# TABLE OF CONTENTS

- Change log .................................................................................................................. 7
- Hardware acceleration .................................................................................................. 8
  - What's new in FortiOS 6.0.3 ..................................................................................... 8
  - What's new in FortiOS 6.0.2 ..................................................................................... 8
  - What's new in FortiOS 6.0 ......................................................................................... 8
- Content processors (CP4, CP5, CP6, CP8, and CP9) .................................................. 9
  - CP9 capabilities ...................................................................................................... 9
  - CP8 capabilities ...................................................................................................... 9
  - CP6 capabilities ...................................................................................................... 10
  - CP5 capabilities ...................................................................................................... 10
  - CP4 capabilities ...................................................................................................... 11
  - Determining the content processor in your FortiGate unit ....................................... 11
  - Viewing SSL acceleration status ............................................................................ 11
  - Disabling CP offloading for firewall policies .......................................................... 11
- Security processors (SPs) ............................................................................................ 13
  - SP processing flow .................................................................................................. 13
  - Displaying information about security processing modules .................................... 14
- Network processors (NP6, NP6Lite, and NP4) .......................................................... 16
  - Accelerated sessions on FortiView All Sessions page ............................................. 16
  - NP session offloading in HA active-active configuration .......................................... 17
  - Configuring NP HMAC check offloading ................................................................ 17
  - Software switch interfaces and NP processors ....................................................... 17
  - Disabling NP acceleration for individual IPsec VPN phase 1s ................................. 17
  - Disabling NP offloading for unsupported IPsec encryption or authentication algorithms .................................................................................................................. 18
  - Disabling NP offloading for firewall policies .......................................................... 18
  - Determining the network processors installed in your FortiGate ......................... 18
  - NP hardware acceleration alters packet flow ......................................................... 19
  - NPx traffic logging and monitoring ........................................................................ 20
  - sFlow and NetFlow and hardware acceleration ....................................................... 21
  - Checking that traffic is offloaded by NP processors .............................................. 21
    - Using the packet sniffer ....................................................................................... 21
    - Checking the firewall session offload tag ............................................................ 21
    - Verifying IPsec VPN traffic offloading ................................................................. 22
  - Dedicated management CPU ................................................................................. 23
  - Preventing packet ordering problems ..................................................................... 23
- Strict protocol header checking disables hardware acceleration ............................. 24
- NTurbo and IPSA ......................................................................................................... 25
  - NTurbo offloads flow-based processing ................................................................... 25
  - IPSA offloads flow-based advanced pattern matching ............................................ 26
- NP6 and NP6lite acceleration ....................................................................................... 27
  - NP6 session fast path requirements ........................................................................ 28
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet fast path requirements</td>
<td>29</td>
</tr>
<tr>
<td>Mixing fast path and non-fast path traffic</td>
<td>29</td>
</tr>
<tr>
<td>NP6Lite processors</td>
<td>29</td>
</tr>
<tr>
<td>NP6 and NP6Lite processors and sFlow and NetFlow</td>
<td>29</td>
</tr>
<tr>
<td>NP6 processors and traffic shaping</td>
<td>30</td>
</tr>
<tr>
<td>Viewing your FortiGate NP6 processor configuration</td>
<td>30</td>
</tr>
<tr>
<td>Disabling NP6 and NP6Lite hardware acceleration (fastpath)</td>
<td>31</td>
</tr>
<tr>
<td>Optimizing NP6 performance by distributing traffic to XAUI links</td>
<td>32</td>
</tr>
<tr>
<td>Enabling bandwidth control between the ISF and NP6 XAUI ports</td>
<td>33</td>
</tr>
<tr>
<td>Increasing NP6 offloading capacity using link aggregation groups (LAGs)</td>
<td>34</td>
</tr>
<tr>
<td>NP6 processors and redundant interfaces</td>
<td>34</td>
</tr>
<tr>
<td>Configuring inter-VDOM link acceleration with NP6 processors</td>
<td>35</td>
</tr>
<tr>
<td>Using VLANs to add more accelerated inter-VDOM links</td>
<td>36</td>
</tr>
<tr>
<td>Confirm that the traffic is accelerated</td>
<td>37</td>
</tr>
<tr>
<td>Disabling offloading IPSec Diffie-Hellman key exchange</td>
<td>38</td>
</tr>
<tr>
<td>Access control lists (ACLs)</td>
<td>38</td>
</tr>
<tr>
<td>Configuring individual NP6 processors</td>
<td>39</td>
</tr>
<tr>
<td>Enabling per-session accounting for offloaded NP6 and NP6lite sessions</td>
<td>44</td>
</tr>
<tr>
<td>Configuring NP6 session timeouts</td>
<td>45</td>
</tr>
<tr>
<td>Configure the number of IPsec engines NP6 processors use</td>
<td>46</td>
</tr>
<tr>
<td>Stripping clear text padding and IPSec session ESP padding</td>
<td>47</td>
</tr>
<tr>
<td>Disable NP6 CAPWAP offloading</td>
<td>47</td>
</tr>
<tr>
<td>Optionally disable NP6 offloading of traffic passing between 10Gbps and 1Gbps interfaces</td>
<td>47</td>
</tr>
<tr>
<td>Offloading RDP traffic</td>
<td>48</td>
</tr>
<tr>
<td>NP6 session drift</td>
<td>48</td>
</tr>
<tr>
<td>Optimizing FortiGate-3960E and 3980E IPsec VPN performance</td>
<td>49</td>
</tr>
<tr>
<td>FortiGate-3960E and 3980E support for high throughput traffic streams</td>
<td>50</td>
</tr>
<tr>
<td>Recalculating packet checksums if the iph.reserved bit is set to 0</td>
<td>51</td>
</tr>
<tr>
<td>NP6 IPsec engine status monitoring</td>
<td>51</td>
</tr>
</tbody>
</table>

