
  edit "www.google.ca"
    set cache-exemption enable
    set url-pattern "www.google.ca"
  next
end
```

Exempting web sites from web caching

You may want to exempt some URLs from web caching for a number of reasons. For example, if your users access websites that are not compatible with FortiGate web caching you can add the URLs of these web sites to the web caching exempt list. You can add URLs and numeric IP addresses to the web cache exempt list.

You can also add URLs to the web cache exempt list by going to **Network > Explicit Proxy**, going to the **URL Match List**

Name	URL Pattern	Cache Exemption	Forward Server	Status	Comments
No matching entries found					

and selecting **Create New**. Add a URL pattern to be exempt and select **Exempt from Cache**.

New URL Match Entry

Name	<input type="text"/>
Comments	<input type="text"/> 0/255
URL Pattern	<input type="text"/>
Forward to Server	<input type="checkbox"/>
Exempt from Cache	<input checked="" type="checkbox"/>
Enable this URL	<input checked="" type="checkbox"/>

You can also add URLs and addresses to be exempt from caching using the CLI. Enter the following command to add `www.example.com` to the web cache exempt list:

```
config web-proxy url-match
  set cache-exemption enable
  set url-pattern www.example.com
end
```

Exempting specific files from caching

You can exempt files from being cached, so long as you specify its full URL. Enter the following command to add the URL, with the file extension (in this example, `.exe`), to the web cache exempt list:

```
config web-proxy url-match
  edit "exe"
    set url-pattern "iavs9x.u.avast.com/custom/iavs9x/20160613t1237z/avast_free_
      antivirus_setup_online.exe"
    set cache-exemption enable
  next
end
```



You cannot use wildcards to exempt file extensions in general from caching.

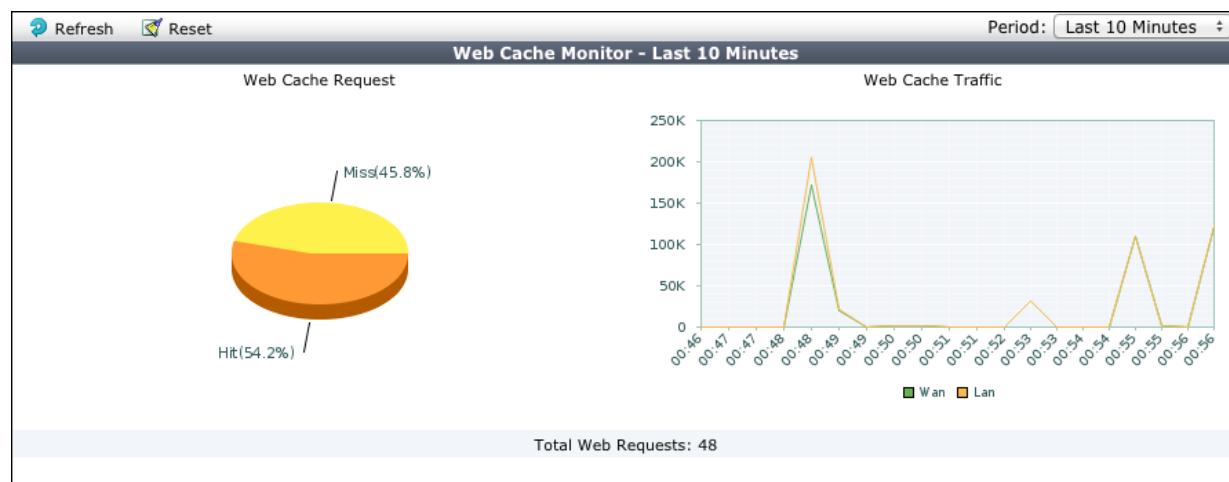
Monitoring web caching performance

The web cache monitor shows the percentage of web cache requests that retrieved content from the cache (hits) and the percentage that did not receive content from the cache (misses). A higher the number of hits usually indicates that the web cache is being more effective at reducing WAN traffic.

The web cache monitor also shows a graph of web traffic on the WAN and LAN. A lower WAN line on the graph indicates the web cache is reducing traffic on the WAN. The web cache monitor also displays the total number of web requests processed by the web cache.

To view the web cache monitor, go to **Monitor > Cache Monitor**.

Web cache monitor



Example web caching of HTTP and HTTPS Internet content for users on an internal network

This example describes how to configure web caching of HTTP and HTTPS for users on a private network connecting to the Internet.

Network topology and assumptions

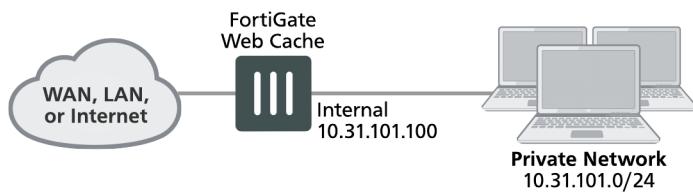
This example includes a client network with subnet address 10.31.101.0 connecting to web servers on the Internet. All of the users on the private network access the Internet though a single general security policy on the FortiGate unit that accepts all sessions connecting to the Internet. Web caching for HTTP and HTTPS traffic is added to this security policy.

Since users on the private network have unrestricted access to the Internet and can be accessing many web servers the `webcache-https` is set to `any` and users may see error messages on their web browsers when accessing HTTPS content.

The GUI is less versatile than the CLI so the example instructions for the GUI give settings for one port for each protocol, while the CLI example shows how to use multiple ports.

The example also describes how to configure the security policy to cache HTTP traffic on port 80 and 8080 in the CLI, by adding a proxy options profile that looks for HTTP traffic on TCP ports 80 and 8080. The example also describes how to configure the security policy to cache HTTPS traffic on port 443 and 8443 using the same proxy options profile.

Example web caching topology



General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Add HTTP web caching to the security policy that all users on the private network use to connect to the Internet.
2. Add HTTPS web caching.
3. Add a protocol options profile to look for HTTP traffic on ports 80 and 8080 and HTTPS traffic on ports 443 and 8443 and add this protocol options profile to the security policy.

If you perform any additional actions between procedures, your configuration may have different results.

Configuration steps - web-based manager

Use the following steps to configure the example configuration from the FortiGate web-based manager.

To add HTTP web caching to a security policy

1. Go to **Policy & Objects > IPv4 Policy** and add a security policy that allows all users on the internal network to access the Internet.

Incoming Interface	Internal
Outgoing Interface	wan1
Source	all
Destination	all
Schedule	always
Service	ALL
Action	ACCEPT

2. Toggle **NAT** to enabled, and select **Use Outgoing Interface Address**.
3. Turn on **Web cache**.
4. Select **OK**.

To add HTTPS web caching

1. From the CLI enter the following command to add HTTPS web caching to the policy.

Assume the index number of the policy is 5.

```
config firewall policy
  edit 5
    set webcache-https any
  end
```

To cache HTTP traffic on port 80 and HTTPS on 8443

1. Go to **Network > Explicit Proxy** and edit the Explicit Proxy options profile.
2. Under **Explicit Web Proxy** ,
 - For the **HTTP port**, enter 80.
 - For **HTTPS port**, select **Specify** and enter 8443 in the field.
3. Click on **Apply**.



You need to use the CLI to add the protocol options profile unless you also add a security profile that uses proxy-based inspection.

Configuration steps - CLI

Use the following steps to configure the example configuration from the FortiGate CLI.

To add HTTP and HTTPS web caching to a security policy

1. Enter the following command to add a security policy that allows all users on the internal network to access the Internet and that includes web caching of HTTP and HTTPS traffic.

```
config firewall policy
  edit 0
    set srcintf internal
    set srcaddr all
    set dstintf wan1
    set distinf all
    set schedule always
    set service ALL
    set action accept
    set nat enable
    set webcache enable
    set webcache-https any
  end
```

To cache HTTP traffic on port 80 and 8080 and HTTPS traffic on ports 443 and 8443

1. Enter the following command to edit the **default** proxy options profile to configure it to look for HTTP traffic on ports 80 and 8080:

```
config firewall profile-protocol-options
  edit default
    config http
      set status enable
      set ports 80 8080
```

```
end
```

2. Enter the following command to edit the **certification-inspection** SSL SSH options profile to configure it to look for HTTPS traffic on ports 443 and 8443:

```
config firewall ssl-ssh-profile
  edit certificate-inspection
    config https
      set status certificate-inspection
      set ports 443 8443
    end
```

3. Enter the following command to add the **default** proxy options profile and the **certificate-inspection** SSL SSH profile to the firewall policy.

```
config firewall policy
  edit 5
    set utm-status enable
    set profile-protocol-options default
    set ssl-ssh-profile certificate-inspection
  end
```

Example reverse proxy web caching and SSL offloading for an Internet web server using a static one-to-one virtual IP

This section describes configuring SSL offloading for a reverse proxy web caching configuration using a static one-to-one firewall virtual IP (VIP). While the static one-to-one configuration described in this example is valid, it's also common to change the destination port of the unencrypted HTTPS traffic to a commonly used HTTP port such as 8080 using a port forwarding virtual IP.

Network topology and assumptions

In this configuration, clients on the Internet use HTTP and HTTPS to browse to a web server that is behind a FortiGate unit. A policy added to the FortiGate unit forwards the HTTP traffic to the web server. The policy also offloads HTTPS decryption and encryption from the web server so the web server only sees HTTP traffic.

The FortiGate unit also caches HTTP and HTTPS pages from the web server so when users access cached pages the web server does not see the traffic. Replies to HTTPS sessions are encrypted by the FortiGate unit before returning to the clients.

In this configuration, the FortiGate unit is operating as a web cache in reverse proxy mode. Reverse proxy caches can be placed directly in front of a web server. Web caching on the FortiGate unit reduces the number of requests that the web server must handle, therefore leaving it free to process new requests that it has not serviced before.

Using a reverse proxy configuration:

- avoids the capital expense of additional web servers by increasing the capacity of existing servers
- serves more requests for static content from web servers
- serves more requests for dynamic content from web servers
- reduces operating expenses including the cost of bandwidth required to serve content
- accelerates the response time of web servers and of page download times to end users.

When planning a reverse proxy implementation, the web server's content should be written so that it is "cache aware" to take full advantage of the reverse proxy cache.

In reverse proxy mode, the FortiGate unit functions more like a web server for clients on the Internet. Replicated content is delivered from the proxy cache to the external client without exposing the web server or the private network residing safely behind the firewall.

In this example, the site URL translates to IP address 192.168.10.1, which is the port2 IP address of the FortiGate unit. The port2 interface is connected to the Internet.

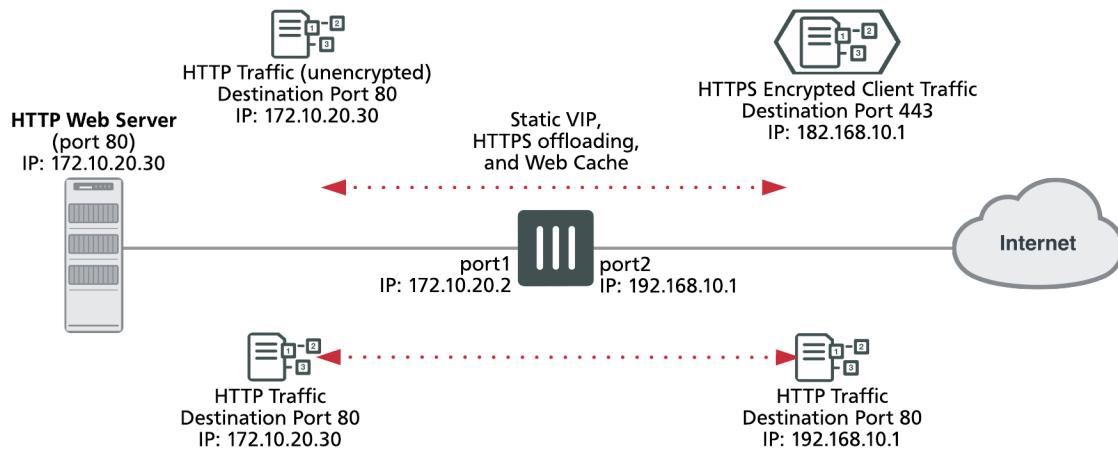
This example assumes that all HTTP traffic uses port 80 and all HTTPS traffic uses port 443.

The FortiGate unit includes the web server CA and an SSL server configuration for IP address 172.10.20.30 and port to 443. The name of the file containing the CA is Rev_Proxy_Cert_1.crt.

The destination address of incoming HTTP and HTTPS sessions is translated to the IP address of the web server using a static one-to-one virtual IP that performs destination address translation (DNAT) for the HTTP packets. The DNAT translates the destination address of the packets from 192.168.10.1 to 172.10.20.30 but does not change the destination port number.

When the SSL server on the FortiGate unit decrypts the HTTPS packets their destination port is changed to port 80.

Reverse proxy web caching and SSL offloading for an Internet web server using static one-to-one virtual IPs



General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Configure the FortiGate unit as a reverse proxy web cache server.
2. Configure the FortiGate unit for SSL offloading of HTTPS traffic.
3. Add an SSL server to offload SSL encryption and decryption for the web server.

Also note that if you perform any additional actions between procedures, your configuration may have different results.

Configuration steps - web-based manager

To configure the FortiGate unit as a reverse proxy web cache server

1. Go to **Policy & Objects > Virtual IPs** and select **Create New** to add a static NAT virtual IP that translates destination IP addresses from 192.168.10.1 to 172.10.20.30 (and does not translate destination ports):

VIP Type	IPv4
Name	Reverse_proxy_VIP
Interface	port2
Type	Static NAT
Optional Filters	Do not select.
External IP Address/Range	192.168.10.1
Mapped IP Address/Range	172.10.20.30
Port Forwarding	Do not select.

2. Select **OK**.
3. Go to **Policy & Objects > IPv4 Policy** and select **Create New** to add a port2 to port1 security policy that accepts HTTP and HTTPS traffic from the Internet.

Do not select security profiles. Set the destination address to the virtual IP. You do not have to enable NAT.

Incoming Interface	port2
Outgoing Interface	port1
Source	all
Destination	Reverse_proxy_VIP
Schedule	always
Service	HTTP HTTPS
Action	ACCEPT

4. Turn on **Web Cache**.
5. Select **OK**.
6. From the CLI enter the following command to add HTTPS web caching to the security policy

Assume the index number of the policy is 5.

```
config firewall policy
  edit 5
    set webcache-https ssl-server
```

end

To configure the FortiGate unit to offload SSL encryption and cache HTTPS content

1. Go to **System > Certificates** and select **Import** to import the web server's CA.

For **Type**, select **Local Certificate**. Select the **Browse** button to locate the file (example file name: Rev_Proxy_Cert_1.crt).

The certificate key size must be 1024 or 2048 bits. 4096-bit keys are not supported.

2. Select **OK** to import the certificate.
3. From the CLI, enter the following command to add the SSL server and to add the server's certificate to the SSL server.

The SSL server **ip** must match the destination address of the SSL traffic after being translated by the virtual IP (172.10.20.30) and the SSL server **port** must match the destination port of the SSL traffic (443). The SSL server operates in half mode since it performs a single-step conversion (HTTPS to HTTP or HTTP to HTTPS).

```
config firewall ssl-server
  edit rev_proxy_server
    set ip 172.10.20.30
    set port 443
    set ssl-mode half
    set ssl-cert Rev_Proxy_Cert_1
end
```

Configuration steps - CLI

To configure the FortiGate unit as a reverse proxy web cache server

1. Enter the following command to add a static NAT virtual IP that translates destination IP addresses from 192.168.10.1 to 172.10.20.30 (and does not translate destination ports):

```
config firewall vip
  edit Reverse_proxy_VIP
    set extintf port2
    set type static-nat
    set extip 192.168.10.1
    set mappedip 172.10.20.30
end
```

2. Enter the following command to add a port2 to port1 security policy that accepts HTTP and HTTPS traffic from the Internet. Enable web caching and HTTPS web caching.

Do not select security profiles. Set the destination address to the virtual IP. You do not have to enable NAT.

```
config firewall policy
  edit 0
    set srcintf port2
    set srcaddr all
    set dstintf port1
    set dstaddr Reverse_proxy_VIP
    set schedule always
    set service HTTP HTTPS
    set action accept
```

```

set webcache enable
set webcache-https ssl-server
end

```

To add an SSL server to offload SSL encryption and decryption for the web server

1. Place a copy of the web server's CA (file name Rev_Proxy_Cert_1.crt) in the root folder of a TFTP server.
2. Enter the following command to import the web server's CA from a TFTP server. The IP address of the TFTP server is 10.31.101.30:

```
execute vpn certificate local import tftp Rev_Proxy_Cert_1.crt 10.31.101.30
```

The certificate key size must be 1024 or 2048 bits. 4096-bit keys are not supported.

3. From the CLI, enter the following command to add the SSL server.

The SSL server `ip` must match the destination address of the SSL traffic after being translated by the virtual IP (172.10.20.30) and the SSL server `port` must match the destination port of the SSL traffic (443). The SSL server operates in half mode since it performs a single-step conversion (HTTPS to HTTP or HTTP to HTTPS).

```

config firewall ssl-server
  edit rev_proxy_server
    set ip 172.10.20.30
    set port 443
    set ssl-mode half
    set ssl-cert Rev_Proxy_Cert_1
  end

```

4. Configure other `ssl-server` settings that you may require for your configuration.

Using a FortiCache as a cache service

Some FortiGate devices don't have sufficient memory or disk space to run a cache service. This feature allows a FortiGate to connect to a FortiCache that has a higher cache capability than most FortiGates.

Syntax:

```

config wanopt remote-storage
  set status {enable|disable}
  set local-cache-id <name ID for connection>
  set remote-cache-id <ID of the remote device>
  set remote-cache-ip <IP address of the remote device>
end

```

Option	Description
status	Enable or disable whether the FortiGate uses a remote caching device as web-cache storage. If disabled, uses local disk(s) as web storage.
local-cache-id	ID that this device uses to connect to the remote caching device

Option	Description
<code>remote-cache-id</code>	ID of the remote caching device that this FortiGate connects to
<code>remote-cache-ip</code>	IP address of the remote caching device that this FortiGate connects to.

WCCP concepts

The Web Cache Communication Protocol (WCCP) can be used to provide web caching with load balancing and fault tolerance. In a WCCP configuration, a WCCP server receives HTTP requests from user's web browsers and redirects the requests to one or more WCCP clients. The clients either return cached content or request new content from the destination web servers before caching it and returning it to the server which in turn returns the content to the original requestor. If a WCCP configuration includes multiple WCCP clients, the WCCP server load balances traffic among the clients and can detect when a client fails and failover sessions to still operating clients. WCCP is described by the [Web Cache Communication Protocol Internet draft](#).

The sessions that are cached by WCCP depend on the configuration of the WCCP clients. If the client is a FortiGate unit, you can configure the port numbers and protocol number of the sessions to be cached. For example, to cache HTTPS traffic on port 443 the WCCP client port must be set to 443 and protocol must be set to 6. If the WCCP client should also cache HTTPS traffic on port 993 the client ports option should include both port 443 and 993.

On a FortiGate unit, WCCP sessions are accepted by a security policy before being cached. If the security policy that accepts sessions that do not match the port and protocol settings in the WCCP clients the traffic is dropped.

WCCP is configured per-VDOM. A single VDOM can operate as a WCCP server or client (not both at the same time). FortiGate units are compatible with third-party WCCP clients and servers. If a FortiGate unit is operating as an Internet firewall for a private network, you can configure it to cache and serve some or all of the web traffic on the private network using WCCP by adding one or more WCCP clients, configuring WCCP server settings on the FortiGate unit and adding WCCP security policies that accept HTTP session from the private network.

FortiGate units support WCCPv1 and WCCPv2. A FortiGate unit in NAT or transparent mode can operate as a WCCP server. To operate as a WCCP client a FortiGate unit must be in NAT mode. FortiGate units communicate between WCCP servers and clients over UDP port 2048. This communication can be encapsulated in a GRE tunnel or just use layer 2 forwarding.



A WCCP server can also be called a WCCP router. A WCCP client can also be called a WCCP cache engine.

WCCP Cisco to FortiGate client using L2-forwarding tunneling

FortiGate supports the option of using Mask mode, in addition to Hash mode, when operating as a WCCP client using L2 forwarding. As a result, you can configure a WCCP FortiGate client to connect to a Cisco Nexus, which doesn't accept the Hash mode assignment method, using the Mask mode assignment method.

WCCP configuration

WCCP configuration overview

To configure WCCP you must create a service group that includes WCCP servers and clients. WCCP servers intercept sessions to be cached (for example, sessions from users browsing the web from a private network). To intercept sessions to be cached the WCCP server must include a security policy that accepts sessions to be cached and WCCP must be enabled in this security policy.

The server must have an interface configured for WCCP communication with WCCP clients. That interface sends and receives encapsulated GRE traffic to and from WCCP clients. The server must also include a WCCP service group that includes a service ID and the addresses of the WCCP clients as well as other WCCP configuration options.

To use a FortiGate unit as a WCCP client, the FortiGate unit must be set to be a WCCP client (or cache engine). You must also configure an interface on the client for WCCP communication. The client sends and receives encapsulated GRE traffic to and from the WCCP server using this interface.

The client must also include a WCCP service group with a service ID that matches a service ID on the server. The client service group also includes the IP address of the servers in the service group and specifies the port numbers and protocol number of the sessions that will be cached on the client.

When the client receives sessions from the server on its WCCP interface, it either returns cached content over the WCCP interface or connects to the destination web servers using the appropriate interface depending on the client routing configuration. Content received from web servers is then cached by the client and returned to the WCCP server over the WCCP link. The server then returns the received content to the initial requesting user web browser.

Finally you may also need to configure routing on the server and client FortiGate units and additional security policies may have to be added to the server to accept sessions not cached by WCCP.

WCCP service groups, service numbers, service IDs and well known services

A FortiGate unit configured as a WCCP server or client can include multiple server or client configurations. Each of these configurations is called a WCCP service group. A service group consists of one or more WCCP servers (or routers) and one or more WCCP clients working together to cache a specific type of traffic. The service group configuration includes information about the type of traffic to be cached, the addresses of the WCCP clients and servers and other information about the service.

A service group is identified with a numeric WCCP service ID (or service number) in the range 0 to 255. All of the servers and clients in the same WCCP service group must have service group configurations with the same WCCP service ID.

The value of the service ID provides some information about the type of traffic to be cached by the service group. Service IDs in the range 0 to 50 are reserved for well known services. A well known service is any service that is defined by the WCCP standard as being well known. Since the service is well known, just the service ID is required to identify the traffic to be cached.

Even though the well known service ID range is 0 to 50, at this time only one well known service has been defined. Its service ID 0, which is used for caching HTTP (web) traffic.

So to configure WCCP to cache HTTP sessions you can add a service group to the WCCP router and WCCP clients with a service ID of 0. No other information about the type of traffic to cache needs to be added to the service group.

Since service IDs 1 to 50 are reserved for well known services and since these services are not defined yet, you should not add service groups with IDs in the range 1 to 50.



FortiOS does allow you to add service groups with IDs between 1 and 50. Since these service groups have not been assigned well known services, however, they will not cache any sessions. Service groups with IDs 51 to 255 allow you to set the port numbers and protocol number of the traffic to be cached. So you can use service groups with IDs 51 to 255 to cache different kinds of traffic based on port numbers and protocol number of the traffic. Service groups 1 to 50; however, do not allow you to set port numbers or protocol numbers so cannot be used to cache any traffic.

To cache traffic other than HTTP traffic you must add service groups with IDs in the range 51 to 255. These service group configurations must include the port numbers and protocol number of the traffic to be cached. It is the port and protocol number configuration in the service group that determines what traffic will be cached by WCCP.

Example WCCP server and client configuration for caching HTTP sessions (service ID = 0)

Enter the following command to add a WCCP service group to a WCCP server that caches HTTP sessions. The IP address of the server is 10.31.101.100 and the WCCP clients are on the 10.31.101.0 subnet. The service ID of this service group is 0.

```
config system wccp
  edit 0
    set router-id 10.31.101.100
    set server-list 10.31.101.0 255.255.255.0
  end
```

Enter the following commands to configure a FortiGate unit to operate as a WCCP client and add a service group that configures the client to cache HTTP sessions. The IP address of the server is 10.31.101.100 and IP address of this WCCP clients is 10.31.101.1 subnet. The service ID of this service group is 0.

```
config system settings
  set wccp-cache-engine enable
end

config system wccp
  edit 0
    set cache-id 10.31.101.1
    set router-list 10.31.101.100
  end
```



You cannot enter the `wccp-cache-engine enable` command if you have already added a WCCP service group. When you enter this command an interface named `w.<vdom_name>` is added to the FortiGate configuration (for example `w.root`). All traffic redirected from a WCCP router is considered to be received at this interface of the FortiGate unit operating as a WCCP client. A default route to this interface with lowest priority is added.

Example WCCP server and client configuration for caching HTTPS sessions

Enter the following command to add a service group to a WCCP server that caches HTTPS content on port 443 and protocol 6. The IP address of the server is 10.31.101.100 and the WCCP clients are on the 10.31.101.0 subnet. The service ID of this service group is 80.

```
config system settings
    set wccp-cache-engine enable
end

config system wccp
    edit 80
        set router-id 10.31.101.100
        set server-list 10.31.101.0 255.255.255.0
        set ports 443
        set protocol 6
end
```

Enter the following commands to configure a FortiGate unit to operate as a WCCP client and add a service group that configures client to cache HTTPS sessions on port 443 and protocol 6. The IP address of the server is 10.31.101.100 and IP address of this WCCP clients is 10.31.101.1 subnet. The service ID of this service group must be 80 to match the service ID added to the server.

```
config system settings
    set wccp-cache-engine enable
end

config system wccp
    edit 80
        set cache-id 10.31.101.1
        set router-list 10.31.101.100
        set ports 443
        set protocol 6
end
```

Example WCCP server and client configuration for caching HTTP and HTTPS sessions

You could do this by configuring two WCCP service groups as described in the previous examples. Or you could use the following commands to configure one service group for both types of traffic. The example also caches HTTP sessions on port 8080.

Enter the following command to add a service group to a WCCP server that caches HTTP sessions on ports 80 and 8080 and HTTPS sessions on port 443. Both of these protocols use protocol number 6. The IP address of the server is 10.31.101.100 and the WCCP clients are on the 10.31.101.0 subnet. The service ID of this service group is 90.

```
config system wccp
    edit 90
```

```
set router-id 10.31.101.100
set server-list 10.31.101.0 255.255.255.0
set ports 443 80 8080
set protocol 6
end
```

Enter the following commands to configure a FortiGate unit to operate as a WCCP client and add a service group that configures client to cache HTTP sessions on port 80 and 8080 and HTTPS sessions on port 443. The IP address of the server is 10.31.101.100 and IP address of this WCCP clients is 10.31.101.1 subnet. The service ID of this service group must be 90 to match the service ID added to the server.

```
config system settings
    set wccp-cache-engine enable
end
config system wccp
    edit 90
        set cache-id 10.31.101.1
        set router-list 10.31.101.100
        set ports 443 80 8080
        set protocol 6
    end
end
```

Other WCCP service group options

In addition to using WCCP service groups to define the types of traffic to be cached by WCCP the following options are available for servers and clients.

Server configuration options

The server configuration must include the `router-id`, which is the WCCP server IP address. This is the IP address of the interface that the server uses to communicate with WCCP clients.

The `group-address` is used for multicast WCCP configurations to specify the multicast addresses of the clients.

The `server-list` defines the IP addresses of the WCCP clients that the server can connect to. Often the server list can be the address of the subnet that contains the WCCP clients.

The `authentication` option enables or disables authentication for the WCCP service group. Authentication must be enabled on all servers and clients in a service group and members of the group must have the same password.

The `forward-method` option specifies the protocol used for communication between the server and clients. The default forwarding method is GRE encapsulation. If required by your network you can also select to use unencapsulated layer-2 packets instead of GRE or select `any` to allow both. The `return-method` allows you to specify the communication method from the client to the server. Both GRE and layer-2 are supported.

The `assignment-method` determines how the server load balances sessions to the clients if there are multiple clients. Load balancing can be done using hashing or masking.

Client configuration options

The client configuration includes the `cache-id` which is the IP address of the FortiGate interface of the client that communicates with WCCP server. The `router-list` option is the list of IP addresses of the WCCP servers in the WCCP service group.

The `ports` option lists the port numbers of the sessions to be cached by the client and the `protocol` sets the protocol number of the sessions to be cached. For TCP sessions the protocol is 6.

The `service-type` option can be `auto`, `dynamic` or `standard`. Usually you would not change this setting.

The client configuration also includes options to influence load balancing including the `primary-hash`, `priority`, `assignment-weight` and `assignment-bucket-format`.

Example caching HTTP sessions on port 80 using WCCP

In this example configuration (shown below), a FortiGate unit with host name `WCCP_srv` is operating as an Internet firewall for a private network is also configured as a WCCP server. The `port1` interface of `WCCP_srv` is connected to the Internet and the `port2` interface is connected to the internal network.

All HTTP traffic on port 80 that is received at the `port2` interface of `WCCP_srv` is accepted by a `port2` to `port1` security policy with WCCP enabled. All other traffic received at the `port2` interface is allowed to connect to the Internet by adding a general `port2` to `port1` security policy below the HTTP on port 80 security policy.

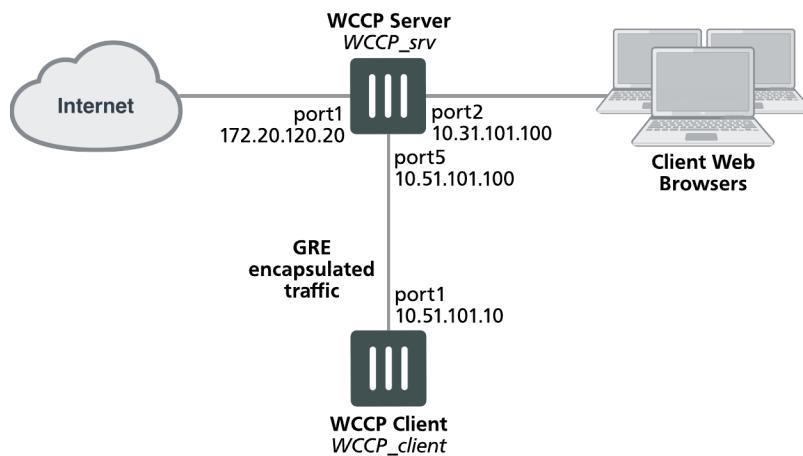
A WCCP service group is added to `WCCP_srv` with a service ID of 0 for caching HTTP traffic on port 80. The `port5` interface of `WCCP_srv` is configured for WCCP communication.

A second FortiGate unit with host name `WCCP_client` is operating as a WCCP client. The `port1` interface of `WCCP_client` is connected to `port5` of `WCCP_srv` and is configured for WCCP communication.

`WCCP_client` is configured to cache HTTP traffic because it also has a WCCP service group with a service ID of 0.

`WCCP_client` connects to the Internet through `WCCP_srv`. To allow this, a `port5` to `port1` security policy is added to `WCCP_srv`.

FortiGate WCCP server and client configuration



Configuring the WCCP server (WCCP_srv)

Use the following steps to configure `WCCP_srv` as the WCCP server for the example network. The example steps only describe the WCCP-related configuration.

To configure WCCP_srv as a WCCP server

1. Add a port2 to port1 security policy that accepts HTTP traffic on port 80 and is configured for WCCP:

```
config firewall policy
  edit 0
    set srcintf port2
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service HTTP
    set wccp enable
    set nat enable
  end
```

2. Add another port2 to port1 security policy to allow all other traffic to connect to the Internet.

```
config firewall policy
  edit 0
    set srcintf port2
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set nat enable
  end
```

3. Move this policy below the WCCP policy in the port2 to port1 policy list.

4. Enable WCCP on the port5 interface.

```
config system interface
  edit port5
    set wccp enable
  end
```

5. Add a WCCP service group with service ID 0.

```
config system wccp
  edit 0
    set router-id 10.51.101.100
    set server-list 10.51.101.0 255.255.255.0
  end
```

6. Add a firewall address and security policy to allow the WCCP_client to connect to the internet.

```
config firewall address
  edit WCCP_client_addr
    set subnet 10.51.101.10
  end
config firewall policy
  edit 0
    set srcintf port5
    set dstintf port1
    set srcaddr WCCP_client_addr
    set dstaddr all
    set action accept
```

```
set schedule always
set service ALL
set nat enable
end
```

Configuring the WCCP client (WCCP_client)

Use the following steps to configure WCCP_client as the WCCP client for the example network. The example steps only describe the WCCP-related configuration.

To configure WCCP_client as a WCCP client

1. Configure WCCP_client to operate as a WCCP client.

```
config system settings
  set wccp-cache-engine enable
end
```



You cannot enter the `wccp-cache-engine enable` command if you have already added a WCCP service group. When you enter this command an interface named `w.<vdom_name>` is added to the FortiGate configuration (for example `w.root`). All traffic redirected from a WCCP router is considered to be received at this interface of the FortiGate unit operating as a WCCP client. A default route to this interface with lowest priority is added.

2. Enable WCCP on the port1 interface.

```
config system interface
  edit port1
    set wccp enable
end
```

3. Add a WCCP service group with service ID 0.

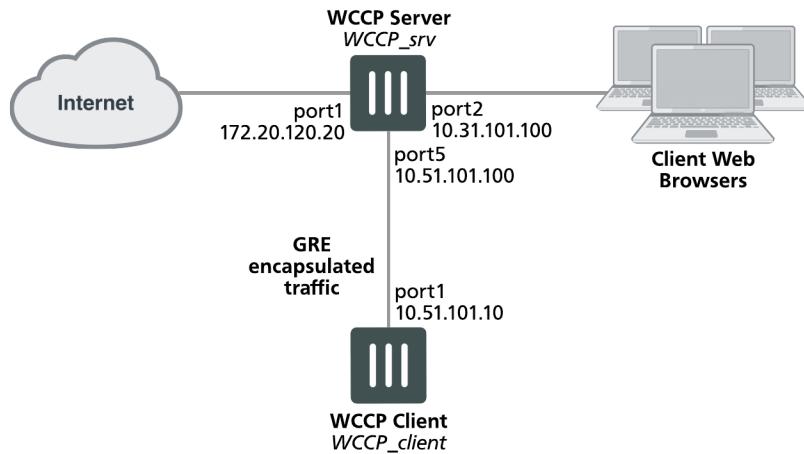
```
config system wccp
  edit 0
    set cache-id 10.51.101.10
    set router-list 10.51.101.100
end
```

Example caching HTTP sessions on port 80 and HTTPS sessions on port 443 using WCCP

This example configuration is the same as that described in [Example caching HTTP sessions on port 80 and HTTPS sessions on port 443 using WCCP on page 351](#) except that WCCP now also cached HTTPS traffic on port 443. To cache HTTP and HTTPS traffic the WCCP service group must have a service ID in the range 51 to 255 and you must specify port 80 and 443 and protocol 6 in the service group configuration of the WCCP client.

Also the security policy on the WCCP_srv that accepts sessions from the internal network to be cached must accept HTTP and HTTPS sessions.

FortiGate WCCP server and client configuration



Configuring the WCCP server (WCCP_srv)

Use the following steps to configure WCCP_srv as the WCCP server for the example network. The example steps only describe the WCCP-related configuration.

To configure WCCP_srv as a WCCP server

1. Add a port2 to port1 security policy that accepts HTTP traffic on port 80 and HTTPS traffic on port 443 and is configured for WCCP:

```
config firewall policy
  edit 0
    set srtintf port2
    set dstintf port1
    set srcaddr all
    set dstaddr all
    set action accept
    set schedule always
    set service HTTP HTTPS
    set wccp enable
    set nat enable
  end
```

2. Add another port2 to port1 security policy to allow all other traffic to connect to the Internet.

```
config firewall policy
    edit 0
        set srtintf port2
        set dstintf port1
        set srcaddr all
        set dstaddr all
        set action accept
        set schedule always
        set service ALL

    set nat enable
end
```

3. Move this policy below the WCCP policy in the port2 to port1 policy list.
4. Enable WCCP on the port5 interface.

```
config system interface
  edit port5
    set wccp enable
  end
```

5. Add a WCCP service group with service ID 90 (can be any number between 51 and 255).

```
config system wccp
  edit 90
    set router-id 10.51.101.100
    set server-list 10.51.101.0 255.255.255.0
  end
```

6. Add a firewall address and security policy to allow the WCCP_client to connect to the internet.

```
config firewall address
  edit WCCP_client_addr
    set subnet 10.51.101.10
  end
config firewall policy
  edit 0
    set srcintf port5
    set dstintf port1
    set srcaddr WCCP_client_addr
    set dstaddr all
    set action accept
    set schedule always
    set service ALL
    set nat enable
  end
```

Configuring the WCCP client (WCCP_client)

Use the following steps to configure WCCP_client as the WCCP client for the example network. The example steps only describe the WCCP-related configuration.

To configure WCCP_client as a WCCP client

1. Configure WCCP_client to operate as a WCCP client.

```
config system settings
  set wccp-cache-engine enable
end
```



You cannot enter the `wccp-cache-engine enable` command if you have already added a WCCP service group. When you enter this command an interface named `w.<vdom_name>` is added to the FortiGate configuration (for example `w.root`). All traffic redirected from a WCCP router is considered to be received at this interface of the FortiGate unit operating as a WCCP client. A default route to this interface with lowest priority is added.

2. Enable WCCP on the port1 interface.

```
config system interface
  edit port1
```

```
    set wccp enable
end
```

3. Add a WCCP service group with service ID 90. This service group also specifies to cache sessions on ports 80 and 443 (for HTTP and HTTPS) and protocol number 6.

```
config system wccp
edit 90
    set cache-id 10.51.101.10
    set router-list 10.51.101.100
    ports 80 443
    set protocol 6
end
```

WCCP packet flow

The following packet flow sequence assumes you have configured a FortiGate unit to be a WCCP server and one or more FortiGate units to be WCCP clients.

1. A user's web browser sends a request for web content.
2. The FortiGate unit configured as a WCCP server includes a security policy that intercepts the request and forwards it to a WCCP client.

The security policy can apply UTM features to traffic accepted by the policy.
3. The WCCP client receives the WCCP session.
4. The client either returns requested content to the WCCP server if it is already cached, or connects to the destination web server, receives and caches the content and then returns it to the WCCP server.
5. The WCCP server returns the requested content to the user's web browser.
6. The WCCP router returns the request to the client web browser.

The client web browser is not aware that all this is taking place and does not have to be configured to use a web proxy.

Configuring the forward and return methods and adding authentication

The WCCP forwarding method determines how intercepted traffic is transmitted from the WCCP router to the WCCP cache engine. There are two different forwarding methods:

- GRE forwarding (the default) encapsulates the intercepted packet in an IP GRE header with a source IP address of the WCCP router and a destination IP address of the target WCCP cache engine. The result is a tunnel that allows the WCCP router to be multiple hops away from the WCCP cache server.
- L2 forwarding rewrites the destination MAC address of the intercepted packet to match the MAC address of the target WCCP cache engine. L2 forwarding requires that the WCCP router is Layer 2 adjacent to the WCCP client.

You can use the following command on a FortiGate unit configured as a WCCP router to change the forward and return methods to L2:

```
config system wccp
edit 1
    set forward-method L2
```

```
    set return-method L2
end
```

You can also set the forward and return methods to any in order to match the cache server configuration.

By default the WCCP communication between the router and cache servers is unencrypted. If you are concerned about attackers sniffing the information in the WCCP stream you can use the following command to enable hash-based authentication of the WCCP traffic. You must enable authentication on the router and the cache engines and all must have the same password.