**FortiGate NP6 architectures**                                                                 | 53   |
<p>| FortiGate-300D fast path architecture                                   | 53   |
| FortiGate-300E and 301E fast path architecture                          | 54   |
| FortiGate-400D fast path architecture                                   | 55   |
| FortiGate-400E and 401E fast path architecture                          | 56   |
| FortiGate-500D fast path architecture                                   | 57   |
| FortiGate-500E and 501E fast path architecture                          | 58   |
| FortiGate-600E and 601E fast path architecture                          | 59   |
| FortiGate-600D fast path architecture                                   | 60   |
| FortiGate-800D fast path architecture                                   | 61   |
| FortiGate-900D fast path architecture                                   | 63   |
| FortiGate-1000D fast path architecture                                  | 64   |
| FortiGate-1200D fast path architecture                                  | 66   |
| FortiGate-1500D fast path architecture                                  | 67   |
| FortiGate-1500DT fast path architecture                                 | 69   |</p>
<table>
<thead>
<tr>
<th>Model</th>
<th>Fast Path Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>FortiGate-2000E</td>
<td>.................................................................................................................. 71</td>
</tr>
<tr>
<td>FortiGate-2500E</td>
<td>.................................................................................................................. 73</td>
</tr>
<tr>
<td>FortiGate-3000D</td>
<td>.................................................................................................................. 75</td>
</tr>
<tr>
<td>FortiGate-3100D</td>
<td>.................................................................................................................. 77</td>
</tr>
<tr>
<td>FortiGate-3200D</td>
<td>.................................................................................................................. 78</td>
</tr>
<tr>
<td>FortiGate-3400E and 3401E</td>
<td>........................................................................................................ 80</td>
</tr>
<tr>
<td>FortiGate-3600E and 3601E</td>
<td>........................................................................................................ 82</td>
</tr>
<tr>
<td>FortiGate-3700D</td>
<td>.................................................................................................................. 83</td>
</tr>
<tr>
<td>FortiGate-3700DX</td>
<td>.................................................................................................................. 87</td>
</tr>
<tr>
<td>FortiGate-3700D low latency</td>
<td>......................................................................................................... 83</td>
</tr>
<tr>
<td>FortiGate-3700D normal latency</td>
<td>....................................................................................................... 85</td>
</tr>
<tr>
<td>FortiGate-3700DX low latency</td>
<td>........................................................................................................ 88</td>
</tr>
<tr>
<td>FortiGate-3700DX normal latency</td>
<td>..................................................................................................... 90</td>
</tr>
<tr>
<td>FortiGate-3800D</td>
<td>.................................................................................................................. 92</td>
</tr>
<tr>
<td>FortiGate-3810D</td>
<td>.................................................................................................................. 94</td>
</tr>
<tr>
<td>FortiGate-3815D</td>
<td>.................................................................................................................. 96</td>
</tr>
<tr>
<td>FortiGate-3960E</td>
<td>.................................................................................................................. 97</td>
</tr>
<tr>
<td>FortiGate-3980E</td>
<td>.................................................................................................................. 99</td>
</tr>
<tr>
<td>FortiGate-5001D</td>
<td>.................................................................................................................. 101</td>
</tr>
<tr>
<td>NP6 default interface mapping</td>
<td>................................................................................................... 102</td>
</tr>
<tr>
<td>NP6 interface mapping with split ports</td>
<td>.................................................................................................. 103</td>
</tr>
<tr>
<td>FortiGate-5001E and 5001E1</td>
<td>...................................................................................................... 103</td>
</tr>
<tr>
<td>NP6 default interface mapping</td>
<td>.................................................................................................. 104</td>
</tr>
<tr>
<td>NP6 interface mapping with split ports</td>
<td>.................................................................................................. 105</td>
</tr>
<tr>
<td>FortiGate-6000 series</td>
<td>.................................................................................................................. 105</td>
</tr>
<tr>
<td>FortiController-5902D</td>
<td>.................................................................................................................. 108</td>
</tr>
<tr>
<td>NP6 content clustering mode interface mapping</td>
<td>.............................................................................. 108</td>
</tr>
<tr>
<td>NP6 default interface mapping</td>
<td>.................................................................................................. 109</td>
</tr>
<tr>
<td>FortiGate-7030E</td>
<td>.................................................................................................................. 110</td>
</tr>
<tr>
<td>FortiGate-7040E</td>
<td>.................................................................................................................. 111</td>
</tr>
<tr>
<td>FortiGate-7060E</td>
<td>.................................................................................................................. 111</td>
</tr>
<tr>
<td>FIM-7901E</td>
<td>.................................................................................................................. 112</td>
</tr>
<tr>
<td>FIM-7904E</td>
<td>.................................................................................................................. 113</td>
</tr>
<tr>
<td>FIM-7910E</td>
<td>.................................................................................................................. 114</td>
</tr>
<tr>
<td>FIM-7920E</td>
<td>.................................................................................................................. 115</td>
</tr>
<tr>
<td>FPM-7620E</td>
<td>.................................................................................................................. 116</td>
</tr>
</tbody>
</table>

**FortiGate NP6lite architectures** ............................................................................ 118

FortiGate-200E and 201E fast path architecture .......................................................... 118

**Hardware acceleration get and diagnose commands** .................................................. 120

- get hardware npu np6 .................................................................................................. 120
- diagnose npu np6 .......................................................................................................... 120
- Using diagnose npu np6 npu-feature to verify enabled NP6 features ......................... 121
- Using diagnose npu np6lite npu-feature to verify enabled NP6Lite features ................ 122
- Using the diagnose sys session/session6 list command ............................................... 123
  - Displaying NP6 offloading information for a session ............................................... 123
Example offloaded IPv4 NP6 session ................................................................. 123
Example IPv4 session that is not offloaded ....................................................... 124
Example IPv4 IPsec NP6 session ...................................................................... 124
Example IPv6 NP6 session .............................................................................. 125
Example NAT46 NP6 session ........................................................................... 125
Example NAT64 NP6 session ........................................................................... 125
diagnose npu np6 session-stats <np6-id> (number of NP6 IPv4 and IPv6 sessions) 126
diagnose npu np6 ipsec-stats (NP6 IPsec statistics) ......................................... 127
diagnose npu np6 sse-stats <np6-id> (number of NP6 sessions and dropped sessions) 128
diagnose npu np6 dce <np6-id> (number of dropped NP6 packets) ................. 128
diagnose hardware deviceinfo nic <interface-name> (number of packets dropped by an interface) ........................................................................................................ 129
diagnose npu np6 synproxy-stats (NP6 SYN-proxyed sessions and unacknowledged SYNs) .......................................................... 129
# Change log

<table>
<thead>
<tr>
<th>Date</th>
<th>Change description</th>
</tr>
</thead>
<tbody>
<tr>
<td>April 29, 2019</td>
<td>FortiGate-6000 information added to FortiGate-6000 series on page 105. FortiGate-7000 information added as well, starting with FortiGate-7030E fast path architecture on page 110.</td>
</tr>
<tr>
<td>February 28, 2019</td>
<td>New model added: FortiGate-400E and 401E fast path architecture on page 56.</td>
</tr>
<tr>
<td>February 6, 2018</td>
<td>New models added: FortiGate-600E and 601E fast path architecture on page 59, and FortiGate-3400E and 3401E fast path architecture on page 80.</td>
</tr>
<tr>
<td>July 26, 2018</td>
<td>FortiOS 6.0.2 document release. See Hardware acceleration on page 8.</td>
</tr>
</tbody>
</table>
Hardware acceleration

Most FortiGate models have specialized acceleration hardware, (called Security Processing Units (SPUs)) that can offload resource intensive processing from main processing (CPU) resources. Most FortiGate units include specialized content processors (CPs) that accelerate a wide range of important security processes such as virus scanning, attack detection, encryption and decryption. (Only selected entry-level FortiGate models do not include a CP processor.) Many FortiGate models also contain security processors (SPs) that accelerate processing for specific security features such as IPS and network processors (NPs) that offload processing of high volume network traffic.

This document describes the Security Processing Unit (SPU) hardware that Fortinet builds into FortiGate devices to accelerate traffic through FortiGate units. Three types of SPUs are described:

- Content processors (CPs) that accelerate a wide range of security functions
- Security processors (SPs) that accelerate specific security functions
- Network processors (NPs and NPLites) that offload network traffic to specialized hardware that is optimized to provide high levels of network throughput.

What’s new in FortiOS 6.0.3

The following list contains new Hardware Acceleration features added in FortiOS 6.0.3. Click on a link to navigate to that section for further information.

- NP6 IPsec engine status monitoring, see NP6 IPsec engine status monitoring on page 51.
- New command added to FortiGates with NP6lite processors: `diagnose npu np6lite npu-feature`, see Using `diagnose npu np6 npu-feature` to verify enabled NP6 features on page 121.

What’s new in FortiOS 6.0.2

The following list contains new Hardware Acceleration features added in FortiOS 6.0.2. Click on a link to navigate to that section for further information.

- Per-session accounting for NP6Lite processors, see Enabling per-session accounting for offloaded NP6 and NP6lite sessions on page 44.

What’s new in FortiOS 6.0

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

- New options for optimizing FortiGate-3960E and 3980E IPsec VPN performance, see Optimizing FortiGate-3960E and 3980E IPsec VPN performance on page 49.
Content processors (CP4, CP5, CP6, CP8, and CP9)

Most FortiGate models contain Security Processing Unit (SPU) Content Processors (CPs) that accelerate many common resource intensive security related processes. CPs work at the system level with tasks being offloaded to them as determined by the main CPU. Capabilities of the CPs vary by model. Newer FortiGate units include CP9 processors. Older CP versions still in use in currently operating FortiGate models include the CP4, CP5, CP6, and CP8.

CP9 capabilities

The CP9 content processor provides the following services:

- Flow-based inspection (IPS, application control etc.) pattern matching acceleration with over 10Gbps throughput
- IPS pre-scan
- IPS signature correlation
- Full match processors
- High performance VPN bulk data engine
- IPSec and SSL/TLS protocol processor
- DES/3DES/AES128/192/256 in accordance with FIPS46-3/FIPS81/FIPS197
- HMAC in accordance with RFC2104/2403/2404 and FIPS198
- ESN mode
- GCM support for NSA "Suite B" (RFC6379/RFC6460) including GCM-128/256; GMAC-128/256
- Key Exchange Processor that supports high performance IKE and RSA computation
- Public key exponentiation engine with hardware CRT support
- Primary checking for RSA key generation
- Handshake accelerator with automatic key material generation
- True Random Number generator
- Elliptic Curve support for NSA "Suite B"
- Sub public key engine (PKCE) to support up to 4096 bit operation directly (4k for DH and 8k for RSA with CRT)
- DLP fingerprint support
- TTTD (Two-Thresholds-Two-Divisors) content chunking
- Two thresholds and two divisors are configurable

CP8 capabilities

The CP8 content processor provides the following services:

- Flow-based inspection (IPS, application control etc.) pattern matching acceleration
- High performance VPN bulk data engine
- IPSec and SSL/TLS protocol processor
- DES/3DES/AES in accordance with FIPS46-3/FIPS81/FIPS197
Content processors (CP4, CP5, CP6, CP8, and CP9)

- ARC4 in compliance with RC4
- MD5/SHA-1/SHA256 with RFC1321 and FIPS180
- HMAC in accordance with RFC2104/2403/2404 and FIPS198
- Key Exchange Processor support high performance IKE and RSA computation
- Public key exponentiation engine with hardware CRT support
- Primarily checking for RSA key generation
- Handshake accelerator with automatic key material generation
- Random Number generator compliance with ANSI X9.31
- Sub public key engine (PKCE) to support up to 4096 bit operation directly
- Message authentication module offers high performance cryptographic engine for calculating SHA256/SHA1/MD5 of data up to 4G bytes (used by many applications)
- PCI express Gen 2 four lanes interface
- Cascade Interface for chip expansion

**CP6 capabilities**

- Dual content processors
- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC with RFC1321 and FIPS180
- HMAC in accordance with RFC2104/2403/2404 and FIPS198
- IPsec protocol processor
- High performance IPsec engine
- Random Number generator compliance with ANSI X9.31
- Key exchange processor for high performance IKE and RSA computation
- Script Processor
- SSL/TLS protocol processor for SSL content scanning and SSL acceleration

**CP5 capabilities**

- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC with RFC1321/2104/2403/2404 and FIPS180/FIPS198
- IPsec protocol processor
- High performance IPsec Engine
- Random Number generator compliant with ANSI X9.31
- Public Key Crypto Engine supports high performance IKE and RSA computation
- Script Processor
Content processors (CP4, CP5, CP6, CP8, and CP9)

CP4 capabilities

- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC
- IPsec protocol processor
- Random Number generator
- Public Key Crypto Engine
- Content processing engine
- ANSI X9.31 and PKCS#1 certificate support

Determining the content processor in your FortiGate unit

Use the `get hardware status` CLI command to determine which content processor your FortiGate unit contains. The output looks like this:

```plaintext
get hardware status
Model name: FortiGate-100D
ASIC version: CP8
ASIC SRAM: 64M
CPU: Intel(R) Atom(TM) CPU D525 @ 1.80GHz
Number of CPUs: 4
RAM: 1977 MB
Compact Flash: 15331 MB /dev/sda
Hard disk: 15272 MB /dev/sda
USB Flash: not available
Network Card chipset: Intel(R) PRO/1000 Network Connection (rev.0000)
Network Card chipset: bcm-sw Ethernet driver 1.0 (rev.)
```

The ASIC version line lists the content processor model number.

Viewing SSL acceleration status

You can view the status of SSL acceleration using the following command:

```plaintext
get vpn status ssl hw-acceleration-status
```

Acceleration hardware detected: kxp=on cipher=on

Where `kxp` means key exchange acceleration.

Disabling CP offloading for firewall policies

If you want to completely disable offloading to CP processors for test purposes or other reasons, you can do so in security policies. Here are some examples:

For IPv4 security policies.
config firewall policy
    edit 1
        set auto-asic-offload disable
    end

For IPv6 security policies.

config firewall policy6
    edit 1
        set auto-asic-offload disable
    end

For multicast security policies.

config firewall multicast-policy
    edit 1
        set auto-asic-offload disable
    end

Disabling auto-asic-offload also disables NP offloading.
Security processors (SPs)

FortiGate Security Processing (SP) modules, such as the SP3 but also including the XLP, XG2, XE2, FE8, and CE4, work at both the interface and system level to increase overall system performance by accelerating specialized security processing. You can configure the SP to favor IPS over firewall processing in hostile high-traffic environments.