```
config system wccp
edit 1
    set authentication enable
    set password <password>
end
```

WCCP messages

When the WCCP service is active on a web cache server it periodically sends a WCCP HERE I AM broadcast or unicast message to the FortiGate unit operating as a WCCP router. This message contains the following information:

- Web cache identity (the IP address of the web cache server).
- Service info (the service group to join).

If the information received in the previous message matches what is expected, the FortiGate unit replies with a WCCP I SEE YOU message that contains the following details:

- Router identity (the FortiGate unit's IP address).
- Sent to IP (the web cache IP addresses to which the packets are addressed)

When both ends receive these two messages the connection is established, the service group is formed and the designated web cache is elected.

Troubleshooting WCCP

Two types of debug commands are available for debugging or troubleshooting a WCCP connection between a FortiGate unit operating as a WCCP router and its WCCP cache engines.

Real time debugging

The following commands can capture live WCCP messages:

```
diag debug en
diag debug application wccpd <debug level>
```

Application debugging

The following commands display information about WCCP operations:

```
get test wccpd <integer>
diag test application wccpd <integer>
```

Where <integer> is a value between 1 and 6:

1. Display WCCP stats
2. Display WCCP config
3. Display WCCP cache servers
4. Display WCCP services
5. Display WCCP assignment
6. Display WCCP cache status

Enter the following command to view debugging output:

```
diag test application wccpd 3
```

Sample output from a successful WCCP connection:

```
service-0 in vdom-root: num=1, usable=1
cache server ID:
len=44, addr=172.16.78.8, weight=4135, status=0
rcv_id=6547, usable=1, fm=1, nq=0, dev=3 (k3),
to=192.168.11.55
ch_no=0, num_router=1:
192.168.11.55
```

Sample output from the same command from an unsuccessful WCCP connection (because of a service group password mismatch):

```
service-0 in vdom-root: num=0, usable=0
diag debug application wccpd -1
Sample output:
wccp_on_recv() -98: vdom-root recv: num=160, dev=3 (3),
172.16.78.8->192.168.11.55
wccp2_receive_pkt() -1124: len=160, type=10, ver=0200,
length=152
wccp2_receive_pkt() -1150: found component:t=0, len=20
wccp2_receive_pkt() -1150: found component:t=1, len=24
wccp2_receive_pkt() -1150: found component:t=3, len=44
wccp2_receive_pkt() -1150: found component:t=5, len=20
wccp2_receive_pkt() -1150: found component:t=8, len=24
wccp2_check_security_info() -326: MD5 check failed
```

Web proxy concepts

These are concepts that apply to both Transparent and Explicit Proxy.

Proxy policy

Information on Proxy policy options can be found at [Proxy option components on page 50](#)

Configuration information can be found at [Web proxy configuration on page 365](#)

Proxy authentication

Beginning in FortiOS 5.6, authentication is separated from authorization for user based policy. You can add authentication to proxy policies to control access to the policy and to identify users and apply different UTM features to different users. The described authentication methodology works with **Explicit Web Proxy** and **Transparent Proxy**.

Authentication of web proxy sessions uses HTTP basic and digest authentication as described in [RFC 2617 \(HTTP Authentication: Basic and Digest Access Authentication\)](#) and prompts the user for credentials from the browser allowing individual users to be identified by their web browser instead of IP address. HTTP authentication allows the FortiGate unit to distinguish between multiple users accessing services from a shared IP address.

The methodology of adding authentication has changed from FortiOS version 5.4 and previous version. Split-policy has been obsoleted and instead of identity-based-policy, authentication is managed by authentication scheme, setting and rule settings. These authentication settings are no longer configured with the individual policies. Authentication is set up in the contexts of:

```
config authentication scheme
config authentication setting
config authentication rule
```

The Authentication rule table defines how to identify user-ID. It uses the match factors:

- Protocol
- Source Address

For one address and protocol, there is only one authentication rule. It is possible to configure multiple authentication methods for one address. The client browser will choose one authentication method from the authentication methods list, but you can not control which authentication method will be chosen by the browser.

Matching

If a rule is matched, the authentication methods defined in the rule will be used to authenticate a user. The procedure works as the following:

1. If it is IP-based, look up active user list to see if a user existed from the source IP. If found, return the user ID.
2. If no method is set, an anonymous user is created to associate to the source-IP. Return the anonymous user. It is another way to bypass user authentication for some source IPs.
3. Use authentication methods to authenticate the user.
 - If no active method is defined, a failure will result to return an anonymous user.
 - Otherwise, a valid or guest user has to be identified to move on.
 - Return the identified user ID.

Once a user is returned, the policy match resumes until a policy is matched or default policy will be used.

Processing policies for authentication

Authentication rules are checked once a User-ID is needed in order to resolve a match to a policy

Use the following scenario as an example of the process.

There are 3 policies:

- `policy1` does not have an associated user group
- `policy2` has an associated user group
- `policy3` does not have an associated user group

Step 1

If the traffic, based on protocol and source address matches `policy 1`, no user authentication is needed. The traffic is processed by `policy1`.

Step 2

If the traffic does not match `policy 1`, and any factor of `policy 2` is not matched, continue to next policy.

If all the factors except the user-group of `policy 2` are matched the authentication rule table is checked to get user-ID in the process in based on the procedure described earlier in Matching.

Step 3

When a user-ID is returned, whether it is a valid user or anonymous user, it is checked to see if the user is authorized by the user group associated with `policy2`. If yes, it is a match of `policy2`, and the traffic is processed by `policy2`. If not move on the next policy.

Step 4

For the purposes of the scenario, it will be assumed that the traffic either matches `policy3` or that `policy3` is the final policy that denies everything.

CLI syntax

Removals:

- "split-policy" from `firewall explicit-proxy-policy`.

The previous method to set up a split policy was:

```
config firewall explicit-proxy-policy
```

```
edit 1
  set proxy web
  set identity-based enable
  set groups <User group>
  config identity-based-policy
    edit 1
      set schedule "always"
      set utm-status enable
      set users "guest"
      set profile-protocol-options "default"
      next
    end
  next
end
```

- "auth relative" from firewall explicit-proxy-policy

The following attributes have been removed from firewall explicit-proxy-policy:

- identity-based
- ip-based
- active-auth-method
- sso-auth-method
- require-tfa

Moves:

users and groups from

```
firewall explicit-proxy-policy identity-based-policy
to

config firewall proxy-policy
  edit 1
    set groups <Group name>
    set users <User name>
  end
```

Additions:

authentication scheme

```
config authentication scheme
  edit <name>
    set method [ntlm|basic|digest|form|negotiate|fssso|rssso|none]
```

- **ntlm** - NTLM authentication.
- **basic** - Basic HTTP authentication.
- **digest** - Digest HTTP authentication.
- **form** - Form-based HTTP authentication.
- **negotiate** - Negotiate authentication.
- **fssso** - FSSO authentication.
- **rssso** - RADIUS Single Sign-On authentication.
- **none** - No authentication.

authentication setting

```
config authentication setting
    set active-auth-scheme <string>
    set sso-auth-scheme <string>
    set captive-portal <string>
    set captive-portal-port <integer value from 1 to 65535>
```

- **active-auth-scheme** - Active authentication method.
- **sso-auth-scheme** - SSO authentication method.
- **captive-portal** - Captive portal host name.
- **captive-portal-port** - Captive portal port number.

authentication rule

```
config authentication rule
    edit <name of rule>
        set status [enable|disable]
        set protocol [http|ftp|socks]
        set srcaddr <name of address object>
        set srcaddr6 <name of address object>
        set ip-based [enable|disable]
        set active-auth-method <string>
        set sso-auth-method <string>
        set web-auth-cookie [enable|disable]
        set transaction-based [enable|disable]
        set comments
```

- **status** - Enable/disable auth rule status.
- **protocol** - set protocols to be matched
- **srcaddr /srcaddr6** - Source address name. [srcaddr or srcaddr6(web proxy only) must be set].
- **ip-based** - Enable/disable IP-based authentication.
- **active-auth-method** - Active authentication method.
- **sso-auth-method** - SSO authentication method (require ip-based enabled)
- **web-auth-cookie** - Enable/disable Web authentication cookie.
- **transaction-based** - Enable/disable transaction based authentication.
- **comments** - Comment.

Configuring authentication in transparent proxy

You can enable transparent web-proxy feature to support authentication. Follow these steps

1. Configure a firewall policy
2. Enable a UTM profile in the firewall policy. Whenever there is a UTM item enabled, the feature enables the **profile-protocol-options**.
3. Go to the **Proxy Options** profile.
 - In the GUI this is **Security Profiles > Proxy Options**.
 - In the CLI it is `config firewall profile-protocol-options`.
 Edit the profile used by the policy.
4. Enable HTTP in the profile.

In the GUI toggle on **HTTP** under **Protocol Port Mapping**

In the CLI, the command sequence is:

```
config firewall profile-protocol-options
  edit <profile id>
    config http
      set status enable
    end
```

Fill out any other appropriate values.

5. Configure the proxy-policy, and set the value transparent-web for proxy option, others configuration are same as the explicit-web proxy

In the GUI, go to **Policy & Objects > Proxy Policy**. In the **Proxy Type** field choose **Transparent Web**.

In the CLI, the command sequence is:

```
config firewall proxy-policy
  edit <profile id>
    set proxy transparent-web
  end
```

Fill out any other appropriate values.

6. Setup the authentication rule and scheme

With this configuration, if a HTTP request passes through FortiGate without explicit web proxy being applied, the traffic will be redirected to WAD daemon after it matches the proxy with HTTP-policy enabled, then WAD will do the proxy-policy matching, and all of the proxy authentication method can be used for the request.

Proxy addresses

Information on Proxy addresses can be found at [Proxy addresses on page 229](#)

Proxy address group

In the same way that IPv4 and IPv6 addresses can only be grouped together, Proxy addresses can only be grouped with other Proxy addresses. Unlike the other address groups, the Proxy address groups are further divided into source address groups and destination address groups. To see the configuration steps go to [Proxy address groups on page 231](#)

Web proxy firewall services and service groups

Configure web proxy services by selecting **Explicit Proxy** when configuring a service. Web proxy services can be selected in a explicit web proxy policy when adding one from the CLI. If you add a policy from the web-based manager the service is set to the **webproxy** service. The webproxy service should be used in most cases, it matches with any traffic with any port number. However, if you have special requirements, such as using a custom protocol type or a reduced port range or need to add an IP/FQDN to an proxy service you can create custom explicit web proxy services.

Web proxy services are similar to standard firewall services. You can configure web proxy services to define one or more protocols and port numbers that are associated with each web proxy service. Web proxy services can also be grouped into web proxy service groups.

One way in which web proxy services differ from firewall services is the protocol type you can select. The following protocol types are available:

- ALL
- CONNECT
- FTP
- HTTP
- SOCKS-TCP
- SOCKS-UDP

To add a web proxy service go to **Policy & Objects > Services** and select **Create New**. Set **Service Type** to **Explicit Proxy** and configure the service as required.

To add a web proxy service from the CLI enter:

```
config firewall service custom
  edit my-socks-service
    set explicit-proxy enable
    set category Web Proxy
    set protocol SOCKS-TCP
    set tcp-portrange 3450-3490
  end
```

To add a web proxy service group go to **Policy & Objects > Services** and select **Create New > Service Group**. Set **Type** to **Explicit Proxy** and add web proxy services to the group as required.

To add a web proxy service group from the CLI enter:

```
config firewall service group
  edit web-group
    set explicit-proxy enable
    set member webproxy my-socks-service
  end
```

Learn client IP

If there is another NATing device between the FortiGate and the Client (browser), this feature can be used to identify the real client in spite of the address translation. Knowing the actual client is imperative in cases where authorization is taking place.

The settings for the feature are in the CLI in the context of

```
config web-proxy global
```

Once here, enable the feature with the command:

```
  set learn-client-ip enable
```

Once the feature is enabled, the other settings become available.

```
  learn-client-ip-from-header
```

This command has the following options:

true-client-ip	Support HTTP header True-Client-IP.
x-real-ip	Support HTTP header X-Real-IP.
x-forwarded-for	Support HTTP header X-Forwarded-For.

learn-client-ip-srcaddr/learn-client-ip-srcaddr6

The options for this setting are selected from the list of IPv4 address or IPv6 address objects.

Example

Below is a config example where the real client ip address will be used to match policy or fssso authentication after the learn-client-ip feature enabled.

The value of `learn-client-ip-from-header` option can be set to `true-client-ip`, `x-real-ip` or `x-forwarded-for`, but in this case it has been set to `x-forwarded-for`.

```
config web-proxy global
    set proxy-fqdn "default.fqdn"
    set webproxy-profile "default"
    set learn-client-ip enable
    set learn-client-ip-from-header x-forwarded-for
    set learn-client-ip-srcaddr "all"
    end

config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "mgmt1"
        set srcaddr "all"
        set dstaddr "all"
        set service "w"
        set action accept
        set schedule "always"
        set groups "fssol1"
        set utm-status enable
        set av-profile "default"
        set dlp-sensor "default"
        set profile-protocol-options "default"
        set ssl-ssh-profile "deep-inspection"
        end

    config authentication rule
        edit "rule1"
            set srcaddr "all"
            set sso-auth-method "scheme1"
            end

    config authentication scheme
        edit "scheme1"
            set method fssso
```

end

Web proxy configuration

General web proxy configuration steps

You can use the following general steps to configure the explicit web proxy.

To enable the explicit web proxy - web-based manager:

1. Go to **Network > Explicit Proxy** and enable **Explicit Web Proxy**. From here you can optionally change the HTTP port that the proxy listens on (the default is 8080) and optionally specify different ports for HTTPS, FTP, PAC, and other options.
2. Optionally enable **IPv6 Explicit Proxy** to turn on the explicit web proxy for IPv6 traffic.



If you enable both the IPv4 and the IPv6 explicit web proxy you can combine IPv4 and IPv6 addresses in a single explicit web proxy policy to allow both IPv4 and IPv6 traffic through the proxy.

3. Select **Apply**.
4. Go to **Network > Interfaces** and select one or more interfaces for which to enable the explicit web proxy. Edit the interface. Under the **Miscellaneous** heading select **Enable Explicit Web Proxy**.



Enabling the explicit web proxy on an interface connected to the Internet is a security risk because anyone on the Internet who finds the proxy could use it to hide their source address. If you enable the proxy on such an interface make sure authentication is required to use the proxy.

5. Go to **Policy & Objects > Addresses** and select **Create New** to add a firewall address that matches the source address of packets to be accepted by the explicit proxy.

Category	Address
Name	Internal_subnet
Type	IP Range
Subnet / IP Range	10.31.101.1 - 10.31.101.255
Interface	any*

*The **Interface** must be set to **Any**.

You can also set the **Type** to **URL Pattern (Explicit Proxy)** to add a destination URL that is only used by the explicit proxy. For example, to create an explicit policy that only allows access to Fortinet.com:

Category	Address
Name	Fortinet-web-sites
Type	URL Pattern (Explicit Proxy)
URL Pattern	fortinet.com
Interface	any

6. Go to **Policy & Objects > Proxy Policy** and select **Create New**. Configure the policy as required to accept the traffic that you want to be allowed to use the explicit web proxy.
7. Set the **Outgoing Interface** parameter by selecting the field with the "+" next to the field label. Selecting the field will slide out a window from the right where you can select from the available interfaces. You can select one or more specific interfaces. For more information on interfaces, check the Concepts section called [Interfaces and Zones](#).
8. The **Source** of the policy must match the client's source IP addresses. The interface of this firewall address must be set to **any**.
9. The **Destination** field should match the addresses of web sites that clients are connecting to. Usually the destination address would be **all** if proxying Internet web browsing. You could also specify a URL firewall address to limit the policy to allowing access to this URL.
10. Set the **Schedule** parameter by using the drop down menu to select a preconfigured schedule. The "+" icon next to the **Search** field is a shortcut for creating a new schedule object. For more information on addresses, check the Firewall Objects section called [Firewall schedules](#)
11. If **Default Firewall Policy Action** is set to **Deny** (under **Network > Explicit Proxy**), traffic sent to the explicit web proxy that is not accepted by a web-proxy policy is dropped. If **Default Firewall Policy Action** is set to **Allow** then all web-proxy sessions that don't match with a security policy are allowed.

For example, the following security policy allows users on an internal network to access fortinet.com websites through the wan1 interface of a FortiGate unit.

Explicit Proxy Type	Web
Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	Fortinet-web-sites
Schedule	always
Action	ACCEPT



The explicit web-proxy accepts VIP addresses for destination addresses. If an external IP matches a VIP policy, the IP is changed to the mapped-IP of the VIP.

12. Set the Disclaimer Options

You can configure a disclaimer for each Authentication Rule by enabling one of the options here. The

choices are:

Disable	No disclaimer (default setting)
By Domain	The disclaimer will be displayed on different domains. The explicit web proxy will check the referring header to mitigate the javascript/css/images/video/etc page.
By Policy	The disclaimer will be displayed if the HTTP request matches a different explicit firewall policy.
By User	The disclaimer will be displayed when a new user logs on.

If you chose a disclaimer option other than **Disable**, you will have the option to enable **Customize Messages**. If enabled, select the **Edit Disclaimer Message** button to customize the message to your needs. This can be done as text or as HTML. The default HTML version is there if you just want to make minor changes.

13. Enable **Security Profiles** as required. Once the profile type is toggled to enabled, you can use the drop down menu to select a specific profile. The available profile types are:

- **AntiVirus**
- **WebFilter**
- **Application Control**
- **IPS**
- **DLP Sensor**
- **ICAP**
- **Web Application Firewall**

Just like with a regular policy, as soon as any of the **Security Profiles** is enabled, the following fields, with their own drop down menus for specific profiles will appear:

- **Proxy Options**
- **SSL/SSH Inspection**

14. Select **OK**.

To enable the explicit web proxy - CLI:

1. Enter the following command to turn on the IPv4 and IPv6 explicit web proxy for HTTP and HTTPS traffic.

```
config web-proxy explicit
    set status enable
    set ipv6-status enable
end
```

You can also enter the following command to enable the web proxy for FTP sessions in a web browser.

```
config web-proxy explicit
    set ftp-over-http enable
end
```

The default explicit web proxy configuration has `sec-default-action set to deny` and requires

you to add a security policy to allow access to the explicit web proxy.

2. Enter the following command to enable the explicit web proxy for the internal interface.

```
config system interface
  edit internal
    set explicit-web-proxy enable
  end
end
```

3. Use the following command to add a firewall address that matches the source address of users who connect to the explicit web proxy.

```
config firewall address
  edit Internal_subnet
    set type iprange
    set start-ip 10.31.101.1
    set end-ip 10.31.101.255
  end
```

The source address for a web-proxy security policy cannot be assigned to a FortiGate interface.

4. Optionally use the following command to add a destination URL that is only used by the explicit proxy. For example, to create an explicit policy that only allows access to Fortinet.com:

```
config firewall address
  edit Fortinet-web-sites
    set type url
    set url fortinet.com
  end
```

5. Use the following command to add an explicit web proxy policy that allows all users on the internal subnet to use the explicit web proxy for connections through the wan1 interface to the Internet.

```
config firewall proxy-policy
  edit 0
    set proxy explicit-web
    set dstintf wan1
    set scraddr Internal_subnet
    set dstaddr all
    set action accept
    set service webproxy
    set schedule always
  end
```

6. Use the following command to add an explicit web proxy policy that allows authenticated users on the internal subnet to use the explicit web proxy for connections through the wan1 interface to the Internet.

```
config firewall proxy-policy
  edit 0
    set proxy explicit-web
    set dstintf wan1
    set scraddr Internal_subnet
    set dstaddr Fortinet-web-sites
    set action accept
    set service webproxy
    set schedule always
    set groups <User group>
  end
end
```

7. Use the following command to change global web proxy settings, for example to set the maximum request length for the explicit web proxy to 10:

```
config web-proxy global
    set max-request-length 10
end
```

8. Determine whether or not to use Botnet feature.

The option `scan-botnet-connections` uses the following syntax:

```
config firewall proxy-policy
    edit <policy id>
        set scan-botnet-connections [disable|block|monitor]
    end
```

Where:

- `disable` means do not scan connections to botnet servers
- `block` means block connection to botnet servers
- `monitor` means log connections to botnet servers

Logging options in web proxy profiles

There is an option on what action to take regarding the authenticated user's name in the header information for reading by upstream proxies and systems. This option can be used when a FortiGate is operating as an explicit proxy and authenticating users. The header is the `x-authenticated-user` and is used by the upstream proxy to ensure correct policy enforcement and to log the user's activity.

The `log-header-change` option enables the logging of any header changes in the web-proxy profile, including changes to authenticated users or groups.

Syntax

```
config web-proxy profile
    edit <profile ID#>
        set header-x-authenticated-user {pass|add|remove}
        set header-x-authenticated-groups {pass|add|remove}
        set log-header-change {enable|disable}
    end
```

Option	Description
<code>header-x-authenticated-user</code>	Action to take on the HTTP <code>x-authenticated-user</code> header in forwarded requests: <ul style="list-style-type: none"> • <code>pass</code> - Forward the same HTTP header • <code>add</code> - Add the HTTP header • <code>remove</code> - Remove the HTTP header

Option	Description
header-x-authenticated-groups	Action to take on the HTTP <code>x-authenticated-groups</code> header in forwarded requests: <ul style="list-style-type: none"> • <code>pass</code> - Forward the same HTTP header • <code>add</code> - Add the HTTP header • <code>remove</code> - Remove the HTTP header
log-header-change	enable or disable the logging of HTTP header changes

Policy matching based on referrer headers and query strings

Web proxy policies support creating web proxy addresses to match referrer headers and query strings.

Matching referrer headers

For example, to create a web proxy address to match the referrer header to block access to the following YouTube URL `http://youtube.com/user/test321`. The http request will have the following format:

```
GET /user/test321 HTTP/1.1
Host: www.youtube.com
User-Agent: curl/7.52.1
Accept: */*
```

Create the following web proxy addresses to match this page:

```
config firewall proxy-address
  edit youtube
    set type host-regex
    set host-regex ".*youtube.com"
  next
  edit test321
    set host "youtube"
    set path "/user/test321"
    set referrer enable
  end
```

Then create two proxy policies, one that allows access to all traffic and a second one that blocks access to the page that matches the referrer header:

```
config firewall proxy-policy
  edit 1
    set uuid 92273e4e-8c53-51e7-a7bd-f26e6e15fc98
    set proxy explicit-web
    set dstintf "wan2"
    set srcaddr "all"
    set dstaddr "all"
    set service "webproxy-connect"
    set action accept
    set schedule "always"
    set utm-status enable
    set profile-protocol-options "test"
    set ssl-ssh-profile "test"
  next
  edit 2
```

```

set uuid d35ad06a-8c53-51e7-8511-17200f682a4a
set proxy explicit-web
set dstintf "wan2"
set srcaddr "all"
set dstaddr "test321"
set service "webproxy"
set action accept
set schedule "always"
set utm-status enable
set av-profile "default"
set profile-protocol-options "test"
set ssl-ssh-profile "test"
end

```

Matching query strings

To match the video with URL youtube.com/watch?v=XXXXXXXXXX, (where XXXXXXXXX is an example YouTube query string) you need to match an HTTP request with the following format:

```

GET /user/watch?v=GLCH1ldlwQsg HTTP/1.1
Host: www.youtube.com
User-Agent: curl/7.52.1
Accept: */*

```

Create the following web proxy addresses to match this video or query string:

```

config firewall proxy-address
  edit "youtube"
    set uuid 4ad63880-971e-51e7-7b2e-c69423ac6314
    set type host-regex
    set host-regex ".*youtube.com"
  next
  edit "query-string"
    set uuid 7687a8c0-9727-51e7-5063-05edda03abbf
    set host "youtube"
    set path "/watch"
    set query "v=XXXXXXXXXX"
  end

```

Then create two proxy policies, one that allows access to all traffic and a second one that blocks access to the page that matches the query string

```

config firewall proxy-policy
  edit 1
    set uuid 92273e4e-8c53-51e7-a7bd-f26e6e15fc98
    set proxy explicit-web
    set dstintf "wan2"
    set srcaddr "all"
    set dstaddr "all"
    set service "webproxy-connect"
    set action accept
    set schedule "always"
    set utm-status enable
    set profile-protocol-options "test"
    set ssl-ssh-profile "test"
  next
  edit 2
    set uuid d35ad06a-8c53-51e7-8511-17200f682a4a
    set proxy explicit-web
    set dstintf "wan2"

```

```
set srcaddr "all"
set dstaddr "query-string"
set service "webproxy"
set action accept
set schedule "always"
set utm-status enable
set av-profile "default"
set profile-protocol-options "test"
set ssl-ssh-profile "test"
next
end
```

Multiple web proxy PAC files in one VDOM

Proxy auto-config (PAC) files automatically choose the appropriate proxy server for browsers and other user agents. Not every user in an organization has the same proxy server requirements. Supporting multiple PAC files provides granular control. To manage multiple PAC files, you use PAC policies.

This capability is available only when the FortiGate is in **Proxy-based** inspection mode.

If there is no matching PAC policy (by name), in the PAC policies, the global PAC file is used by default.

To enable Proxy mode:

GUI

1. Go to **System > Settings**.
2. In **System Operation Settings**, set the **Inspection Mode** to **Proxy**.

CLI

```
config system settings
  set inspection-mode proxy
end
```

To configure a PAC policy

```
config web-proxy explicit
  set status enable
  set pack-file-server-status enable
  config pac-policy
    edit <policy ID#>
      set srcaddr <name of IPv4 address object>
      set srcaddr6 <name of IPv6 address object>
      set dstaddr <name of address object>
      set pac-file-name <string>
      set pac-file-data "<PAC-file>"
    end
  end
```

Option	Description
srcaddr or srcaddr6	This address must conform to the following criteria: <ul style="list-style-type: none"> • a range, mask or wildcard mask type of address or address group • source type proxy-address or group It can be either IPv4 or IPv6.
dstaddr	This address must conform to the following criteria: <ul style="list-style-type: none"> • a range, mask or wildcard type of address or address group • it must be resolved as the FortiGate address
pac-file-name	Name of the PAC file.
pac-file-data	Enter the contents of the PAC file enclosed in quotes. It is permissible to use the Return key when entering the contents. Place the closing quote at the end of the last line. If quotes are used within the content of the file, use the escape character \ before the quote. Example: \"

The `pack-file-server-status` setting must be set to `enable` in order for the `config pac-policy` command to work.

Explicit proxy concepts

The following is information that is specific to Explicit Proxy concepts. Any information that is common to Web Proxy in general is covered in the more inclusive section of [Web proxy concepts on page 357](#)

The FortiGate explicit web proxy

You can use the FortiGate explicit web proxy to enable explicit proxying of IPv4 and IPv6 HTTP, and HTTPS traffic on one or more FortiGate interfaces. The explicit web proxy also supports proxying FTP sessions from a web browser and proxy auto-config (PAC) to provide automatic proxy configurations for explicit web proxy users. From the CLI you can also configure the explicit web proxy to support SOCKS sessions from a web browser.

The explicit web and FTP proxies can be operating at the same time on the same or on different FortiGate interfaces.



If explicit web proxy options are not visible on the web-based manager, go to **System > Feature Visibility** and turn on **Explicit Proxy**.

In most cases you would configure the explicit web proxy for users on a network by enabling the explicit web proxy on the FortiGate interface connected to that network. Users on the network would configure their web browsers to use a proxy server for HTTP and HTTPS, FTP, or SOCKS and set the proxy server IP address to the IP address of the FortiGate interface connected to their network. Users could also enter the PAC URL into their web browser PAC configuration to automate their web proxy configuration using a PAC file stored on the FortiGate unit.



Enabling the explicit web proxy on an interface connected to the Internet is a security risk because anyone on the Internet who finds the proxy could use it to hide their source address.

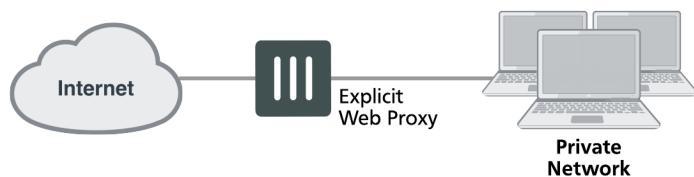
If the FortiGate unit is operating in transparent mode, users would configure their browsers to use a proxy server with the FortiGate management IP address.

If the FortiGate unit is operating with multiple VDOMs the explicit web proxy is configured for each VDOM.

The web proxy receives web browser sessions to be proxied at FortiGate interfaces with the explicit web proxy enabled. The web proxy uses FortiGate routing to route sessions through the FortiGate unit to a destination interface. Before a session leaves the exiting interface, the explicit web proxy changes the source addresses of the session packets to the IP address of the exiting interface. When the FortiGate unit is operating in transparent mode the explicit web proxy changes the source addresses to the management IP address. You can configure the explicit web proxy to keep the original client IP address. See [The FortiGate explicit web proxy on page 374](#).

For more information about explicit web proxy sessions, see [The FortiGate explicit web proxy on page 374](#).

Example explicit web proxy topology



To allow all explicit web proxy traffic to pass through the FortiGate unit you can set the explicit web proxy default firewall policy action to accept. However, in most cases you would want to use security policies to control explicit web proxy traffic and apply security features such as access control/authentication, virus scanning, web filtering, application control, and traffic logging. You can do this by keeping the default explicit web proxy security policy action to deny and then adding web-proxy security policies.

You can also change the explicit web proxy default security policy action to accept and add explicit web proxy security policies. If you do this, sessions that match web-proxy security policies are processed according to the security policy settings. Connections to the explicit web proxy that do not match a web-proxy security policy are allowed with no restrictions or additional security processing. This configuration is not recommended and is not a best practice.

The explicit web-proxy can accept VIP addresses for destination address. If an external IP matches a VIP policy, the IP is changed to the mapped-IP of the VIP.

Web-proxy policies can selectively accept or deny traffic, apply authentication, enable traffic logging, and use security profiles to apply virus scanning, web filtering, IPS, application control, DLP, and SSL/SSH inspection to explicit web proxy traffic.

You cannot configure IPsec, SSL VPN, or Traffic shaping for explicit web proxy traffic. Web Proxy policies can only include firewall addresses not assigned to a FortiGate unit interface or with interface set to **Any**. (On the web-based manager you must set the interface to **Any**. In the CLI you must unset the associated-interface.)

Authentication of explicit web proxy sessions uses HTTP authentication and can be based on the user's source IP address or on cookies from the user's web browser. For more information, see [The FortiGate explicit web proxy on page 374](#).

To use the explicit web proxy, users must add the IP address of a FortiGate interface on which the explicit web proxy is enabled and the explicit web proxy port number (default 8080) to the proxy configuration settings of their web browsers.

On FortiGate units that support it, you can also enable web caching for explicit web proxy sessions.



For the time being, traffic shaping is not supported per policy for explicit proxy. For explicit proxy traffic, traffic shaping can be carried out per interface.

Other explicit web proxy options

You can change the following explicit web proxy options as required by your configuration.

HTTP port, HTTPS port, FTP port, PAC port

The TCP port that web browsers use to connect to the explicit proxy for HTTP, HTTPS, FTP and PAC services. The default port is 8080 for all services. By default HTTPS, FTP. and PAC use the same port as HTTP. You can change any of these ports as required. Users configuring their web browsers to use the explicit web proxy should add the same port numbers to their browser configurations.

Multi-port support for Explicit Proxy

Support exists for the use of multiple ports and port range in the explicit FTP or Web proxies. These changes have been added in both CLI and GUI.

CLI:

```
set http-incoming-port <port_low>[-<port_high>]
```

Where:

- `port_low` - the low value of the port
- `port_high` - the high value of the port

The `port_high` value can be omitted if `port_low` and `port_high` are the same.

Proxy FQDN

Enter the fully qualified domain name (FQDN) for the proxy server. This is the domain name to enter into browsers to access the proxy server.

Max HTTP request length

Enter the maximum length of an HTTP request in Kbytes. Larger requests will be rejected.

Max HTTP message length

Enter the maximum length of an HTTP message in Kbytes. Larger messages will be rejected.

Multiple incoming ports and port ranges

Web proxy can be configured to listen on multiple ports on the same IP as well as listen for HTTP and HTTPS on those same (or different) ports. This is done in the CLI.

Define the IP ranges using a hyphen (-). As shown below, `port_high` is not necessary to specify if `port_low` is equal to `port_high`.

CLI syntax

```
config web-proxy explicit
    set http-incoming-port <port_low> [-<port_high>]
end
```

Internet services

FortiOS can use the Internet Service Database (introduced in 5.4.1) as a web-proxy policy matching factor. This can only be done in the CLI.

CLI syntax:

```
config firewall proxy-policy
  edit 0
    set internet-service <application-id>
    set internet-service-custom <application-name>
```

IP pools

IP Pools can be used with web proxy. When using this option of setting the IP pool name, the outgoing IP will be selected.

CLI syntax

```
config firewall proxy-policy
  edit <example>
    set poolname <name>
  end
```

Proxy chaining (web proxy forwarding servers)

For the explicit web proxy you can configure web proxy forwarding servers to use proxy chaining to redirect web proxy sessions to other proxy servers. Proxy chaining can be used to forward web proxy sessions from the FortiGate unit to one or more other proxy servers on your network or on a remote network. You can use proxy chaining to integrate the FortiGate explicit web proxy with an web proxy solution that you already have in place.

A FortiGate unit can forward sessions to most web proxy servers including a remote FortiGate unit with the explicit web proxy enabled. No special configuration of the explicit web proxy on the remote FortiGate unit is required.

You can deploy the explicit web proxy with proxy chaining in an enterprise environment consisting of small satellite offices and a main office. If each office has a FortiGate unit, users at each of the satellite offices can use their local FortiGate unit as an explicit web proxy server. The satellite office FortiGate units can forward explicit web proxy sessions to an explicit web proxy server at the central office. From here the sessions can connect to web servers on the Internet.

FortiGate proxy chaining does not support authenticating with the remote forwarding server.

Adding a web proxy forwarding server

To add a forwarding server, select **Create New** in the **Web Proxy Forwarding Servers** section of the **Explicit Proxy** page by going to **Network > Explicit Proxy**.

Server Name	Enter the name of the forwarding server.
--------------------	--

Proxy Address	Enter the IP address of the forwarding server.
Proxy Address Type	Select the type of IP address of the forwarding server. A forwarding server can have an FQDN or IP address.
Port	Enter the port number on which the proxy receives connections. Traffic leaving the FortiGate explicit web proxy for this server has its destination port number changed to this number.
Server Down action	Select what action the explicit web proxy to take if the forwarding server is down. Block means if the remote server is down block traffic. Use Original Server means do not forward traffic to the forwarding sever but instead forward it from the FortiGate to its destination. In other words operate as if there is no forwarding server configured.
Enable Health Monitor	Select to enable health check monitoring and enter the address of a remote site. See " Web proxy forwarding server monitoring and health checking ".
Health Check Monitor Site	

Use the following CLI command to add a web proxy forwarding server named `fwd-srv` at address `proxy.example.com` and port 8080.

```
config web-proxy forward-server
  edit fwd-srv
    set addr-type fqdn
    set fqdn proxy.example.com
    set port 8080
  end
```

Web proxy forwarding server monitoring and health checking

By default, a FortiGate unit monitors web proxy forwarding server by forwarding a connection to the remote server every 10 seconds. If the remote server does not respond it is assumed to be down. Checking continues and when the server does send a response the server is assumed to be back up. If you configure health checking, every 10 seconds the FortiGate unit attempts to get a response from a web server by connecting through the remote forwarding server.

You can configure health checking for each remote server and specify a different website to check for each one.

If the remote server is found to be down you can configure the FortiGate unit to block sessions until the server comes back up or to allow sessions to connect to their destination, bypassing the remote forwarding server. You cannot configure the FortiGate unit to fail over to another remote forwarding server.

Configure the server down action and enable health monitoring from the web-based manager by going to **Network > Explicit Proxy**, selecting a forwarding server, and changing the server down action and changing the health monitor settings.

Use the following CLI command to enable health checking for a web proxy forwarding server and set the server down option to bypass the forwarding server if it is down.

```

config web-proxy forward-server
  edit fwd-srv
    set healthcheck enable
    set monitor http://example.com
    set server-down-option pass
  end

```

Grouping forwarding servers and load balancing traffic to them

You can add multiple web proxy forwarding servers to a forwarding server group and then add the server group to an explicit web proxy policy instead of adding a single server. Forwarding server groups are created from the FortiGate CLI but can be added to policies from the web-based manager (or from the CLI).

When you create a forwarding server group you can select a load balancing method to control how sessions are load balanced to the forwarding servers in the server group. Two load balancing methods are available:

- **Weighted** load balancing sends more sessions to the servers with higher weights. You can configure the weight for each server when you add it to the group.
- **Least-session** load balancing sends new sessions to the forwarding server that is processing the fewest sessions.

When you create a forwarding server group you can also enable **affinity**. Enable affinity to have requests from the same client processed by the same server. This can reduce delays caused by using multiple servers for a single multi-step client operation. Affinity takes precedence over load balancing.

You can also configure the behavior of the group if all of the servers in the group are down. You can select to **block** traffic or you can select to have the traffic **pass** through the FortiGate explicit proxy directly to its destination instead of being sent to one of the forwarding servers.

Use the following command to add a forwarding server group that uses weighted load balancing to load balance traffic to three forwarding servers. Server weights are configured to send most traffic to server2. The group has affinity enabled and blocks traffic if all of the forward servers are down:

```

config web-proxy forward-server
  edit server_1
    set ip 172.20.120.12
    set port 8080
  next
  edit server_2
    set ip 172.20.120.13
    set port 8000
  next
  edit server_3
    set ip 172.20.120.14
    set port 8090
  next
end

config web-proxy forward-server-group
  edit New-fwd-group
    set affinity enable
    set ldb-method weight
    set group-down-option block
    config server-list
      edit server_1
        set weight 10
      next
      edit server_2
        set weight 40

```

```

next
edit server_3
    set weight 10
next
end

```

Adding proxy chaining to an explicit web proxy policy

You enable proxy chaining for web proxy sessions by adding a web proxy forwarding server or server group to an explicit web proxy policy. In a policy you can select one web proxy forwarding server or server group. All explicit web proxy traffic accepted by this security policy is forwarded to the specified web proxy forwarding server or server group.

To add an explicit web proxy forwarding server - web-based manager:

1. Go to **Policy & Objects > Proxy Policy** and select **Create New**.
2. Configure the policy:

Explicit Proxy Type	Web
Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	all
Schedule	always
Action	ACCEPT
Web Proxy Forwarding Server	Select, fwd-srv

3. Select **OK** to save the security policy.

To add an explicit web proxy forwarding server - CLI:

1. Use the following command to add a security policy that allows all users on the 10.31.101.0 subnet to use the explicit web proxy for connections through the wan1 interface to the Internet. The policy forwards web proxy sessions to a remote forwarding server named `fwd-srv`

```

config firewall proxy-policy
    edit 0
        set proxy explicit-web
        set dstintf wan1
        set scraddr Internal_subnet
        set dstaddr all
        set action accept
        set schedule always
        set webproxy-forward-server fwd-srv
    end

```

Security profiles, threat weight, device identification, and the explicit web proxy

You can apply all security profiles to explicit web proxy sessions. This includes antivirus, web filtering, intrusion protection (IPS), application control, data leak prevention (DLP), and SSL/SSH inspection. Security profiles are applied by selecting them in an explicit web proxy policy or in authentication rules added to web proxy policies.

Traffic accepted by explicit web proxy policies contributes to threat weight data.

The explicit web proxy is not compatible with device identification.

Since the traffic accepted by the explicit web proxy is known to be either HTTP, HTTPS, or FTP over HTTP and since the ports are already known by the proxy, the explicit web proxy does not use all of the SSL/SSH inspection options. The explicit web proxy does support the following proxy options:

- Enable chunked bypass
- HTTP oversized file action and threshold

The explicit web proxy does not support the following proxy options:

- Client comforting
- Server comforting
- Monitor content information from dashboard. URLs visited by explicit web proxy users are not added to dashboard usage and log and archive statistics widgets.

For explicit web proxy sessions, the FortiGate unit applies antivirus scanning to HTTP POST requests and HTTP responses. The FortiGate unit starts virus scanning a file in an HTTP session when it receives a file in the body of an HTML request. The explicit web proxy can receive HTTP responses from either the originating web server or the FortiGate web cache module.

Explicit web proxy sessions and user limits

Web browsers and web servers open and close multiple sessions with the explicit web proxy. Some sessions open and close very quickly. HTTP 1.1 keepalive sessions are persistent and can remain open for long periods of time. Sessions can remain on the explicit web proxy session list after a user has stopped using the proxy (and has, for example, closed their browser). If an explicit web proxy session is idle for more than 3600 seconds it is torn down by the explicit web proxy. See [RFC 2616](#) for information about HTTP keepalive/persistent HTTP sessions.

This section describes proxy sessions and user limits for both the explicit web proxy and the explicit FTP proxy. Session and user limits for the two proxies are counted and calculated together. However, in most cases if both proxies are active there will be many more web proxy sessions than FTP proxy sessions.

The FortiGate unit adds two sessions to its session table for every explicit proxy session started by a web browser and every FTP session started by an FTP client. An entry is added to the session table for the session from the web browser or client to the explicit proxy. All of these sessions have the same destination port as the explicit web proxy port (usually 8080 for HTTP and 21 for FTP). An entry is also added to the session table for the session between the exiting FortiGate interface and the web or FTP server destination of the session. All of these sessions have a FortiGate interface IP address and the source address of the session and usually have a destination port of 80 for HTTP and 21 for FTP.

Proxy sessions that appear in FortiView do not include the Policy ID of the web-proxy or ftp-proxy security policy that accepted them. However, the explicit proxy sessions include a destination port that matches the explicit

proxy port number (usually 8080 for the web proxy and 21 for the FTP proxy). The proxied sessions from the FortiGate unit have their source address set to the IP address of the FortiGate unit interface that the sessions use to connect to their destinations (for example, for connections to the Internet the source address would be the IP address of the FortiGate interface connected to the Internet).

FortiOS limits the number of explicit proxy users. This includes both explicit FTP proxy and explicit web proxy users. The number of users varies by FortiGate model from as low as 10 to up to 18000 for high end models. You cannot raise this limit.

If your FortiGate unit is configured for multiple VDOMs you can go to **System > Global Resources** to view the maximum number of **Concurrent explicit proxy users** and optionally reduce the limit. You can also use the following command:

```
config global
  config system resource-limits
    set proxy 50
  end
end
```

To limit the number of explicit proxy users for a VDOM, from the web-based manager enable multiple VDOMs and go to **System > VDOM** and edit a VDOM or use the following command to change the number of explicit web proxy users for VDOM_1:

```
config global
  config system vdom-property
    edit VDOM_1
      set proxy 25
  end
end
```

You can use the `diagnose wad user list` command to view the number of explicit web proxy users. Users may be displayed with this command even if they are no longer actively using the proxy. All idle sessions time out after 3600 seconds.

You can use the command `diagnose wad user clear` to clear current explicit proxy users. You can also use the command `diagnose wad user clear <user-name>` to clear individual users. This means delete information about all users and force them re-authenticate.



Users that authenticate with explicit web-proxy or ftp-proxy security policies do not appear in the **Monitor > Firewall User Monitor** list and selecting **De-authenticate All Users** has no effect on explicit proxy users.

How the number of concurrent explicit proxy users is determined depends on their authentication method:

- For session-based authenticated users, each authenticated user is counted as a single user. Since multiple users can have the same user name, the proxy attempts to identify users according to their authentication membership (based upon whether they were authenticated using RADIUS, LADAP, FSAE, local database etc.). If a user of one session has the same name and membership as a user of another session, the explicit proxy assumes this is one user.
- For IP Based authentication, or no authentication, or if no web-proxy security policy has been added, the source IP address is used to determine a user. All sessions from a single source address are assumed to be from the same user.

The explicit proxy does not limit the number of active sessions for each user. As a result the actual explicit proxy session count is usually much higher than the number of explicit web proxy users. If an excessive number of

explicit web proxy sessions is compromising system performance you can limit the amount of users if the FortiGate unit is operating with multiple VDOMs.

Explicit proxy configuration

The following is information that is specific to Explicit Proxy configuration. Any configuration information that is common to Web Proxy in general is covered in the more inclusive section of [Web proxy configuration on page 365](#).

Configuring an external IP address for the IPv4 explicit web proxy

You can use the following command to set an external IP address (or pool) that will be used by the explicit web proxy policy.

```
config web-proxy explicit
    set status enable
    set outgoing-ip <ip1> <ip2> ... <ipN>
end
```

Configuring an external IP address for the IPv6 explicit web proxy

You can use the following command to set an external IP address (or pool) that will be used by the explicit web proxy policy.

```
config web-proxy explicit
    set status enable
    set outgoing-ipv6 <ip1> <ip2> ... <ipN>
end
```

Restricting the IP address of the IPv4 explicit web proxy

You can use the following command to restrict access to the explicit web proxy using only one IP address. The IP address that you specify must be the IP address of an interface that the explicit web proxy is enabled on. You might want to use this option if the explicit FTP proxy is enabled on an interface with multiple IP addresses.

For example, to require users to connect to the IP address 10.31.101.100 to connect to the explicit web proxy:

```
config web-proxy explicit
    set incoming-ip 10.31.101.100
end
```

Restricting the outgoing source IP address of the IPv4 explicit web proxy

You can use the following command to restrict the source address of outgoing web proxy packets to a single IP address. The IP address that you specify must be the IP address of an interface that the explicit web proxy is

enabled on. You might want to use this option if the explicit web proxy is enabled on an interface with multiple IP addresses.

For example, to restrict the outgoing packet source address to 172.20.120.100:

```
config web-proxy explicit
    set outgoing-ip 172.20.120.100
end
```

Restricting the IP address of the explicit IPv6 web proxy

You can use the following command to restrict access to the IPv6 explicit web proxy to use only one IPv6 IP address. The IPv6 address that you specify must be the IPv6 address of an interface that the explicit web proxy is enabled on. You might want to use this option if the explicit web proxy is enabled on an interface with multiple IPv6 addresses.

For example, to require users to connect to the IPv6 address 2001:db8:0:2::30 to connect to the explicit IPv6 web proxy:

```
config web-proxy explicit
    set incoming-ipv6 2001:db8:0:2::30
end
```

Restricting the outgoing source IP address of the IPv6 explicit web proxy

You can use the following command to restrict the source address of outgoing web proxy packets to a single IPv6 address. The IP address that you specify must be the IPv6 address of an interface that the explicit web proxy is enabled on. You might want to use this option if the explicit web proxy is enabled on an interface with multiple IPv6 addresses.

For example, to restrict the outgoing packet source address to 2001:db8:0:2::50:

```
config web-proxy explicit
    set outgoing-ipv6 2001:db8:0:2::50
end
```

Explicit proxy firewall address types

Explicit proxy firewall address types improve granularity over header matching for explicit web proxy policies. You can enable this option using the **Show in Address List** button on the Address and Address Group New/Edit forms under **Policy & Objects > Addresses**.

The following address types are available:

- **URL Pattern** - destination address
- **Host Regex Match** - destination address
- **URL Category** - destination address (URL filtering)
- **HTTP Method** - source address
- **User Agent** - source address

- **HTTP Header** - source address
- **Advanced (Source)** - source address (combines User Agent, HTTP Method, and HTTP Header)
- **Advanced (Destination)** - destination address (combines Host Regex Match and URL Category)

Proxy auto-config (PAC) configuration

A proxy auto-config (PAC) file defines how web browsers can choose a proxy server for receiving HTTP content. PAC files include the `FindProxyForURL(url, host)` JavaScript function that returns a string with one or more access method specifications. These specifications cause the web browser to use a particular proxy server or to connect directly.

To configure PAC for explicit web proxy users, you can use the port that PAC traffic from client web browsers use to connect to the explicit web proxy. explicit web proxy users must configure their web browser's PAC proxy settings to use the PAC port.

PAC file content

You can edit the default PAC file from the web-based manager or use the following command to upload a custom PAC file:

```
config web-proxy explicit
    set pac-file-server-status enable
    set pac-file-data <pac_file_str>
end
```

Where `<pac_file_str>` is the contents of the PAC file. Enter the PAC file text in quotes. You can copy the contents of a PAC text file and paste the contents into the CLI using this option. Enter the command followed by two sets of quotes then place the cursor between the quotes and paste the file content.

The maximum PAC file size is 256 kbytes. If your FortiGate unit is operating with multiple VDOMs each VDOM has its own PAC file. The total amount of FortiGate memory available to store all of these PAC files 2 MBytes. If this limit is reached you will not be able to load any additional PAC files.

You can use any PAC file syntax that is supported by your users's browsers. The FortiGate unit does not parse the PAC file.

To use PAC, users must add an automatic proxy configuration URL (or PAC URL) to their web browser proxy configuration. The default FortiGate PAC file URL is:

```
http://<interface_ip>:<PAC_port_int>/<pac_file_str>
```

For example, if the interface with the explicit web proxy has IP address 172.20.120.122, the PAC port is the same as the default HTTP explicit web proxy port (8080) and the PAC file name is proxy.pac the PAC file URL would be:

```
http://172.20.120.122:8080/proxy.pac
```

From the CLI you can use the following command to display the PAC file URLs:

```
get web-proxy explicit
```

Unknown HTTP version

You can select the action to take when the proxy server must handle an unknown HTTP version request or message. Set unknown HTTP version to Reject or Best Effort. Best Effort attempts to handle the HTTP traffic as best as it can. Reject treats known HTTP traffic as malformed and drops it. The Reject option is more secure.

Authentication realm

You can enter an authentication realm to identify the explicit web proxy. The realm can be any text string of up to 63 characters. If the realm includes spaces enclose it in quotes. When a user authenticates with the explicit web proxy the HTTP authentication dialog includes the realm so you can use the realm to identify the explicitly web proxy for your users.

Implementing botnet features

The option `scan-botnet-connections` can be added to an explicit proxy policy.

CLI Syntax:

```
config firewall proxy-policy
  edit <policy_id>
    set scan-botnet-connections [disable|block|monitor]
  end
```

where:

- `disable` means do not scan connections to botnet servers.
- `block` means block connections to botnet servers.
- `monitor` means log connections to botnet servers.

Adding disclaimer messages to explicit proxy policies

This feature allows you to create user exceptions for specific URL categories (including warning messages) based on user groups. The **Disclaimer Options** are configured under **Policy & Objects > Proxy Policy**.

You can also configure a disclaimer for each Authentication Rule by setting **Action to Authenticate**.

Disclaimer explanations

- **Disable:** No disclaimer (default setting).
- **By Domain:** The disclaimer will be displayed on different domains. The explicit web proxy will check the referring header to mitigate the javascript/css/images/video/etc page.
- **By Policy:** The disclaimer will be displayed if the HTTP request matches a different explicit firewall policy.
- **By User:** The disclaimer will be displayed when a new user logs on.

Changing HTTP headers

You can create explicit web proxy profiles that can add, remove and change HTTP headers. The explicit web proxy profile can be added to a web explicit proxy policy and will be applied to all of the HTTP traffic accepted by that policy.

You can change the following HTTP headers:

- client-ip
- via header for forwarded requests
- via header for forwarded responses
- x-forwarded-for
- front-end-https

For each of these headers you can set the action to:

- Pass to forward the traffic without changing the header
- Add to add the header
- Remove to remove the header

You can also configure how the explicit web proxy handles custom headers. The proxy can add or remove custom headers from requests or responses. If you are adding a header you can specify the content to be included in the added header.

Create web proxy profiles from the CLI:

```
config web-proxy profile
    edit <name>
        set header-client-ip {add | pass | remove}
        set header-via-request {add | pass | remove}
        set header-via-response {add | pass | remove}
        set header-x-forwarded-for {add | pass | remove}
        set header-front-end-https {add | pass | remove}
        config headers
            edit <id>
                set action {add-to-request | add-to-response | remove-from-request | remove-from-response}
                set content <string>
                set name <name>
            end
        end
    end
```

Use the following command to add a web proxy profile to an explicit proxy policy:

```
config firewall proxy-policy
    edit <id>
        set webproxy-profile <name>
    end
```

Preventing the explicit web proxy from changing source addresses

By default in NAT mode the explicit web proxy changes the source address of packets leaving the FortiGate to the IP address of the FortiGate interface that the packets are exiting from. In transparent mode the source address is

changed to the management IP.

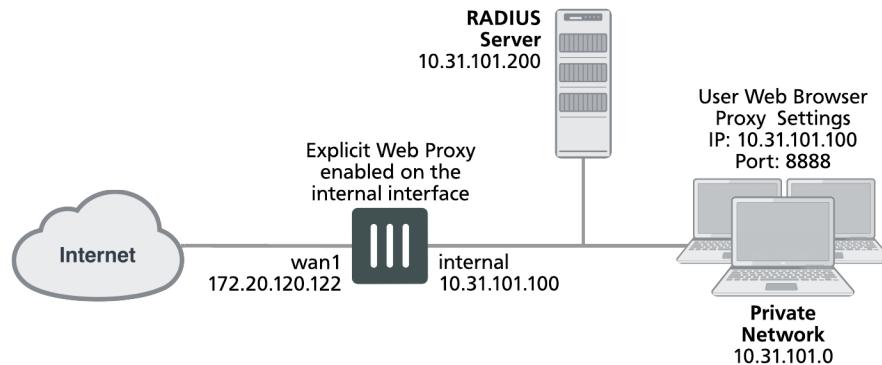
This configuration hides the IP addresses of clients and allows packets to return to the FortiGate unit interface without having to route packets from clients. You can use the following command to configure the explicit web proxy to keep the original client's source IP address:

```
config firewall proxy-policy
  edit 0
    set proxy explicit-web
    set transparent enable
  end
```

Example users on an internal network browsing the Internet through the explicit web proxy with web caching, RADIUS authentication, web filtering, and virus scanning

This example describes how to configure the explicit web proxy for the example network shown below. In this example, users on the internal network connect to the explicit web proxy through the Internal interface of the FortiGate unit. The explicit web proxy is configured to use port 8888 so users must configure their web browser proxy settings to use port 8888 and IP address 10.31.101.100.

Example explicit web proxy network topology



Explicit web proxy users must authenticate with a RADIUS server before getting access to the proxy. The explicit proxy policy that accepts explicit web proxy traffic applies per session authentication and includes a RADIUS server user group. The authentication rule also applies web filtering and virus scanning.

General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Enable the explicit web proxy for HTTP and HTTPS and change the HTTP and HTTPS ports to 8888.
2. Enable the explicit web proxy on the internal interface.
3. Add a RADIUS server and user group for the explicit web proxy.
4. Add an authentication explicit proxy policy. Enable web caching. Add an authentication rule and enable antivirus and web filtering.

Configuring the explicit web proxy - web-based manager

Use the following steps to configure the explicit web proxy.

To enable and configure the explicit web proxy

1. Go to **System > Feature Visibility** and turn on the **Explicit Proxy** feature.
2. Go to **Network > Explicit Proxy** and change the following settings:

Enable Explicit Web Proxy	Select HTTP/HTTPS .
Listen on Interfaces	No change. This field will eventually show that the explicit web proxy is enabled for the Internal interface.
HTTP Port	8888
HTTPS Port	0
Realm	You are authenticating with the explicit web proxy.
Default Firewall Policy Action	Deny

3. Select **Apply**.

To enable the explicit web proxy on the Internal interface

1. Go to **Network > Interfaces**.
2. Edit the internal interface.
3. Select **Enable Explicit Web Proxy**.
4. Select **OK**.

To add a RADIUS server and user group for the explicit web proxy

1. Go to **User & Device > RADIUS Servers** and select **Create New** to add a new RADIUS server:

Name	RADIUS_1
Primary Server Name/IP	10.31.101.200
Primary Server Secret	RADIUS_server_secret

2. Select **OK**.
3. Go to **User & Device > User Groups** and select **Create New** to add a new user group.

Name	Explicit_proxy_user_group
Type	Firewall
Remote Groups	RADIUS_1
Group Name	Any

4. Select **OK**.

To add an explicit proxy policy

1. Go to **Policy & Objects > Addresses** and select **Create New**.
2. Add a firewall address for the internal network:

Category	Address
Name	Internal_subnet
Type	Subnet / IP Range
Subnet / IP Range	10.31.101.0
Interface	Any

3. Go to **Policy & Objects > Proxy Policy** and select **Create New**.
4. Configure the explicit web proxy policy.

Explicit Proxy Type	Web
Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	all
Action	AUTHENTICATE

5. Under **Configure Authentication Rules** select **Create New** to add an authentication rule:

Groups	Explicit_policy
Source User(s)	Leave blank
Schedule	always

6. Turn on **Antivirus** and **Web Filter** and select the **default** profiles for both.
7. Select the **default** proxy options profile.
8. Select **OK**.
9. Make sure **Enable IP Based Authentication** is not selected.
10. Turn on **Web Cache**.
11. Select **OK**.

Configuring the explicit web proxy - CLI

Use the following steps to configure the example explicit web proxy configuration from the CLI.

To enable the explicit web proxy on the Internal interface

1. Enter the following command to enable the explicit web proxy on the internal interface.

```

config system interface
  edit internal
    set explicit-web-proxy enable
end

```

To enable and configure the explicit web proxy

1. Enter the following command to enable the explicit web proxy and set the TCP port that proxy accepts HTTP and HTTPS connections on to 8888.

```

config web-proxy explicit
  set status enable
  set http-incoming-port 8888
  set https-incoming-port 8888
  set realm "You are authenticating with the explicit web proxy"
  set sec-default-action deny
end

```

To add a RADIUS server and user group for the explicit web proxy

1. Enter the following command to add a RADIUS server:

```

config user radius
  edit RADIUS_1
    set server 10.31.101.200
    set secret RADIUS_server_secret
end

```

2. Enter the following command to add a user group for the RADIUS server.

```

config user group
  edit Explicit_proxy_user_group
    set group-type firewall
    set member RADIUS_1
end

```

To add a security policy for the explicit web proxy

1. Enter the following command to add a firewall address for the internal subnet:

```

config firewall address
  edit Internal_subnet
    set type iprange
    set start-ip 10.31.101.1
    set end-ip 10.31.101.255
end

```

2. Enter the following command to add the explicit web proxy security policy:

```

config firewall proxy-policy
  edit 0
    set proxy explicit-web
    set dstintf wan1
    set srcaddr Internal_subnet
    set dstaddr all
    set action accept
    set service webproxy
    set webcache enable
    set identity-based enable
    set ipbased disable

```

```
set active-auth-method basic
set groups <User group>
end
```

Testing and troubleshooting the configuration

You can use the following steps to verify that the explicit web proxy configuration is working as expected:

To test the explicit web proxy configuration

1. Configure a web browser on the internal subnet to use a web proxy server at IP address 10.31.101.100 and port 8888.

2. Browse to an Internet web page.

The web browser should pop up an authentication window that includes the phrase that you added to the Realm option.

3. Enter the username and password for an account on the RADIUS server.

If the account is valid you should be allowed to browse web pages on the Internet.

4. Close the browser and clear its cache and cookies.

5. Restart the browser and connect to the Internet.

You could also start a second web browser on the same PC. Or you could start a new instance of the same browser as long as the browser asks for a user name and password again.

You should have to authenticate again because identity-based policies are set to session-based authentication.

6. If this basic functionality does not work, check your FortiGate and web browser configuration settings.

7. Browse to a URL on the URL filter list and confirm that the web page is blocked.

8. Browse to <http://eicar.org> and attempt to download an anti-malware test file.

The antivirus configuration should block the file.

Sessions for web-proxy security policies do not appear on the Top Sessions dashboard widget and the count column for security policies does not display a count for explicit web proxy security policies.

9. You can use the following command to display explicit web proxy sessions

```
get test wad 60
IP based users:

Session based users:
  user:0x9c20778, username:User1, vf_id:0, ref_cnt:9

Total allocated user:1
```

```
Total user count:3, shared user quota:50, shared user count:3
```

This command output shows one explicit proxy user with user name User1 authenticated using session-based authentication.

Kerberos and NTLM authentication

FortiOS recognizes the client's authentication method from the token and selects the correct authentication scheme to authenticate successfully.

CLI syntax

```
config firewall proxy-policy
  edit 0
```

```
set active-auth-method [ntlm|basic|digest|negotiate|none]
end
```

Kerberos authentication for explicit proxy users

Kerberos authentication is a method for authenticating both explicit web proxy and transparent web proxy users. It has several advantages over NTLM challenge response:

- Does not require FSSO/AD agents to be deployed across domains.
- Requires fewer round-trips than NTLM SSO, making it less latency sensitive.
- Is (probably) more scalable than challenge response.
- Uses existing Windows domain components rather than added components.
- NTLM may still be used as a fallback for non-Kerberos clients.

Enhancements to Kerberos explicit and transparent web proxy

FortiOS 5.6.x authentication is managed by schemes and rules based on protocol and source address. As such, configurable authentication settings have been introduced to enhance authentication.

CLI commands (`config authentication rule`, `scheme`, and `setting`) allow explicit proxy rules and schemes to be created to separate user authentication (e.g. authentication rules and schemes used to match conditions in order to identify users) from user authorization (proxy-based policies with users and/or user groups).

CLI syntax - config authentication rule