SP processors include their own IPS engine which is similar to the FortiOS IPS engine but with the following limitations:

- The SP IPS engine does not support SSL deep inspection. When you have SSL deep inspection enabled for a security policy that includes flow-based inspection or IPS, offloading to the SP is disabled and traffic is processed by the FortiGate CPU and CP processors.
- The SP IPS engine does not support FortiGuard Web Filtering. When you enable flow-based FortiGuard Web Filtering on a FortiGate unit with an SP processor, the SP processor cannot perform FortiGuard lookups and web pages fail to load.

The following security processors are available:

- The SP3 (XLP) is built into the FortiGate-5101B and provides IPS acceleration. No special configuration is required. All IPS processing, including traffic accepted by IPv4 and IPv6 traffic policies and IPv4 and IPv6 DoS policies is accelerated by the built-in SP3 processors.
- The FMC-XG2 is an FMC module with two 10Gb/s SPF+ interfaces that can be used on FortiGate-3950B and FortiGate-3951B units.
- The FortiGate-3140B also contains a built-in XG2 using ports 19 and 20.
- The ADM-XE2 is a dual-width AMC module with two 10Gb/s interfaces that can be used on FortiGate-3810A and FortiGate-5001A-DW systems.
- The ADM-FE2 is a dual-width AMC module with eight 1Gb/s interfaces that can be used with the FortiGate-3810A.
- The ASM-CE4 is a single-width AMC module with four 10/100/1000 Mb/s interfaces that can be used on FortiGate-3016B and FortiGate-3810A units.

Traffic is blocked if you enable IPS for traffic passing over inter-VDOM links if that traffic is being offloaded by an SP processor. If you disable SP offloading, traffic will be allowed to flow. You can disable offloading in individual firewall policies by disabling auto-asic-offload for those policies. You can also use the following command to disable all IPS offloading:

```
config ips global
    set np-accel-mode none
    set cp-accel-mode none
end
```

SP processing flow

SP processors provide an integrated high performance fast path multilayer solution for both intrusion protection and firewall functions. The multilayered protection starts from anomaly checking at packet level to ensure each packet is sound and reasonable. Immediately after that, a sophisticated set of interface based packet anomaly protection, DDoS protection, policy based intrusion protection, firewall fast path, and behavior based methods are employed to prevent DDoS attacks from the rest of system.
Then the packets enter an interface/policy based intrusion protection system, where each packet is evaluated against a set of signatures. The end result is streams of user packets that are free of anomaly and attacks, entering the fast path system for unicast or multicast fast path forwarding.

**SP processing flow**

![SP processing flow diagram]

**Displaying information about security processing modules**

You can display information about installed SP modules using the CLI command

```plaintext
diagnose npu spm
```

For example, for the FortiGate-5101C:

```plaintext
FG-5101C # diagnose npu spm list
Available SP Modules:

<table>
<thead>
<tr>
<th>ID</th>
<th>Model</th>
<th>Slot</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>xh0</td>
<td>built-in</td>
<td>port1, port2, port3, port4, base1, base2, fabric1, fabric2, eth10, eth11, eth12, eth13, eth14, eth15, eth16, eth17, eth18, eth19</td>
</tr>
</tbody>
</table>
```

You can also use this command to get more info about SP processing. This example shows how to display details about how the module is processing sessions using the syn proxy.

```plaintext
diagnose npu spm dos synproxy <sp_id>
```
This is a partial output of the command:

Number of proxied TCP connections : 0
Number of working proxied TCP connections : 0
Number of retired TCP connections : 0
Number of valid TCP connections : 0
Number of attacks, no ACK from client : 0
Number of no SYN-ACK from server : 0
Number of reset by server (service not supported): 0
Number of established session timeout : 0
Client timeout setting : 3 Seconds
Server timeout setting : 3 Seconds
Network processors (NP6, NP6Lite, and NP4)

FortiASIC network processors work at the interface level to accelerate traffic by offloading traffic from the main CPU. Current models contain NP6, and NP6lite network processors. Older FortiGate models include NP1 network processors (also known as FortiAccel, or FA2), NP2, NP4, and NP4Lite network processors.

The traffic that can be offloaded, maximum throughput, and number of network interfaces supported by each varies by processor model:

- NP6 supports offloading of most IPv4 and IPv6 traffic, IPsec VPN encryption, CAPWAP traffic, and multicast traffic. The NP6 has a maximum throughput of 40 Gbps using 4 x 10 Gbps XAUI or Quad Serial Gigabit Media Independent Interface (QSGMII) interfaces or 3 x 10 Gbps and 16 x 1 Gbps XAUI or QSGMII interfaces. For details about the NP6 processor, see NP6 and NP6lite acceleration on page 27 and for information about FortiGate models with NP6 processors, see FortiGate NP6 architectures on page 53.

- NP6lite is similar to the NP6 but with a lower throughput and some functional limitations (for example, the NP6lite does not offload CAPWAP traffic). The NP6lite has a maximum throughput of 10 Gbps using 2x QSGMII and 2x Reduced gigabit media-independent interface (RGMII) interfaces. For details about the NP6 processor, see NP6 and NP6lite acceleration on page 27 and for information about FortiGate models with NP6 processors, see FortiGate NP6lite architectures on page 118.

- NP4 supports offloading of most IPv4 firewall traffic and IPsec VPN encryption. The NP4 has a capacity of 20 Gbps through 2 x 10 Gbps interfaces. For details about NP4 processors, see and for information about FortiGate models with NP4 processors, see .

- NP4lite is similar to the NP4 but with a lower throughput (but with about half the performance ) and some functional limitations.

- NP2 supports IPv4 firewall and IPsec VPN acceleration. The NP2 has a capacity of 2 Gbps through 2 x 10 Gbps interfaces or 4 x 1 Gbps interfaces.

- NP1 supports IPv4 firewall and IPsec VPN acceleration with 2 Gbps capacity. The NP1 has a capacity of 2 Gbps through 2 x 1 Gbps interfaces.

- The NP1 does not support frames greater than 1500 bytes. If your network uses jumbo frames, you may need to adjust the MTU (Maximum Transmission Unit) of devices connected to NP1 ports. Maximum frame size for NP2, NP4, and NP6 processors is 9216 bytes.

- For both NP1 and NP2 network processors, ports attached to a network processor cannot be used for firmware installation by TFTP.

Sessions that require proxy-based security features (for example, virus scanning, IPS, application control and so on) are not fast pathed and must be processed by the CPU. Sessions that require flow-based security features can be offloaded to NP4 or NP6 network processors if the FortiGate supports NTurbo.

Accelerated sessions on FortiView All Sessions page

When viewing sessions in the FortiView All Sessions console, NP4/ NP6 accelerated sessions are highlighted with an NP4 or NP6 icon. The tooltip for the icon includes the NP processor type and the total number of accelerated sessions.

You can also configure filtering to display FortiASIC sessions.
NP session offloading in HA active-active configuration

Network processors can improve network performance in active-active (load balancing) high availability (HA) configurations, even though traffic deviates from general offloading patterns, involving more than one network processor, each in a separate FortiGate unit. No additional offloading requirements apply.

Once the primary FortiGate unit’s main processing resources send a session key to its network processor(s), network processor(s) on the primary unit can redirect any subsequent session traffic to other cluster members, reducing traffic redirection load on the primary unit’s main processing resources.

As subordinate units receive redirected traffic, each network processor in the cluster assesses and processes session offloading independently from the primary unit. Session key states of each network processor are not part of synchronization traffic between HA members.

Configuring NP HMAC check offloading

Hash-based Message Authentication Code (HMAC) checks offloaded to network processors by default. You can enter the following command to disable this feature:

```
configure system global
  set ipsec-hmac-offload disable
end
```

Software switch interfaces and NP processors

FortiOS supports creating a software switch by grouping two or more FortiGate physical interfaces into a single virtual or software switch interface. All of the interfaces in this virtual switch act like interfaces in a hardware switch in that they all have the same IP address and can be connected to the same network. You create a software switch interface from the CLI using the command `config system switch-interface`.

The software switch is a bridge group of several interfaces, and the FortiGate CPU maintains the mac-port table for this bridge. As a result of this CPU involvement, traffic processed by a software switch interface is not offloaded to network processors.

Disabling NP acceleration for individual IPsec VPN phase 1s

Use the following command to disable NP offloading for an interface-based IPsec VPN phase 1:

```
config vpn ipsec phase1-interface
  edit phase1-name
    set npu-offload disable
end
```

Use the following command to disable NP offloading for a policy-based IPsec VPN phase 1:

```
config vpn ipsec phase1
  edit phase1-name
    set npu-offload disable
end
```

The `npu-offload` option is enabled by default.
Disabling NP offloading for unsupported IPsec encryption or authentication algorithms

In general, more recent IPsec VPN encryption and authentication algorithms may not be supported by older NP processors. For example, NP4 network processors do not support SHA-256, SHA-384, and SHA-512. IPsec traffic with unsupported algorithms is not offloaded and instead is processed by the FortiGate CPU. In addition, this configuration may cause packet loss and other performance issues. If you experience packet loss or performance problems you should set the `npu-offload` option to `disable`. Future FortiOS versions should prevent selecting algorithms not supported by the hardware.

Disabling NP offloading for firewall policies

Use the following options to disable NP offloading for specific security policies:

For IPv4 security policies.

```bash
config firewall policy
edit 1
set auto-asic-offload disable
end
```

For IPv6 security policies.

```bash
config firewall policy6
edit 1
set auto-asic-offload disable
end
```

For multicast security policies.

```bash
config firewall multicast-policy
edit 1
set auto-asic-offload disable
end
```

Determining the network processors installed in your FortiGate

Use either of the following command to list the NP6 processors in your FortiGate unit:

```
get hardware npu np6 port-list
diagnose npu np6 port-list
```

Use either of the following command to list the NP6Lite processors in your FortiGate unit:

```
get hardware npu np6lite port-list
diagnose npu np6lite port-list
```

To list other network processors on your FortiGate unit, use the following CLI command.

```
get hardware npu <model> list
```

`<model>` can be `legacy, np1, np2` or `np4`.

The output lists the interfaces that have the specified processor. For example, for a FortiGate-5001B:

```
get hardware npu np4 list
```
Network processors (NP6, NP6Lite, and NP4)

<table>
<thead>
<tr>
<th>ID</th>
<th>Model</th>
<th>Slot</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>On-board</td>
<td></td>
<td>port1 port2 port3 port4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fabric1 base1 npu0-vlink0 npu0-vlink1</td>
</tr>
<tr>
<td>1</td>
<td>On-board</td>
<td></td>
<td>port5 port6 port7 port8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fabric2 base2 npu1-vlink0 npu1-vlink1</td>
</tr>
</tbody>
</table>

The npu0-vlink0, npu1-vlink1 etc interfaces are used for accelerating inter-VDOM links.

NP hardware acceleration alters packet flow

NP hardware acceleration generally alters packet flow as follows:

1. Packets initiating a session pass to the FortiGate unit’s main processing resources (CPU).
2. The FortiGate unit assesses whether the session matches fast path (offload) requirements. To be suitable for offloading, traffic must possess only characteristics that can be processed by the fast path. The list of requirements depends on the processor, see NP6 and NP6lite acceleration on page 27 or . If the session can be fast pathed, the FortiGate unit sends the session key or IPsec security association (SA) and configured firewall processing action to the appropriate network processor.
3. Network processors continuously match packets arriving on their attached ports against the session keys and SAs they have received.
   - If a network processor’s network interface is configured to perform hardware accelerated anomaly checks, the network processor drops or accepts packets that match the configured anomaly patterns. These checks are separate from and in advance of anomaly checks performed by IPS, which is not compatible with network processor offloading. See .
   - The network processor next checks for a matching session key or SA. If a matching session key or SA is found, and if the packet meets packet requirements, the network processor processes the packet according to the configured action and then sends the resulting packet. This is the actual offloading step. Performing this processing on the NP processor improves overall performance because the NP processor is optimized for this task. As well, overall FortiGate performance is improved because the CPU has fewer sessions to process.
**NP network processor packet flow**

- If a matching session key or SA is not found, or if the packet does not meet packet requirements, the packet cannot be offloaded. The network processor sends the data to the FortiGate unit's CPU, which processes the packet.

Encryption and decryption of IPsec traffic originating from the FortiGate can utilize network processor encryption capabilities.

Packet forwarding rates vary by the percentage of offloadable processing and the type of network processing required by your configuration, but are independent of frame size. For optimal traffic types, network throughput can equal wire speed.

**NPx traffic logging and monitoring**

Except for the NP6, network processors do not count offloaded packets, and offloaded packets are not logged by traffic logging and are not included in traffic statistics and traffic log reports.

NP6 processors support per-session traffic and byte counters, Ethernet MIB matching, and reporting through messages resulting in traffic statistics and traffic log reporting.
sFlow and NetFlow and hardware acceleration

NP6 offloading is supported when you configure NetFlow for interfaces connected to NP6 processors.

Configuring sFlow on any interface disables all NP4 and NP6 offloading for all traffic on that interface. As well, configuring NetFlow on any interface disables NP4 offloading for all traffic on that interface.