```
config authentication rule
  edit <name>
    set name <name>
    set status {enable|disable}
    set protocol {http|ftp|socks}
    config srcaddr <addr-name or addrgrp-name>
      edit <name>
        set name <ipv4-policy-name>
      next
    end
    config srcaddr6 <addr-name or addrgrp-name>
      edit <name>
        set name <ipv6-policy-name>
      next
    end
    set ip-based {enable|disable}
    set active-auth-method <scheme-name>
    set sso-auth-method <scheme-name>
    set transaction-based {enable|disable} - basic scheme + session-based
    set web-auth-cookie {enable|disable}
    set comments <comments>
  next
end
```

Note: As shown above, HTTP, FTP, and SOCKSv5 authentication protocols are supported for explicit proxy.

Authentication rules are used to receive user-identity, based on the values set for protocol and source address. Having said this, if a rule fails to match based on source address, there will be no other attempt to match the rule, however the next policy will be attempted. This occurs only when:

- there is an authentication rule, but no authentication method has been set (under `config authentication scheme`; see below), so user identity cannot be found.
- the user is successfully matched in the rule, but fails to match the current policy.

Once a rule is positively matched through protocol and/or source address, it must also match the authentication method specified (`active-auth-method` and `sso-auth-method`). These methods point to schemes, as defined under `config authentication scheme`.

CLI syntax - config authentication scheme

```
config authentication scheme
  edit <name>
    set name <name>
    set method {basic|digest|ntlm|form|negotiate|fssso|rssso}
    set negotiate-ntlm {enable|disable}
    set require-tfa {enable|disable}
    set fssso-guest {enable|disable}
    config user-database
      edit <name>
        set name {local|<ldap-server>|<radius-server>|<fssso-name>|<rssso-name>|<tacacs+-name>}
      next
    end
  next
end
```

Combining authentication rules and schemes, granular control can be exerted over users and IPs, creating an efficient process for users to successfully match a criteria before matching the policy.

Additional options can be set under `config authentication setting`.

CLI syntax - config authentication setting

```
config authentication setting
  set sso-scheme <scheme-name>
  set active-scheme <scheme-name>
  set captive-portal <host-name>
  set captive-portal-port <tcp-port>
end
```

Integration of transparent and explicit proxy HTTP policy checking

A CLI command, under `config firewall profile-protocol-options`, allows HTTP policy checking to be enable or disabled. When enabled, transparent traffic can be matched in a firewall policy and policy user authentication can occur. In addition, separate SSL inspection policies can be created:

```
config firewall profile-protocol-options
  edit <name>
    set http-policy {enable|disable}
  end
```

Internet Service Database in Explicit/Implicit proxy policies

CLI commands, under `config firewall proxy-policy`, implement the Internet Service Database (ISDB) as the webproxy matching factor, and override IP pool is also support:

```
config firewall proxy-policy
  edit <name>
    set proxy {explicit-web|transparent-web|ftp|wanopt}
    set dstintf <dst-name>
    set poolname <ip-pool-name>
  end
```

Multiple port/port range support for explicit web and explicit FTP proxy

Multiple port numbers and/or ranges can be set for explicit proxy, specifically for HTTP/HTTPS and FTP. Go to **Network > Explicit Proxy** and configure settings under **Explicit Web Proxy** and **Explicit FTP Proxy**, or under `config web-proxy explicit` in the CLI Console.

1. General configuration

1.1 Kerberos environment - Windows server setup

1. Build a Windows 2008 Platform server.
2. Enable domain configuration in windows server (dcpromo).
3. Set the domain name TEST.COM (realm name).

1.2 Create users

- `testuser` is a normal user (could be any existing domain user account).
- `testfgt` is the service name. In this case it should be the FQDN for the explicit proxy Interface, For example the hostname in the client browser proxy config.
- Recommendation: create username all in lowercase (even if against corporate standards).
 - The account only requires “domain users” membership
 - Password set to never expire
 - Set a very strong password

1.3 Add FortiGate to DNS



Add the FortiGate FQDN in to the Windows DNS domain, as well as in-addr.arpa

For Lab/Testing add the FortiGate Domain name and IP mapping in the hosts file (windows/system32/drivers/etc/hosts). e.g., TESTFGT.TEST.COM 10.10.1.10

1.4 Generate the Kerberos keytab

Use the `ktpass` command (found on Windows Servers and many domain workstations) to generate the Kerberos keytab.

Example:

```
ktpass -princ HTTP/<domain name of test fgt>@realm -mapuser testfgt -pass <password> -  
crypto all -ptype KRB5_NT_PRINCIPAL -out fgt.keytab
```



In the case where the FortiGate is handling multiple keytabs in Kerberos authentication, use different passwords when generating each keytab.



The ktpass on older Windows servers (i.e. 2003) may not support the “all” crypto option.

Example:

```
ktpass -princ HTTP/testfgt.test.com@TEST.COM -mapuser testfgt -pass 12345678 -crypto all -  
ptype KRB5_NT_PRINCIPAL -out fgt.keytab
```



The realm name is always presented in uppercase, and prefixed with the “@” character.

1.5 Encode base64

Use the `base64` command (available in most Linux distros) command to encode the `fgt.keytab` file. Any LF (Line Feed) need to be deleted from the file.

Example:

```
base64 fgt.keytab > fgt.txt
```



Use Notepad++ or some native Linux text editor. Windows Notepad and Wordpad are likely to introduce errors.

2. FortiGate configuration

2.1 Create LDAP server instance

```
config user ldap
  edit "ldap" <<< Required for authorization
    set server "10.10.1.1" <<< LDAP server IP, normally it should be same as KDC server
    set cnid "cn"
    set dn "dc=test,dc=com"
    set type regular
    set username "CN=admin,CN=Users,DC=test,DC=com" <<< Your domain may require STARTTLS
    set password <FOOS>
  next
end
```

2.2 Define Kerberos as an authentication service

```
config user krb-keytab
  edit "http_service"
    set principal "HTTP/testfgt.test.com@TEST.COM" <<< Same as the principal name in 1.4
```

```

set ldap-server "ldap" <<< the defined ldap server for authorization
set keytab
    "BQIAABNAIAACKJFUKJFUi5DT00ABEhUVFAAG1RPT11fRkdUXzEwMERfQS5CRVJCRVIuQ09NAAAAAQAA
    AAAAKABcAEJQ10MHqovwplu7XzfENJzw=" <<< base64 endoding keytab data, created in step 1.5
next
end

```

2.3 Create user group(s)

```

config user group <<< the group is used for kerberos authentication
    edit "testgrp"
        set member "ldap"
        config match
            edit 1
                set server-name "ldap" <<< Same as ldap-server option in krb-keytab
                set group-name "CN=Domain Users,CN=Users,DC=TEST,DC=com"
            next
        end
    next
end

```

2.4 Create firewall policy

```

config firewall proxy-policy
    edit 1
        set uuid 5e5dd6c4-952c-51e5-b363-120ad77c1414
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set groups "CN=USERS LAB.PS FSSO"
    next
end

```

2.5 Diagnostics

Once the keytab is imported, check that it has been properly decoded. The filename generated will be relatively random, but should be clearly visible.

```

Artoo-Deetoo (root) # fnsysctl ls -la /tmp/kt
drwxr--r-- 2 0 0 Fri Dec 2 10:06:43 2016 60 .
drwxrwxrwt 22 0 0 Tue Dec 6 14:28:29 2016 3280 ..
-rw-r--r-- 1 0 0 Fri Dec 2 10:06:43 2016 392 1.0.89.keytab

```



If there is no file present, then the file hasn't decoded. Check the file for line feeds and try again.

3. Client side walkthrough

3.1 Check Kerberos is working

Log on to the domain by using *testuser*, created in 1.2. Use the *klist* command to list ticket information. In the below example, the client has received *krbtgt*, *CIFS*, and *LDAP* tickets. As there has been no interaction with the FortiGate, there are no references to it.

```
C:\Users\glenk>klist Cached Tickets: (5)

C:\Users\glenk>klist
Cached Tickets: (5)
#0> Client: glenk @ home.local

    Server: krbtgt/HOME.LOCAL @ HOME.LOCAL
    KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
    Ticket Flags 0x60a00000 -> forwardable forwarded renewable pre_authent
    Start Time: 12/6/2016 14:58:06 (local)
    End Time: 12/7/2016 0:58:04 (local)
    Renew Time: 12/13/2016 14:58:04 (local)
    Session Key Type: AES-256-CTS-HMAC-SHA1-96

#1> Client: glenk @ home.local

    Server: krbtgt/HOME.LOCAL @ HOME.LOCAL
    KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
    Ticket Flags 0x40e00000 -> forwardable renewable initial pre_authent
    Start Time: 12/6/2016 14:58:04 (local)
    End Time: 12/7/2016 0:58:04 (local)
    Renew Time: 12/13/2016 14:58:04 (local)
    Session Key Type: AES-256-CTS-HMAC-SHA1-96

#2> Client: glenk @ home.local

    Server: cifs/EthicsGradient.home.local @ HOME.LOCAL
    KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
    Ticket Flags 0x40a40000 -> forwardable renewable pre_authent ok_as_delegate
    Start Time: 12/6/2016 14:58:06 (local)
    End Time: 12/7/2016 0:58:04 (local)
    Renew Time: 12/13/2016 14:58:04 (local)
    Session Key Type: AES-256-CTS-HMAC-SHA1-96

#3> Client: glenk @ home.local

    Server: ldap/EthicsGradient.home.local @ HOME.LOCAL
    KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
    Ticket Flags 0x40a40000 -> forwardable renewable pre_authent ok_as_delegate
    Start Time: 12/6/2016 14:58:06 (local)
    End Time: 12/7/2016 0:58:04 (local)
    Renew Time: 12/13/2016 14:58:04 (local)
    Session Key Type: AES-256-CTS-HMAC-SHA1-96

#4> Client: glenk @ home.local

    Server: LDAP/EthicsGradient.home.local/home.local @ HOME.LOCAL
    KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
    Ticket Flags 0x40a40000 -> forwardable renewable pre_authent ok_as_delegate
    Start Time: 12/6/2016 14:58:06 (local)
    End Time: 12/7/2016 0:58:04 (local)
    Renew Time: 12/13/2016 14:58:04 (local)
    Session Key Type: AES-256-CTS-HMAC-SHA1-96
```

3.2 Configure client

Set up web-proxy in browser through the FortiGate. This can be achieved via a PAC file or direct browser configuration.



Some Firefox documentation indicates that it is necessary to make manual advanced configuration changes to allow Kerberos authentication work. However, builds 48 (and possibly much earlier) require no additional configuration beyond setting of the proxy server.

3.3 Open a connection to the Internet

1. The client accesses the explicit proxy, but a *HTTP 407 Proxy Authentication Required* is returned.
2. As “Negotiate” is set, the client has knowledge of the KRBTGT, it requests a ticket from the KDC with a *krb-tgs-req* message. This includes the REALM (HOME.LOCAL) in the *req-body* section, and the provided instances SNAME and service (in this case, HTTP/artoo-deetoo.home.local).
3. The KDC responds with a next KRB-TGS-REP.

This ticket is then available on the client.

In the example below, the ticket-granted-service has issued Ticket #2.

```
#2> Client: glenk @ home.local
Server: HTTP/artoo-deetoo.home.local @ HOME.LOCAL
KerbTicket Encryption Type: RSADSI RC4-HMAC (NT)
Ticket Flags 0x40a00000 -> forwardable renewable pre_authent
Start Time: 12/6/2016 14:59:45 (local)
End Time: 12/7/2016 0:58:04 (local)
Renew Time: 12/13/2016 14:58:04 (local)
Session Key Type: RSADSI RC4-HMAC (NT)
```

4. The conversation between the client and the proxy continues, as the client responds with the Kerberos ticket in the response.

The whole process takes less than a second to complete. The user should be visible as a FSSO logon in the Web UI.

Transparent web-proxy Kerberos authentication

Transparent web-proxy allows the FortiGate to process level 7 policy matching, even when the explicit web-proxy is not enabled on the client's browser. The transparent web-proxy policy is set in proxy-policy too. The policy matching rule is the same as the explicit web-proxy.

In the firewall policy level, transparent web-proxy is regarded as a special UTM. The HTTP/HTTPS traffic matches the firewall policy first, then traffic is redirected to the web-proxy daemon. If the transparent web-proxy feature is disabled, http-policy options in profile-protocol-options is used to enable transparent web-proxy feature.

IP-based

Kerberos authentication requires the captive portal to be an FQDN address that is resolved to a local IP address. However, it becomes more complicated to setup an FQDN address in a local user deployment. Therefore you can set the `captive-portal-type` to either use an FQDN or IP address.

1. Captive portal and the captive portal port must be configured in transparent web-proxy for support of Kerberos authentication:

```

config authentication setting
    set captive-portal-type {fqdn | ip}
    set captive-portal <fqdn-name> / <ip>
    set captive-portal-port "9998"
end

```

2. Authentication rule, scheme, and krb-keytab need to be configured for Kerberos authentication (note the active-auth-method scheme referenced in the rule):

```

config authentication scheme
    edit <kerberos-scheme>
        set method negotiate
        set negotiate-ntlm <enable>
        set fssso-guest <disable>
    next
end

config authentication rule
    edit <name>
        set status <enable>
        set protocol <http>
        set srcaddr "all"
        set ip-based <enable>
        set active-auth-method <kerberos-scheme>
    next
end

config user krb-keytab
    edit <name>
        set principal "HTTP/TESTFGT.TEST.COM@TEST.COM"
        set ldap-server "ldap"
        set keytab <base64-encoding-keytab-data>
    next
end

```

3. Configure LDAP and user group used for authorization:

```

config user ldap
    edit "ldap"
        set server "10.10.1.1"
        set cnid
        srt dn
        set type <regular>
        set username "CN=admin,CN=Users,DC=test,DC=com"
        set password ENC
            aW51IAhkPMf4D+ZCKpGMU3x8Fpq0G+7uIbAvpb1bXFA5vLfgb4/oRBx+B6R/v+CMCetP84e+Gdz5zEcM
            y0d3cj0BoIhFrpYJfxRs41SEOHezeVXfxwTSf5VJG+F11G/G5RpAY+AE8bortC8MBe7P2/uGQocFHu4
            Ilulp5160Jvyk6Ei3hDZMjTd8iPp5IkRJZVVjQ==
    next
end

config user group
    edit "testgrp"
        set membeber "ldap"
        config match
            edit "1"
                set server-name "ldap"

```

```

        set group-name "CN=Domain Users,CN=Users,DC=TEST,DC=com"
    next
end
next
end

```

4. Create proxy-policy, with groups as the authorizing policy-matching element:

```

config firewall proxy-policy
edit 1
    set uuid 1bbb891a-9cd2-51e7-42ff-d1fa13cac3da
    set proxy explicit-web
    set dstintf "any"
    set srcaddr "all"
    set dstaddr "all"
    set service "webproxy"
    set action accept
    set schedule "always"
    set groups testgrp
next
end

```

5. UTM must be enabled in the firewall policy to support the transparent web-proxy:

```

config firewall policy
edit "1"
    set name "policy1"
    set uuid 8a6ceeac-b016-51e6-2b5c-165070d5bf50
    set srcintf "mgmt1"
    set dstintf "mgmt1"
    set srcaddr "all"
    set dstaddr "all"
    set action <accept>
    set schedule "always"
    set service "ALL"
    set utm-status <enable>
    set profile-protocol-options "transparent-web-proxy"
    set ssl-ssh-profile "deep-inspection"
    set nat <enable>
next
end

config firewall profile-protocol-options
edit "transparent-web-proxy"
config http
    set ports "80 8080"
    unset options
    set http-policy enable
    unset post-lang
end
...
next
end

```

Session-based with web-auth cookie

The web-auth-cookie feature is necessary for session-based authentication under transparent web-proxy.

The configuration is the same as for IP-based authentication, except `ip-based` is disabled in the authentication rule:

```
config authentication rule
  edit "kerberos-rules"
    set status <enable>
    set protocol <http>
    set srcaddr "all"
    set ip-based <disable>
    set active-auth-method <kerberos-scheme>
  next

config authentication setting
  set captive-portal <fqdn-name>
  set captive-portal-port "9998"
end
```

Transparent proxy concepts

In addition to the Explicit Web Proxy, FortiOS supports a Transparent web proxy. While it does not have as many features as Explicit Web Proxy, the transparent proxy has the advantage that nothing needs to be done on the user's system to forward supported web traffic over to the proxy. There is no need to reconfigure the browser or publish a PAC file. Everything is transparent to the end user, hence the name. This makes it easier to incorporate new users into a proxy deployment.

You can use the transparent proxy to apply web authentication to HTTP traffic accepted by a firewall policy. In previous versions of FortiOS, web authentication required using the explicit proxy.

Normal FortiOS authentication is IP address based. Users are authenticated according to their IP address and access is allowed or denied based on this IP address. On networks where authentication based on IP address will not work you can use the Transparent Web proxy to apply web authentication that is based on the user's browser and not on their IP address. This authentication method allows you to identify individual users even if multiple users on your network are connecting to the FortiGate from the same IP address.

More about the transparent proxy

The following changes are incorporated into Transparent proxy, some of which affect Explicit Web Proxy as well.

Flat policies

The split policy feature has been removed. This will make the explicit policy more like the firewall policy.

Authentication

The authentication design is intended to separate authentication from authorization. Authentication has been moved into a new table in the FortiOS. This leaves the authorization as the domain of the explicit proxy policy.

Previously, if authentication was to be used:

1. The policy would be classified as an identity based policy
2. The policy would be split to add the authentication parameters
3. The authentication method would be selected
4. The user/group would be configured

Now:

The user/group is configured in the proxy policy

1. A new authentication rule is added
2. This option refers to the authentication scheme
3. The authentication scheme has the details of the authentication method

The new authentication work flow for transparent proxy:

Toggle the transparent-http-policy match:

```
config firewall profile-protocol-options
  edit <profile ID>
    config http
      set http-policy <enable|disable>
```

If disabled, everything works like before. If enabled, the authentication is triggered differently.

- http-policy work flow:
- For transparent traffic, if there is a regular firewall policy match, when the Layer 7 check option is enabled, traffic will be redirected to WAD for further processing.
- For redirected traffic, layer 7 policy (HTTP policy) will be used to determine how to do security checks.
- If the last matching factor is down to user ID, then it will trigger a new module to handle the L7 policy user authentication.
- Then propagate learned user information back to the system so that it can be used to match traffic for L4 policy.

New proxy type

There is a new subcategory of proxy in the proxy policy called **Transparent Web**. The old **Web Proxy** is now referred to as **Explicit Web Proxy**.

- This is set in the firewall policy
- It is available when the HTTP policy is enabled in the profile-protocol options for the firewall policy
- This proxy type supports OSI layer 7 address matching.
- This proxy type should include a source address as a parameter
- Limitations:
 - It can be used for HTTPS traffic, if deep scanning is not used
 - It only supports SNI address matching, i.e. domain names
 - It does not support header types of address matching
 - It only supports SSO authentication methods, no active authentication methods.

IP pools support

Proxies are now supported on outgoing IP pools.

SOCKSv5

SOCKSv5 authentication is now supported for explicit proxies.

To configure:

```
config authentication rule
  edit <name of rule>
    set protocol socks
  end
```

Forwarding

Proxies support URL redirect/forwarding. This allows a non-proxy forwarding server to be assigned a rule that will redirect web traffic from one URL to another, such as redirecting traffic destined for youtube.com to restrict.youtube.com.

- A new option called "Redirect URL" has been added to the policy
- Traffic forwarding by VIP is supported

Support for explicit proxy address objects & groups into IPv4 firewall policies

This would allow the selection of web filter policy, SSL inspection policy, and proxy policy based on source IP + destination (address|explicit proxy object|category|group of any of those). This enables things like "do full SSL interception on www.google.com, but not the rest of the Search Engines category".

Support application service in the proxy based on HTTP requests.

The application service can be configured using the following CLI commands:

```
config firewall service custom
  edit <name of service>
    set explicit-proxy enable
    set app-service-type <disable|app-id|app-category>
    set app-category <application category ID, integer>
    set application <application ID, integer>
  end
```

Transparent proxy configuration

To implement the Transparent proxy, go to **System > Settings** and scroll down to **Operations Settings** and set the inspection mode to **Proxy**.

Then go to **System > Feature Visibility** and enable **Explicit Proxy**.

Security Features

Feature Set: Custom ▾

<input checked="" type="checkbox"/> Anti-Spam Filter	<input type="button" value="+"/>
<input checked="" type="checkbox"/> AntiVirus	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Application Control	<input type="button" value="+"/>
<input checked="" type="checkbox"/> DLP	<input type="button" value="+"/>
<input checked="" type="checkbox"/> DNS Filter	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Endpoint Control	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Explicit Proxy	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Intrusion Prevention	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Web Application Firewall	<input type="button" value="+"/>
<input checked="" type="checkbox"/> Web Filter	<input type="button" value="+"/>

Then go to **Security Profiles > Proxy Options**, edit a proxy options profile and under **Web Options** enable **HTTP Policy Redirect**.

Web Options

- Chunked Bypass
- Add Fortinet Bar
- HTTP Policy Redirect

Then go to **Policy & Objects > IPv4 Policy** and create or edit a policy that accepts traffic that you want to apply web authentication to. This can be a general policy that accepts many different types of traffic as long as it also accepts the web traffic that you want to apply web authentication to.

Select a **Security Profile** and select the **Proxy Options** profile that you enabled **HTTP Policy Redirect** for.

Name	General Internet Access Policy		
Incoming Interface	port2	▼	
Outgoing Interface	port1	▼	
Source	all	×	
Destination	all	×	
Schedule	always	▼	
Service	ALL	×	
Action	<input checked="" type="checkbox"/> ACCEPT <input type="checkbox"/> DENY <input type="checkbox"/> LEARN		

Firewall / Network Options

NAT	<input checked="" type="checkbox"/>
IP Pool Configuration	<input checked="" type="radio"/> Use Outgoing Interface Address <input type="radio"/> Use Dynamic IP Pool

Security Profiles

AntiVirus	<input checked="" type="checkbox"/>	AV default	▼	<input type="button" value="edit"/>
Web Filter	<input type="checkbox"/>			
DNS Filter	<input type="checkbox"/>			
Application Control	<input type="checkbox"/>			
IPS	<input type="checkbox"/>			
Proxy Options	<input checked="" type="checkbox"/>	PRX default	▼	<input type="button" value="edit"/>
SSL/SSH Inspection	<input checked="" type="checkbox"/>	SSL certificate-inspection	▼	<input type="button" value="edit"/>

Then go to **Policy & Objects > Proxy Policy** create a Transparent Proxy policy to accept the traffic that you want to apply web authentication to. Set the **Proxy Type** to **Transparent Web**. The incoming interface, outgoing interface, destination address, and schedule should either match or be a subset of the same options defined in the IPv4 policy. Addresses added to the Source must match or be a subset of the source addresses added to the IPv4 policy. You can also add the users to be authenticated by the transparent policy to the source field.

Select other transparent policy options as required.

Proxy Type 	<input type="radio"/> Explicit Web	<input checked="" type="radio"/> Transparent Web	<input type="radio"/> FTP
Incoming Interface	 port2		
Outgoing Interface	 port1		
Source	 web_users 		
Destination Address	 all 		
Schedule	 always 		
Action	<input checked="" type="radio"/> ACCEPT <input type="radio"/>  DENY		

Disclaimer Options

Display Disclaimer	<input type="radio"/> Disable	<input type="radio"/> By Domain	<input checked="" type="radio"/> By Policy	<input type="radio"/> By User
Customize Messages		 Edit Disclaimer Message		

Security Profiles

- AntiVirus 
- Web Filter 
- Application Control 
- IPS 

Web Proxy Forwarding Server 

Logging Options

Log Allowed Traffic 	<input checked="" type="radio"/> Security Events	<input type="radio"/> All Sessions
Comments	<input type="text" value="Write a comment..."/> <div style="display: flex; justify-content: space-between; width: 100%;"> 0/1023 </div>	

CLI changes due to addition of transparent proxy

The adding of Transparent Proxy to the existing proxy types has required some changes, removals, moves and additions to the CLI.

Changes:

Previous	New
config firewall explicit-proxy-policy	config firewall proxy-policy

Previous	New
config firewall explicit-proxy-address	config firewall proxy-address
config firewall explicit-proxy-addrgrp	config firewall proxy-addrgrp
config firewall explicit-proxy-policy edit <policy ID> set proxy web end	config firewall proxy-policy edit <policy ID> set proxy explicit-web end

FTP proxy concepts

The FortiGate explicit FTP proxy

You can use the FortiGate explicit FTP proxy to enable explicit FTP proxying on one or more FortiGate interfaces. The explicit web and FTP proxies can be operating at the same time on the same or on different FortiGate interfaces.



Explicit FTP proxies are configured for each VDOM when multiple VDOMs are enabled.

In most cases you would configure the explicit FTP proxy for users on a network by enabling the explicit FTP proxy on the FortiGate interface connected to that network. Users on the network would connect to and authenticate with the explicit FTP proxy before connecting to an FTP server. In this case the IP address of the explicit FTP proxy is the IP address of the FortiGate interface on which the explicit FTP proxy is enabled.

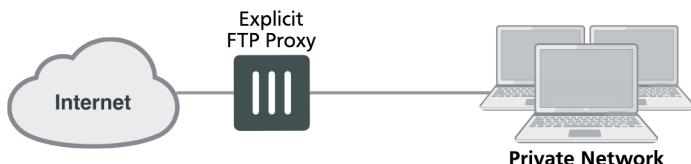


Enabling the explicit FTP proxy on an interface connected to the Internet is a security risk because anyone on the Internet who finds the proxy could use it to hide their source address.

If the FortiGate unit is operating in transparent mode, users would configure their browsers to use a proxy server with the FortiGate unit management IP address.

The FTP proxy receives FTP sessions to be proxied at FortiGate interfaces with the explicit FTP proxy enabled. The FTP proxy uses FortiGate routing to route sessions through the FortiGate unit to a destination interface. Before a session leaves the exiting interface, the explicit FTP proxy changes the source addresses of the session packets to the IP address of the exiting interface. When the FortiGate unit is operating in transparent mode the explicit web proxy changes the source addresses to the management IP address.

Example explicit FTP proxy topology



To allow anyone to anonymously log into explicit FTP proxy and connect to any FTP server you can set the explicit FTP proxy default firewall proxy action to accept. When you do this, users can log into the explicit FTP proxy with any username and password.

In most cases you would want to use explicit proxy policies to control explicit FTP proxy traffic and apply security features, access control/authentication, and logging. You can do this by keeping the default explicit FTP proxy firewall policy action to deny and then adding explicit FTP proxy policies. In most cases you would also want users to authenticate with the explicit FTP proxy. By default an anonymous FTP login is required. Usually you would

add authentication to explicit FTP proxy policies. Users can then authenticate with the explicit FTP proxy according to users or user groups added to the policies. User groups added to explicit FTP proxy policies can use any authentication method supported by FortiOS including the local user database and RADIUS and other remote servers.

If you leave the default firewall policy action set to deny and add explicit FTP proxy policies, all connections to the explicit FTP proxy must match an or else they will be dropped. Sessions that are accepted are processed according to the ftp-proxy security policy settings.

You can also change the explicit FTP proxy default firewall policy action to accept and add explicit FTP proxy policies. If you do this, sessions that match explicit FTP proxy policies are processed according to the policy settings. Connections to the explicit FTP proxy that do not match an explicit FTP proxy policy are allowed and the users can authenticate with the proxy anonymously.

There are some limitations to the security features that can be applied to explicit FTP proxy sessions. See [The FortiGate explicit FTP proxy on page 411](#).

You cannot configure IPsec, SSL VPN, or Traffic shaping for explicit FTP proxy traffic. Explicit FTP proxy policies can only include firewall addresses not assigned to a FortiGate unit interface or with interface set to **any**. (On the web-based manager you must set the interface to **Any**. In the CLI you must `unset the associated-interface`.)

How to use the explicit FTP proxy to connect to an FTP server

To connect to an FTP server using the explicit FTP proxy, users must run an FTP client and connect to the IP address of a FortiGate interface on which the explicit FTP proxy is enabled. This connection attempt must use the configured explicit FTP proxy port number (default 21).

The explicit FTP proxy is not compatible with using a web browser as an FTP client. To use web browsers as FTP clients configure the explicit web proxy to accept FTP sessions.

The following steps occur when a user starts an FTP client to connect to an FTP server using the explicit FTP proxy. Any RFC-compliant FTP client can be used. This example describes using a command-line FTP client. Some FTP clients may require a custom FTP proxy connection script.

1. The user enters a command on the FTP client to connect to the explicit FTP proxy.

For example, if the IP address of the FortiGate interface on which the explicit FTP proxy is enabled is 10.31.101.100, enter:

```
ftp 10.31.101.100
```

2. The explicit FTP proxy responds with a welcome message and requests the user's FTP proxy user name and password and a username and address of the FTP server to connect to:

```
Connected to 10.31.101.100.  
220 Welcome to FortiGate FTP proxy  
Name (10.31.101.100:user):
```

You can change the message by editing the FTP Explicit Banner Message replacement message.

3. At the prompt the user enters their FTP proxy username and password and a username and address for the FTP server. The FTP server address can be a domain name or numeric IP address. This information is entered using the following syntax:

```
<proxy-user>:<proxy-password>:<server-user>@<server-address>
```

For example, if the proxy username and password are `p-name` and `p-pass` and a valid username for

the FTP server is `s-name` and the server's IP address is `ftp.example.com` the syntax would be:

`p-name:p-pass:s-name@ftp.example.com`

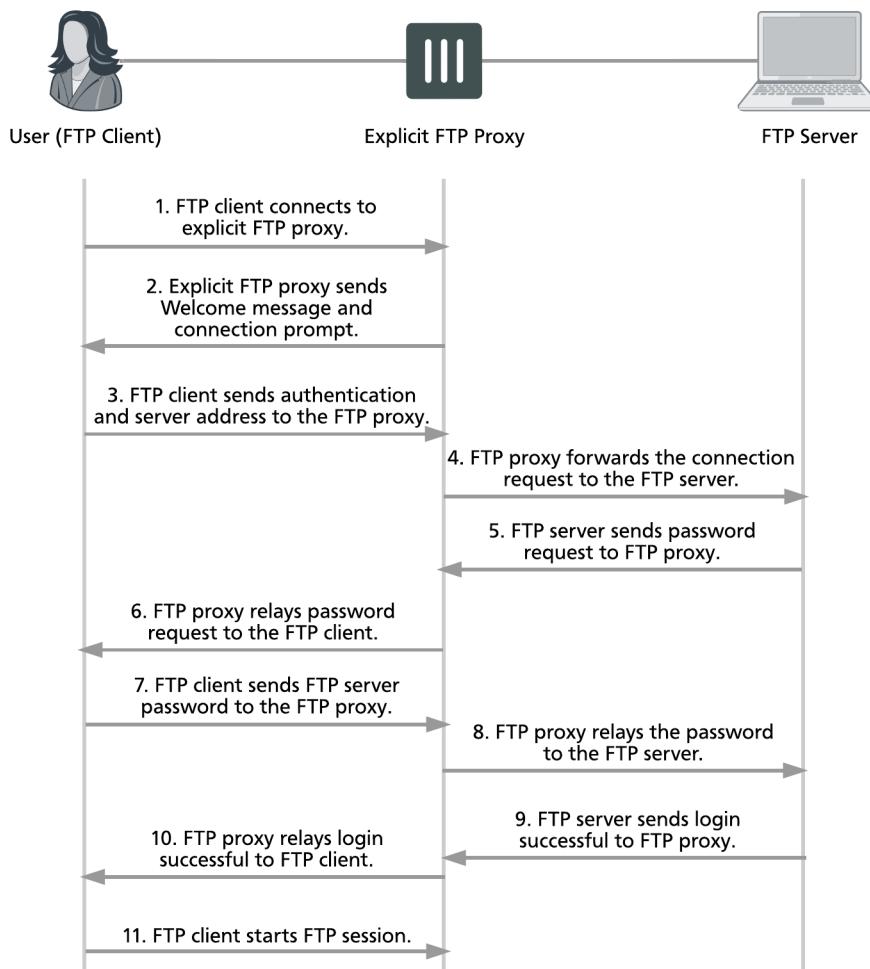


If the FTP proxy accepts anonymous logins `p-name` and `p-pass` can be any characters.

4. The FTP proxy forwards the connection request, including the user name, to the FTP server.
5. If the user name is valid for the FTP server it responds with a password request prompt.
6. The FTP proxy relays the password request to the FTP client.
7. The user enters the FTP server password and the client sends the password to the FTP proxy.
8. The FTP proxy relays the password to the FTP server.
9. The FTP server sends a login successful message to the FTP proxy.
10. The FTP proxy relays the login successful message to the FTP client.
11. The FTP client starts the FTP session.

All commands entered by the client are relayed by the proxy to the server. Replies from the server are relayed back to the FTP client.

Explicit FTP proxy session



From a simple command line FTP client connecting to an the previous sequence could appear as follows:

```

ftp 10.31.101.100 21
Connected to 10.31.101.100.
220 Welcome to FortiGate FTP proxy
Name (10.31.101.100:user): p-name:p-pass:s-name@ftp.example.com
331 Please specify the password.
Password: s-pass
230 Login successful.
Remote system type is UNIX
Using binary mode to transfer files.
ftp>

```

Security profiles, threat weight, device identification, and the explicit FTP proxy

You can apply antivirus, data leak prevention (DLP), and SSL/SSH inspection to explicit FTP proxy sessions. Security profiles are applied by selecting them in an explicit FTP proxy policy or an authentication rule in an FTP

proxy security policy.

Traffic accepted by explicit FTP proxy policies contributes to threat weight data.

The explicit FTP proxy is not compatible with device identification.

Explicit FTP proxy options and SSL/SSH inspection

Since the traffic accepted by the explicit FTP proxy is known to be FTP and since the ports are already known by the proxy, the explicit FTP proxy does not use the FTP port proxy options settings.

When adding UTM features to an FTP proxy security policy, you must select a proxy options profile. In most cases you can select the default proxy options profile. You could also create a custom proxy options profile.

The explicit FTP proxy supports the following proxy options:

- Block Oversized File and oversized file limit

The explicit FTP proxy does not support the following protocol options:

- Client comforting

Explicit FTP proxy sessions and antivirus

For explicit FTP proxy sessions, the FortiGate unit applies antivirus scanning to FTP file GET and PUT requests. The FortiGate unit starts virus scanning a file in an FTP session when it receives a file in the body of an FTP request.

Flow-based virus scanning is not available for explicit FTP proxy sessions. Even if the FortiGate unit is configured to use flow-based antivirus, explicit FTP proxy sessions use the regular virus database.

Explicit FTP proxy sessions and user limits

FTP clients do not open large numbers of sessions with the explicit FTP proxy. Most sessions stay open for a short while depending on how long a user is connected to an FTP server and how large the file uploads or downloads are. So unless you have large numbers of FTP users, the explicit FTP proxy should not be adding large numbers of sessions to the session table.

Explicit FTP proxy sessions and user limits are combined with explicit web proxy session and user limits. For information about explicit proxy session and user limits, see [Explicit proxy sessions and user limits on page 1](#).

FTP proxy configuration

General explicit FTP proxy configuration steps

You can use the following general steps to configure the explicit FTP proxy.

To enable the explicit FTP proxy - web-based manager:

1. Go to **Network > Explicit Proxy > Explicit FTP Proxy Options**. Select **Enable Explicit FTP Proxy** to turn on the explicit FTP proxy.
2. Select Apply.

The **Default Firewall Policy Action** is set to **Deny** and requires you to add a explicit FTP proxy policy to allow access to the explicit FTP proxy. This configuration is recommended and is a best practice because you can use policies to control access to the explicit FTP proxy and also apply security features and authentication.

3. Go to **Network > Interfaces** and select one or more interfaces for which to enable the explicit web proxy. Edit the interface and select **Enable Explicit FTP Proxy**.



Enabling the explicit FTP proxy on an interface connected to the Internet is a security risk because anyone on the Internet who finds the proxy could use it to hide their source address. If you enable the proxy on such an interface make sure authentication is required to use the proxy.

4. Go to **Policy & Objects > Proxy Policy** and select **Create New** and set the **Explicit Proxy Type** to **FTP**.

You can add multiple explicit FTP proxy policies.

5. Configure the policy as required to accept the traffic that you want to be processed by the explicit FTP proxy.

The source address of the policy should match client source IP addresses. The firewall address selected as the source address cannot be assigned to a FortiGate interface. The Interface field of the firewall address must be blank or it must be set to **Any**.

The destination address of the policy should match the IP addresses of FTP servers that clients are connecting to. The destination address could be **all** to allow connections to any FTP server.

If **Default Firewall Policy Action** is set to Deny, traffic sent to the explicit FTP proxy that is not accepted by an explicit FTP proxy policy is dropped. If **Default Firewall Policy Action** is set to Allow then all FTP proxy sessions that don't match a policy are allowed.

For example the following explicit FTP proxy policy allows users on an internal network to access FTP servers on the Internet through the wan1 interface of a FortiGate unit.

Explicit Proxy Type	FTP
----------------------------	-----

Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	all
Schedule	always
Action	ACCEPT

The following explicit FTP proxy policy requires users on an internal network to authenticate with the FortiGate unit before accessing FTP servers on the Internet through the wan1 interface.

Explicit Proxy Type	FTP
Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	all
Action	AUTHENTICATE

6. Select **Create New** to add an **Authentication Rule** and configure the rule as follows:

Groups	Proxy-Group
Source Users	(optional)
Schedule	always

7. Add security profiles as required and select **OK**.
8. You can add multiple authentication rules to apply different authentication for different user groups and users and also apply different security profiles and logging settings for different users.
9. Select **OK**.

To enable the explicit FTP proxy - CLI:

1. Enter the following command to turn on the explicit FTP proxy. This command also changes the explicit FTP proxy port to 2121.

```
config ftp-proxy explicit
  set status enable
  set incoming-port 2121
end
```

The default explicit FTP proxy configuration has `sec-default-action` set to `deny` and requires you to add a security policy to allow access to the explicit FTP proxy.

2. Enter the following command to enable the explicit FTP proxy for the internal interface.

```
config system interface
  edit internal
    set explicit-ftp-proxy enable
  end
end
```

3. Use the following command to add a firewall address that matches the source address of users who connect to the explicit FTP proxy.

```
config firewall address
  edit Internal_subnet
    set type iprange
    set start-ip 10.31.101.1
    set end-ip 10.31.101.255
  end
```

The source address for a ftp-proxy security policy cannot be assigned to a FortiGate unit interface.

4. Use the following command to add an explicit FTP proxy policy that allows all users on the internal subnet to use the explicit FTP proxy for connections through the wan1 interface to the Internet.

```
config firewall proxy-policy
  edit 0
    set proxy ftp
    set dstintf wan1
    set scraddr Internal_subnet
    set dstaddr all
    set action accept
    set schedule always
  end
```

5. Use the following command to add an explicit FTP proxy policy that allows authenticated users on the internal subnet to use the explicit FTP proxy for connections through the wan1 interface to the Internet.