Checking that traffic is offloaded by NP processors

A number of diagnose commands can be used to verify that traffic is being offloaded.

Using the packet sniffer

Use the packet sniffer to verify that traffic is offloaded. Offloaded traffic is not picked up by the packet sniffer so if you are sending traffic through the FortiGate unit and it is not showing up on the packet sniffer you can conclude that it is offloaded.

diag sniffer packet port1 <option>

If you want the packet sniffer to be able to see offloaded traffic you can temporarily disable offloading the traffic, run the packet sniffer to view it and then re-enable offloading. As an example, you may want to sniff the traffic that is accepted by a specific firewall policy. You can edit the policy and set the autoasicoffload option to disable to disable offloading this traffic. You can also disable offloading for IPsec VPN traffic, see Checking that traffic is offloaded by NP processors on page 21.

Checking the firewall session offload tag

Use the diagnose sys session list command to display sessions. If the output for a session includes the npu info field you should see information about session being offloaded. If the output doesn’t contain an npu info field then the session has not been offloaded.

diagnose sys session list
session info: proto=6 proto_state=01 duration=34 expire=3565 timeout=3600 flags=00000000 sockflag=000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=295/3/1 reply=60/1/1 tuples=2
origin->sink: org pre->post, reply pre->post dev=48->6/6->48 gwy=10.1.100.11/11.11.11.1
hook=pre dir=org act=noop 172.16.200.55:56453->10.1.100.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.11:80->172.16.200.55:56453(0.0.0.0:0)
post/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 id_policy_id=0 auth_info=0 chk_client_info=0 vd=4
Verifying IPsec VPN traffic offloading

The following commands can be used to verify IPsec VPN traffic offloading to NP processors.

diagnose vpn ipsec status

**NP1/NP2/NP4_0/sp_0_0:**

- null: 0 0
- des: 0 0
  - 3des: 4075 4074
- aea: 0 0
- seed: 0 0
- null: 0 0
- md5: 4075 4074
- sha1: 0 0
- sha256: 0 0
- sha384: 0 0
- sha512: 0 0

**npu info:**

- flag: 0x81/0x81
- offload: 4/4
- ips_offload: 0/0
- epid: 1/23
- ipid: 23/1
- vlan: 32779/0

**diagnose vpn tunnel list**

- list all ipsec tunnel in vd 3

**name=p1-vdom1 ver=1 serial=5 11.11.11.1:0->11.11.11.2:0 lgwy=static tun=tunnel mode=auto**

- bound_if=47
- proxyid_num=1 child_num=0 refcnt=8 ilast=2 olast=2
- stat: rxp=3076 txp=1667 rxb=4299623276 txb=66323
- dpd: mode=active on=1 idle=5000ms retry=3 count=0 seqno=20
- natt: mode=none draft=0 interval=0 remote_port=0
- proxyid=p2-vdom1 proto=0 sa=1 ref=2 auto_negotiate=0 serial=1
- src: 0:0.0.0.0/0.0.0.0:0
- dst: 0:0.0.0.0/0.0.0.0:0
- SA: ref=6 options=0000000e type=0 soft=0 mtu=1436 expire=1736 replaywin=2048 seqno=680
- life: type=0 bytes=0/0 timeout=1748/1800
- dec: spi=ae01010c esp=3des key=24 18e021bcace225347459189f292fbc2e4677563b07498a07
- ah=md5 key=16 b4f44368741632b4e33e5f5b794253d3
- enc: spi=ae01010d esp=3des key=24 42c94a8a2f72a44f9a37777f8e6aa3b24160b8af15f4a573
- ah=md5 key=16 6214155f76b3a93345dcc9ec02d6415
- dec: pkts/bytes=3073/4299621477, enc: pkts/bytes=1667/66375

**npu_flag=03 npu_rgwy=11.11.12 npu_lgwy=11.11.1 npu_delid=4**

**diagnose sys session list**

- session info: proto=6 proto_state=01 duration=34 expire=3565 timeout=3600 flags=00000000
  - sockflag=00000000 sockport=0 av_idx=0 use=3
- origin-shaper=
- reply-shaper=
- per_ip_shaper=
- ha_id=0 policy_dir=0 tunnel=/p1-vdom2
- state=re may_dirty npu
Network processors (NP6, NP6Lite, and NP4)

statistic(bytes/packets/allow_err): org=112/2/1 reply=112/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=57->7/7->57 gwy=10.1.100.11/11.11.11.1
hook=pre dir=org act=noop 172.16.200.55:35254->10.1.100.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.11:80->172.16.200.55:35254(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 id_policy_id=0 auth_info=0 chk_client_info=0 vd=4
serial=00002d29 tos/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=172.16.200.55, bps=260
npu_state=00000000
npu info: flag=0x81/0x82, offload=7/7, ips_offload=0/0, epid=1/3, ipid=3/1, vlan=32779/0

Dedicated management CPU

The GUI and CLI of FortiGate units with NP6 and NP4 processors may become unresponsive when the system is under heavy processing load because NP6 or NP4 interrupts overload the CPUs preventing CPU cycles from being used for management tasks. You can resolve this issue by using the following command to dedicate CPU core 0 to management tasks.

```
cfg set system npu
  set dedicated-management-cpu [enable | disable]
end
```

All management tasks are then processed by CPU 0 and NP6 or NP4 interrupts are handled by the remaining CPU cores.

Preventing packet ordering problems

In some cases when FortiGate units with NP4, NP6, or NP6lite processors are under heavy load the packets used in the TCP 3-way handshake of some sessions may be transmitted by the FortiGate in the wrong order resulting in the TCP sessions failing.

If you notice TCP sessions failing when a FortiGate with NP4, NP6, or NP6lite processors is very busy you can enable delay-tcp-npu-session in the firewall policy receiving the traffic. This option resolves the problem by delaying the session to make sure that there is time for all of the handshake packets to reach the destination before the session begins transmitting data.

```
cfg set firewall policy
  set delay-tcp-npu-session enable
end
```
Strict protocol header checking disables hardware acceleration

You can use the following command to cause the FortiGate to apply strict header checking to verify that a packet is part of a session that should be processed. Strict header checking includes verifying the layer-4 protocol header length, the IP header length, the IP version, the IP checksum, IP options, and verifying 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.

```
config system global
    check-protocol-header strict
end
```

Enabling strict header checking disables all hardware acceleration. This includes NP, SP, and CP processing.
You can use the following command to configure NTurbo and IPA Advanced (IPSA) offloading and acceleration of firewall sessions that have flow-based security profiles. This includes firewall sessions with IPS, application control, CASI, flow-based antivirus and flow-based web filtering.

```
config ips global
  set np-accel-mode {none | basic}
  set cp-accel-mode {none | basic | advanced}
end
```

**NTurbo offloads flow-based processing**

NTurbo offloads firewall sessions that include flow-based security profiles to NP4 or NP6 network processors. Without NTurbo, or with NTurbo disabled, all firewall sessions that include flow-based security profiles are processed by the FortiGate CPU.

---

**NTurbo can only offload firewall sessions containing flow-based security profiles if the session could otherwise have been offloaded except for the presence of the flow-based security profiles. If something else prevents the session from being offloaded, NTurbo will not offload that session.**

---

**Firewall sessions that include proxy-based security profiles are never offloaded to network processors and are always processed by the FortiGate CPU.**

---

NTurbo creates a special data path to redirect traffic from the ingress interface to IPS, and from IPS to the egress interface. NTurbo allows firewall operations to be offloaded along this path, and still allows IPS to behave as a stage in the processing pipeline, reducing the workload on the FortiGate CPU and improving overall throughput.

---

**NTurbo sessions still offload pattern matching and other processes to CP processors, just like normal flow-based sessions.**

---

If NTurbo is supported by your FortiGate unit, you can use the following command to configure it:

```
config ips global
  set np-accel-mode {basic | none}
end
```

`basic` enables NTurbo and is the default setting for FortiGate models that support NTurbo. `none` disables NTurbo. If the `np-accel-mode` option is not available, then your FortiGate does not support NTurbo.

There are some special cases where sessions may not be offloaded by NTurbo, even when NTurbo is explicitly enabled. In these cases the sessions are handled by the FortiGate CPU.
- NP acceleration is disabled. For example, `auto-asic-offload` is disabled in the firewall policy configuration.
- The firewall policy includes proxy-based security profiles.
- The sessions require FortiOS session-helpers. For example, FTP sessions can not be offloaded to NP processors because FTP sessions use the FTP session helper.
- Interface policies or DoS policies have been added to the ingress or egress interface.
- Tunneling is enabled. Any traffic to or from a tunneled interface (IPSec, IPinIP, SSL VPN, GRE, CAPWAP, etc.) cannot be offloaded by NTurbo.

**IPSA offloads flow-based advanced pattern matching**

IPSA offloads advanced or enhanced pattern matching operations required for flow-based content processing to CP8 and CP9 Content Processors. IPSA offloads enhanced pattern matching for NTurbo firewall sessions and firewall sessions that are not offloaded to NP processors. When IPSA is turned on, flow-based pattern databases are compiled and downloaded to the content processors from the IPS engine and IPS database. Flow-based pattern matching requests are redirected to the CP hardware reducing the load on the FortiGate CPU and accelerating pattern matching.

**IF IPSA is supported on your FortiGate unit, you can use the following command to configure it:**

```plaintext
config ips global
  set cp-accel-mode {advanced | basic | none}
end
```

basic offloads basic pattern matching. advanced offloads more types of pattern matching resulting in higher throughput than basic mode. advanced is only available on FortiGate models with two or more CP8s or one or more CP9s. If the `cp-accel-mode` option is not available, then your FortiGate does not support IPSA.

On FortiGates with one CP8, the default `cp-accel-mode` is basic. Setting the mode to advanced does not change the types of pattern matching that are offloaded.

On FortiGates with two or more CP8s or one or more CP9s the default `cp-accel-mode` is advanced. You can set the mode to basic to offload fewer types of pattern matching.
NP6 and NP6lite acceleration

NP6 and NP6lite network processors provide fastpath acceleration by offloading communication sessions from the FortiGate CPU. When the first packet of a new session is received by an interface connected to an NP6 processor, just like any session connecting with any FortiGate interface, the session is forwarded to the FortiGate CPU where it is matched with a security policy. If the session is accepted by a security policy and if the session can be offloaded its session key is copied to the NP6 processor that received the packet. All of the rest of the packets in the session are intercepted by the NP6 processor and fast-pathed out of the FortiGate unit to their destination without ever passing through the FortiGate CPU. The result is enhanced network performance provided by the NP6 processor plus the network processing load is removed from the CPU. In addition the NP6 processor can handle some CPU intensive tasks, like IPsec VPN encryption/decryption.

NP6lite processors have the same architecture and function in the same way as NP6 processors. All of the descriptions of NP6 processors in this document can be applied to NP6lite possessors except where noted.

Session keys (and IPsec SA keys) are stored in the memory of the NP6 processor that is connected to the interface that received the packet that started the session. All sessions are fast-pathed and accelerated, even if they exit the FortiGate unit through an interface connected to another NP6. There is no dependence on getting the right pair of interfaces since the offloading is done by the receiving NP6.

The key to making this possible is an Integrated Switch Fabric (ISF) that connects the NP6s and the FortiGate unit interfaces together. Many FortiGate units with NP6 processors also have an ISF. The ISF allows any port connectivity. All ports and NP6s can communicate with each other over the ISF. There are no special ingress and egress fast path requirements as long as traffic enters and exits on interfaces connected to the same ISF.

Some FortiGate units, such as the FortiGate-1000D include multiple NP6 processors that are not connected by an ISF. Because the ISF is not present fast path acceleration is supported only between interfaces connected to the same NP6 processor. Since the ISF introduces some latency, models with no ISF provide low-latency network acceleration between network interfaces connected to the same NP6 processor.

Each NP6 has a maximum throughput of 40 Gbps using 4 x 10 Gbps XAUI or Quad Serial Gigabit Media Independent Interface (QSGMII) interfaces or 3 x 10 Gbps and 16 x 1 Gbps XAUI or QSGMII interfaces.

There are at least two limitations to keep in mind:

- The capacity of each NP6 processor. An individual NP6 processor can support between 10 and 16 million sessions. This number is limited by the amount of memory the processor has. Once an NP6 processor hits its session limit, sessions that are over the limit are sent to the CPU. You can avoid this problem by as much as possible distributing incoming sessions evenly among the NP6 processors. To be able to do this you need to be aware of which interfaces connect to which NP6 processors and distribute incoming traffic accordingly.

- The NP6 processors in some FortiGate units employ NP direct technology that removes the ISF. The result is very low latency but no inter-processor connectivity requiring you to make sure that traffic to be offloaded enters and exits the FortiGate through interfaces connected to the same NP processor.
NP6 session fast path requirements

NP6 processors can offload the following traffic and services:

- IPv4 and IPv6 traffic and NAT64 and NAT46 traffic (as well as IPv4 and IPv6 versions of the following traffic types where appropriate).
- Link aggregation (LAG) (IEEE 802.3ad) traffic and traffic from static redundant interfaces (see NP6 session fast path requirements on page 28).
- TCP, UDP, ICMP, SCTP, and RDP traffic.
- IPsec VPN traffic, and offloading of IPsec encryption/decryption (including SHA2-256 and SHA2-512)
- NP6 processor IPsec engines support null, DES, 3DES, AES128, AES192, and AES256 encryption algorithms
- NP6 processor IPsec engines support null, MD5, SHA1, SHA256, SHA 384, and SHA512 authentication algorithms
- IPsec traffic that passes through a FortiGate without being unencrypted.
- Anomaly-based intrusion prevention, checksum offload and packet defragmentation.
- IPIP tunneling (also called IP in IP tunneling), SIT tunneling, and IPv6 tunneling sessions.
- Multicast traffic (including Multicast over IPsec).
- CAPWAP and wireless bridge traffic tunnel encapsulation to enable line rate wireless forwarding from FortiAP devices (not supported by the NP6lite).
- Traffic shaping and priority queuing for both shared and per IP traffic shaping.
- Syn proxying (not supported by the NP6lite).
- DNS session helper (not supported by the NP6lite)/
- Inter-VDOM link traffic.