```
config firewall proxy-policy
  edit 0
    set proxy ftp
    set dstintf wan1
    set scraddr Internal_subnet
    set dstaddr Fortinet-web-sites
    set action accept
    set schedule always
    set groups <User group>
  end
```

Restricting the IP address of the explicit FTP proxy

You can use the following command to restrict access to the explicit FTP proxy using only one IP address. The IP address that you specify must be the IP address of an interface that the explicit FTP proxy is enabled on. You might want to use this option if the explicit FTP proxy is enabled on an interface with multiple IP addresses.

For example, to require users to connect to the IP address 10.31.101.100 to connect to the explicit FTP proxy:

```
config ftp-proxy explicit
  set incoming-ip 10.31.101.100
end
```

Restricting the outgoing source IP address of the explicit FTP proxy

You can use the following command to restrict the source address of outgoing FTP proxy packets to a single IP address. The IP address that you specify must be the IP address of an interface that the explicit FTP proxy is enabled on. You might want to use this option if the explicit FTP proxy is enabled on an interface with multiple IP addresses.

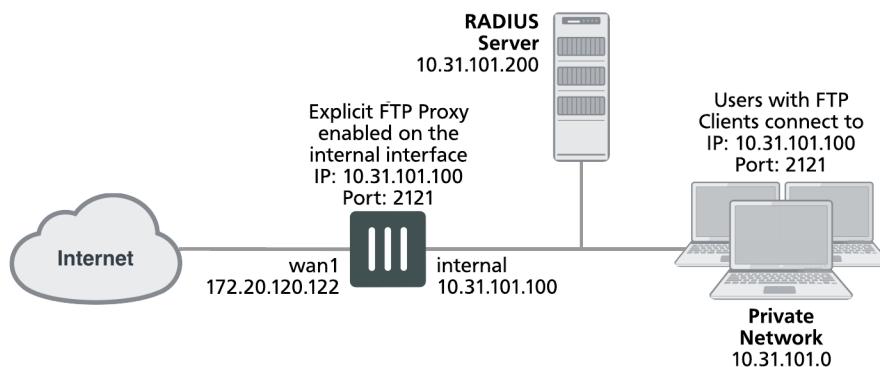
For example, to restrict the outgoing packet source address to 172.20.120.100:

```
config ftp-proxy explicit
    set outgoing-ip 172.20.120.100
end
```

Example users on an internal network connecting to FTP servers on the Internet through the explicit FTP with RADIUS authentication and virus scanning

This example describes how to configure the explicit FTP proxy for the example network shown below. In this example, users on the internal network connect to the explicit FTP proxy through the Internal interface with IP address 10.31.101.100. The explicit web proxy is configured to use port 2121 so to connect to an FTP server on the Internet users must first connect to the explicit FTP proxy using IP address 10.31.101.100 and port 2121.

Example explicit FTP proxy network topology



In this example, explicit FTP proxy users must authenticate with a RADIUS server before getting access to the proxy. To apply authentication, the security policy that accepts explicit FTP proxy traffic includes an identity based policy that applies per session authentication to explicit FTP proxy users and includes a user group with the RADIUS server in it. The identity based policy also applies UTM virus scanning and DLP.

General configuration steps

This section breaks down the configuration for this example into smaller procedures. For best results, follow the procedures in the order given:

1. Enable the explicit FTP proxy and change the FTP port to 2121.
2. Enable the explicit FTP proxy on the internal interface.

3. Add a RADIUS server and user group for the explicit FTP proxy.
4. Add a user identity security policy for the explicit FTP proxy.
5. Enable antivirus and DLP features for the identity-based policy.

Configuring the explicit FTP proxy - web-based manager

Use the following steps to configure the explicit FTP proxy from FortiGate web-based manager.

To enable and configure the explicit FTP proxy

1. Go to **Network > Explicit Proxy > Explicit FTP Proxy Options** and change the following settings:

Enable Explicit FTP Proxy	Select.
Listen on Interface	No change. This field will eventually show that the explicit web proxy is enabled for the Internal interface.
FTP Port	2121
Default Firewall Policy Action	Deny

2. Select **Apply**.

To enable the explicit FTP proxy on the Internal interface

1. Go to **Network > Interfaces**, edit the Internal interface and select **Enable Explicit FTP Proxy**.

To add a RADIUS server and user group for the explicit FTP proxy

1. Go to **User & Device > RADIUS Servers**.
2. Select **Create New** to add a new RADIUS server:

Name	RADIUS_1
Primary Server Name/IP	10.31.101.200
Primary Server Secret	RADIUS_server_secret

3. Go to **User > User > User Groups** and select **Create New**.

Name	Explict_proxy_user_group
Type	Firewall
Remote groups	RADIUS_1
Group Name	ANY

4. Select **OK**.

To add a security policy for the explicit FTP proxy

1. Go to **Policy & Objects > Addresses** and select **Create New**.
2. Add a firewall address for the internal network:

Address Name	Internal_subnet
Type	Subnet
Subnet / IP Range	10.31.101.0
Interface	Any

3. Go to **Policy & Objects > Proxy Policy** and select **Create New**.
4. Configure the explicit FTP proxy security policy.

Explicit Proxy Type	FTP
Source Address	Internal_subnet
Outgoing Interface	wan1
Destination Address	all
Action	AUTHENTICATE

5. Under **Configure Authentication Rules** select **Create New** to add an authentication rule:

Groups	Explicit_policy
Users	Leave blank
Schedule	always

6. Turn on **Antivirus** and **Web Filter** and select the **default** profiles for both.
7. Select the **default** proxy options profile.
8. Select **OK**.
9. Make sure **Enable IP Based Authentication** is not selected and **DefaultAuthentication Method** is set to **Basic**.
10. Select **OK**.

Configuring the explicit FTP proxy - CLI

Use the following steps to configure the example explicit web proxy configuration from the CLI.

To enable and configure the explicit FTP proxy

1. Enter the following command to enable the explicit FTP proxy and set the TCP port that proxy accepts FTP connections on to 2121.

```
config ftp-proxy explicit
  set status enable
  set incoming-port 2121
```

```
    set sec-default-action deny
end
```

To enable the explicit FTP proxy on the Internal interface

1. Enter the following command to enable the explicit FTP proxy on the internal interface.

```
config system interface
edit internal
    set explicit-ftp-proxy enable
end
```

To add a RADIUS server and user group for the explicit FTP proxy

1. Enter the following command to add a RADIUS server:

```
config user radius
edit RADIUS_1
    set server 10.31.101.200
    set secret RADIUS_server_secret
end
```

2. Enter the following command to add a user group for the RADIUS server.

```
config user group
edit Explicit_proxy_user_group
    set group-type firewall
    set member RADIUS_1
end
```

To add a security policy for the explicit FTP proxy

1. Enter the following command to add a firewall address for the internal subnet:

```
config firewall address
edit Internal_subnet
    set type iprange
    set start-ip 10.31.101.1
    set end-ip 10.31.101.255
end
```

2. Enter the following command to add the explicit FTP proxy security policy:

```
config firewall proxy-policy
edit 0
    set proxy ftp
    set dstintf wan1
    set srcaddr Internal_subnet
    set dstaddr all
    set action accept
    set identity-based enable
    set ipbased disable
    set active-auth-method basic
    set groups <User group>
end
```

Testing and troubleshooting the configuration

You can use the following steps to verify that the explicit FTP proxy configuration is working as expected. These steps use a command line FTP client.

To test the explicit web proxy configuration

1. From a system on the internal network start an FTP client and enter the following command to connect to the FTP proxy:

```
ftp 10.31.101.100
```

The explicit FTP proxy should respond with a message similar to the following:

```
Connected to 10.31.101.100.  
220 Welcome to Floodgate FTP proxy  
Name (10.31.101.100:user):
```

2. At the prompt enter a valid username and password for the RADIUS server followed by a user name for an FTP server on the Internet and the address of the FTP server. For example, if a valid username and password on the RADIUS server is ex_name and ex_pass and you attempt to connect to an FTP server at ftp.example.com with user name s_name, enter the following at the prompt:

```
Name (10.31.101.100:user):ex_name:ex_pass:s_name@ftp.example.com
```

3. You should be prompted for the password for the account on the FTP server.
4. Enter the password and you should be able to connect to the FTP server.
5. Attempt to explore the FTP server file system and download or upload files.
6. To test UTM functionality, attempt to upload or download an ECAR test file. Or upload or download a text file containing text that would be matched by the DLP sensor.

For eicar test files, go to <http://eicar.org>.

Diagnose commands for WAN optimization

The following get and diagnose commands are available for troubleshooting WAN optimization, web cache, explicit proxy and WCCP.

get test {wad | wccpd} <test_level>

Display usage information about WAN optimization, explicit proxy, web cache, and WCCP applications. Use <test_level> to display different information.

```
get test wad <test_level>
get test wccpd <test_level>
```

Variable	Description
wad	Display information about WAN optimization, web caching, the explicit web proxy, and the explicit FTP proxy.
wccpd	Display information about the WCCP application.

Examples

Enter the following command to display WAN optimization tunnel protocol statistics. The http tunnel and tcp tunnel parts of the command output below shows that WAN optimization has been processing HTTP and TCP packets.

```
get test wad 1
WAD manager process status: pid=113 n_workers=1 ndebug_workers=0
```

Enter the following command to display all test options:

```
get test wad

WAD process 82 test usage:
 1: display process status
 2: display total memory usage.
 99: restart all WAD processes
 1000: List all WAD processes.
 1001: dispaly debug level name and values
 1002: dispaly status of WANOpt storages
 1068: Enable debug for all WAD workers.
 1069: Disable debug for all WAD workers.
 2yxx: Set No. xx process of type y as diagnosis process.
 3: display all fix-sized advanced memory stats
 4: display all fix-sized advanced memory stats in details
 500000..599999: cmem bucket stats (599999 for usage)
 800..899: mem_diag commands (800 for help & usage)
 800000..899999: mem_diag commands with 1 arg (800 for help & usage)
 8000000..89999999: mem_diag commands with 2 args (800 for help & usage)
 60: show debug stats.
```

```

61: discard all wad debug info that is currently pending
62xxx: set xxxM maximum ouput buffer size for WAD debug. 0, set back to default.
68: Enable process debug
69: Disable process debug
98: gracefully stopping WAD process
9xx: Set xx workers (0: default based on user configuration.)

```

diagnose wad

Display diagnostic information about the WAN optimization daemon (wad).

```

diagnose wad console-log {disable | enable}
diagnose wad debug-url {disable | enable}
diagnose wad filter {clear | dport | dst | list | negate | protocol | sport | src | vd}
diagnose wad history {clear | list}
diagnose wad session {clear | list}
diagnose wad stats {cache | cifs | clear | crypto | ftp | http | list | mapi | mem |
    scan | scripts | summary | tcp | tunnel}
diagnose wad user {clear | list}
diagnose wad tunnel {clear | list}
diagnose wad webcache {clear | list} {10min | hour | day | 30days}

```

Variable	Description
console-log	Enable or disable displaying WAN optimization log messages on the CLI console.
filter	<p>Set a filter for listing WAN optimization daemon sessions or tunnels.</p> <p>clear reset or clear the current log filter settings.</p>
history	Display statistics for one or more WAN optimization protocols for a specified period of time (the last 10 minutes, hour, day or 30 days).
session	Display diagnostics for WAN optimization sessions or clear active sessions.
stats	Display statistics for various parts of WAN optimization such as cache statistics, CIFS statistics, MAPI statistics, HTTP statistics, tunnel statistics etc. You can also clear WAN optimization statistics and display a summary.
tunnel	Display diagnostic information for one or all active WAN optimization tunnels. Clear all active tunnels. Clear all active tunnels.
webcache	Display web cache activity for the specified time period.

Example diagnose wad tunnel list

Enter the following command to list all of the running WAN optimization tunnels and display information about each one. The command output shows 10 tunnels all created by peer-to-peer WAN optimization rules (auto-detect set to off).

```
diagnose wad tunnel list

Tunnel: id=100 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=100 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=348 bytes_out=384

Tunnel: id=99 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=99 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=348 bytes_out=384

Tunnel: id=98 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=98 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=348 bytes_out=384

Tunnel: id=39 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=39 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1068 bytes_out=1104

Tunnel: id=7 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=7 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=8 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=8 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=5 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=5 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=4 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=4 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264
```

```

Tunnel: id=1 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=1 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnel: id=2 type=manual
  vd=0 shared=no uses=0 state=3
  peer name=Web_servers id=2 ip=172.20.120.141
  SSL-secured-tunnel=no auth-grp=
  bytes_in=1228 bytes_out=1264

Tunnels total=10 manual=10 auto=0

```

Example diagnose wad webcache list

This following command displays the web caching stats for the last 10 minutes of activity. The information displayed is divided into 20 slots and each slot contains stats for 30 seconds:

20 * 30 seconds = 600 seconds = 10 minutes

```

diagnose wad webcache list 10min
web cache history vd=0 period=last 10min

```

The first 20 slots are for HTTP requests in the last 10 minutes. Each slot of stats has four numbers, which is the total number of HTTP requests, the number of cacheable HTTP requests, the number of HTTP requests that are processed by the web cache (hits), and the number of HTTP requests that are processed without checking the web cache (bypass). There are many reasons that a HTTP request may bypass web cache.

total	cacheable	hits	bypass
36	10	3	1
128	92	1	10
168	97	2	3
79	56	0	3
106	64	5	3
180	118	6	11
88	53	7	3
80	43	4	4
107	44	9	2
84	12	0	2
228	139	52	10
32	2	0	5
191	88	13	7
135	25	40	3
48	10	0	8
193	13	7	7
67	31	1	2
109	35	24	6
117	36	10	5
22	0	0	4

The following slots are for video requests in the last 10 minutes. Each slot has two numbers for each 30 seconds: total number of video requests, and the number of video requests that are processing using cached data.

video	total	video	hit
0	0	0	
0	0	0	
0	0	0	
0	0	0	
0	0	0	
0	0	0	
0	0	0	
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 following 20 slots are for traffic details in last 10 minutes. Each slot has four numbers for 30 seconds each.

--- LAN ---		--- WAN ---	
bytes_in	bytes_out	bytes_in	bytes_out
34360	150261	141086	32347
105408	861863	858501	100670
128359	1365919	1411849	127341
60103	602813	818075	59967
105867	1213192	1463736	97489
154961	1434784	1344911	158667
73967	370275	369847	70626
129327	602834	592399	123676
115719	663446	799445	111262
58151	724993	631721	59989
175681	2092925	1092556	166212
37805	33042	41528	37779
183686	1255118	1114646	172371
106125	904178	807152	81520
66147	473983	543507	66782
170451	1289530	1201639	165540
69196	544559	865370	68446
134142	579605	821430	132113
96895	668037	730633	89872
59576	248734	164002	59448

diagnose wad csvc

The `diagnose wad csvc` command refers to the `cache-service`. The next options to the command are listed in the table. Some will have their own sub options for refining the output or results.

Option	Description
memory	Cache service memory diagnostics
webcache	Webcache diagnostics
bytecache	Bytecache diagnostics
memcache	Memcache diagnostics
restart	Restart cache service

diagnose wad worker

The `diagnose wad worker` command has some settings that show useful WAD stats for one or all workers. The next options to the command are listed in the table. Some will have their own sub options for refining the output or results.

Option	Description
memory	WAD worker memory diagnostics.
tcp	TCP statistics.
ssl	SSL statistics.
tunnel	Tunnel statistics.
webcache	Webcache diagnostics.
bytecache	Bytecache diagnostics.
memcache	Memcache diagnostics.
restart	Restart workers.

diagnose wacs

Display diagnostic information for the web cache database daemon (wacs).

```
diagnose wacs clear
diagnose wacs recents
diagnose wacs restart
diagnose wacs stats
```

Variable	Description
clear	Remove all entries from the web cache database.
recents	Display recent web cache database activity.
restart	Restart the web cache daemon and reset statistics.
stats	Display web cache statistics.

diagnose wadbd

Display diagnostic information for the WAN optimization database daemon (wadbd).

```
diagnose wadbd {check | clear | recents | restart | stats}
```

Variable	Description
check	Check WAN optimization database integrity.
clear	Remove all entries from the WAN optimization database.
recents	Display recent WAN optimization database activity.
restart	Restart the WAN optimization daemon and reset statistics.
stats	Display WAN optimization statistics.

diagnose debug application {wad | wccpd} [<debug_level>]

View or set the debug level for displaying WAN optimization and web cache-related daemon debug messages. Include a <debug_level> to change the debug level. Leave the <debug_level> out to display the current debug level. Default debug level is 0.

```
diagnose debug application wad [<debug_level>]
diagnose debug application wccpd [<debug_level>]
```

Variable	Description
wad	Set the debug level for the WAN optimization daemon.
wccpd	Set the debug level for the WCCP daemon.

diagnose test application wad 2200

The debug level 2200 switches the debug to explicit proxy mode. You have to enter this debug level first. After that you have to type the command again with a different debug level to check the different explicit proxy statistics. To list what each debug level shows, follow these steps in any FortiGate device:

1. Enable explicit proxy globally and in one interface, to start the wad process. If the wad process is *not* running, you *cannot* list the options.
2. Once the wad process starts, type:

```
diagnose test application wad 2200
diagnose test application wad //// Do not type any debug level value to list all the options.
```

This is the output you will get:

```
# diagnose test application wad 2200
Set diagnosis process: type=wanopt index=0 pid=114
# diagnose test application wad
WAD process 114 test usage:
1: display process status
2: display total memory usage
99: restart all WAD processes
1000: List all WAD processes
1001: dispaly debug level name and values
1002: dispaly status of WANOpt storages
1068: Enable debug for all WAD workers
1069: Disable debug for all WAD workers
2yxx: Set No. xx process of type y as diagnosis process
3: display all fix-sized advanced memory stats
4: display all fix-sized advanced memory stats in details
500000..599999: cmem bucket stats (599999 for usage)
800..899: mem_diag commands (800 for help & usage)
800000..899999: mem_diag commands with 1 arg (800 for help & usage)
80000000..89999999: mem_diag commands with 2 args (800 for help & usage)
60: show debug stats
61: discard all wad debug info that is currently pending
62xxx: set xxxM maximum ouput buffer size for WAD debug (0: set back to default)
68: Enable process debug
69: Disable process debug
98: gracefully stopping WAD process
20: display all listeners
21: display TCP port info
22: display SSL stats
23: flush SSL stats
24: display SSL mem stats
70: display av memory usage
71xxxx: set xxxxMiB maximum AV memory (0: set back to default)
72: toggle av memory protection
73: toggle AV conserve mode (for debug purpose)
90: set to test disk failure
91: unset to test disk failure
92: trigger a disk failure event
100: display explicit proxy settings
101: display firewall policies
102: display security profile mapping for regular firewall policy
```

```
103: display Web proxy forwarding server and group
104: display DNS stats
105: display proxy redirection scan stats
106: list all used fqdns
107: list all firewall address
110: display current web proxy users
111: flush current web proxy users
112: display current web proxy user summary
113: display WAD fss0 state
114: display HTTP digest stats
115: display URL patterns list of cache exemption or forward server
116: toggle dumping URL when daemon crashes
120: display Web Cache stats
121: flush Web Cache stats
122: flush idle Web cache objects
123: display web cache cache sessions
130: display ftpproxy stats
131: clear ftpproxy stats
132: list all current ftpproxy sessions
133: display all catched webfilter profiles
200: display WANOpt profiles
201: display all peers
202: display video cache rules (patterns)
203: display all ssl servers
210: toggle disk-based byte-cache
211: toggle memory-based byte-cache
212: toggle cifs read-ahead
221: display tunnel protocol stats
222: flush tunnel protocol stats
223: display http protocol stats
224: flush http protocol stats
225: display cifs protocol stats
226: flush cifs protocol stats
227: display ftp protocol stats
228: flush ftp protocol stats
229: display mapi protocol stats
230: flush mapi protocol stats
231: display tcp protocol stats
232: flush tcp protocol stats
233: display all protocols stats
234: flush all protocols stats
240: display WAD tunnel stats
241: display tunnel compressor state
242: flush tunnel compressor stats
243: display Byte Cache DB state
244: flush Byte Cache DB stats
245: display Web Cache DB state
246: flush Web Cache DB stats
247: display cache state
248: flush cache stats
249: display memory cache state
250: flush memory cache stats
261yxxxx: set xxxx concurrent Web Cache session for object storage y
262yxxxx: set xxxxK(32K, 64K,...) unconfirmed write/read size per Web Cache object for
          object storage y
263yxxxx: set xxxxK maximum ouput buffer size for object storage y
```

```
264yxx: set lookup lowmark (only if more to define busy status) to be xx for object
         storage y
265yxxx: set xxxK maximum output buffer size for byte storage y
266yxxx: set number of buffered add requests to be xxx for byte storage y
267yxxxx: set number of buffered query requests to be xxxx for byte storage y
268yxxxxx: set number of concurrent query requests to be xxxxx for byte storage y
```



FORTINET®



Copyright© 2018 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.