Sessions that are offloaded must be fast path ready. For a session to be fast path ready it must meet the following criteria:

- Layer 2 type/length must be 0x0800 for IPv4 or 0x86dd for IPv6 (IEEE 802.1q VLAN specification is supported).
- Layer 3 protocol can be IPv4 or IPv6.
- Layer 4 protocol can be UDP, TCP, ICMP, or SCTP.
- In most cases, Layer 3 / Layer 4 header or content modification sessions that require a session helper can be offloaded.
- Local host traffic (originated by the FortiGate unit) can be offloaded.
- If the FortiGate supports, NTurbo sessions can be offloaded if they are accepted by firewall policies that include IPS, Application Control, CASI, flow-based antivirus, or flow-based web filtering.

Offloading Application layer content modification is not supported. This means that sessions are not offloaded if they are accepted by firewall policies that include proxy-based virus scanning, proxy-based web filtering, DNS filtering, DLP, Anti-Spam, VoIP, ICAP, Web Application Firewall, or Proxy options.

If you disable anomaly checks by Intrusion Prevention (IPS), you can still enable hardware accelerated anomaly checks using the `fp-anomaly` field of the `config system interface` CLI command. See NP6 session fast path requirements on page 28.

If a session is not fast path ready, the FortiGate unit will not send the session key or IPsec SA key to the NP6 processor. Without the session key, all session key lookup by a network processor for incoming packets of that session fails,
causing all session packets to be sent to the FortiGate unit’s main processing resources, and processed at normal speeds.

If a session is fast path ready, the FortiGate unit will send the session key or IPsec SA key to the network processor. Session key or IPsec SA key lookups then succeed for subsequent packets from the known session or IPsec SA.

**Packet fast path requirements**

Packets within the session must then also meet packet requirements.

- Incoming packets must not be fragmented.
- Outgoing packets must not require fragmentation to a size less than 385 bytes. Because of this requirement, the configured MTU (Maximum Transmission Unit) for a network processor’s network interfaces must also meet or exceed the network processors’ supported minimum MTU of 385 bytes.

**Mixing fast path and non-fast path traffic**

If packet requirements are not met, an individual packet will be processed by the FortiGate CPU regardless of whether other packets in the session are offloaded to the NP6.

Also, in some cases, a protocol’s session(s) may receive a mixture of offloaded and non-offloaded processing. For example, VoIP control packets may not be offloaded but VoIP data packets (voice packets) may be offloaded.

**NP6Lite processors**

The NP6Lite works the same way as the NP6. Being a lighter version, the NP6Lite has a lower capacity than the NP6. The NP6Lite max throughput is 10 Gbps using 2x QSGMII and 2x Reduced gigabit media-independent interface (RGMII) interfaces.

Also, the NP6Lite does not offload the following types of sessions:

- CAPWAP
- Syn proxy
- DNS session helper

**NP6 and NP6Lite processors and sFlow and NetFlow**

NP6 and NP6Lite offloading is supported when you configure NetFlow for interfaces connected to NP6 or NP6Lite processors. Offloading of other sessions is not affected by configuring NetFlow.

Configuring sFlow on any interface disables all NP6 and NP6Lite offloading for all traffic on that interface.
NP6 processors and traffic shaping

NP6-offloaded sessions support most types of traffic shaping. However, in bandwidth and out bandwidth traffic shaping, set using the following command, is not supported:

```
config system interface
  edit port1
    set outbandwidth <value>
    set inbandwidth <value>
end
```

Configuring in bandwidth traffic shaping has no effect. Configuring out bandwidth traffic shaping imposes more limiting than configured, potentially reducing throughput more than expected.

Viewing your FortiGate NP6 processor configuration

Use either of the following commands to view the NP6 processor hardware configuration of your FortiGate unit:

```
get hardware npu np6 port-list
diagnose npu np6 port-list
```

If your FortiGate has NP6lite processors, you can use either of the following commands:

```
get hardware npu np6lite port-list
diagnose npu np6lite port-list
```

For example, for the FortiGate-5001D the output would be:

```
Chip  XAUI Ports  Max Speed  Cross-chip Speed  offloading
------ ---- ------- ----- --------------
np6_0 0 port3  10G   Yes
1  
  2 base1  1G   Yes
  3
  0-3 port1  40G  Yes
  0-3 fabric1  40G  Yes
  0-3 fabric3  40G  Yes
  0-3 fabric5  40G  Yes
np6_1 0  
  1 port4  10G  Yes
  2
  3 base2  1G  Yes
  0-3 port2  40G  Yes
  0-3 fabric2  40G  Yes
  0-3 fabric4  40G  Yes
------ ---- ------- ----- --------------
```

For more example output for different FortiGate models, see FortiGate NP6 architectures on page 53 and FortiGate NP6lite architectures on page 118.
You can also use the following command to view the features enabled or disabled on the NP6 processors in your FortiGate unit:

```
diagnose npu np6 npu-feature
```

<table>
<thead>
<tr>
<th>Feature</th>
<th>NP6</th>
<th>NP6lite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastpath</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>HPE-type-shaping</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>Standalone</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>IPv4 firewall</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 firewall</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv4 IPSec</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 IPSec</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv4 tunnel</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 tunnel</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>GRE tunnel</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>GRE passthrough</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv4 Multicast</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 Multicast</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CAPWAP</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>RDP Offload</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The following command is available to view the features enabled or disabled on the NP6Lite processors in your FortiGate unit:

```
diagnose npu np6lite npu-feature
```

<table>
<thead>
<tr>
<th>Feature</th>
<th>NP6</th>
<th>NP6lite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fastpath</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>IPv4 firewall</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 firewall</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv4 IPSec</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 IPSec</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv4 tunnel</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 tunnel</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>GRE tunnel</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>IPv4 Multicast</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6 Multicast</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Disabling NP6 and NP6lite hardware acceleration (fastpath)

You can use the following command to disable NP6 offloading for all traffic. This option disables NP6 offloading for all traffic for all NP6 and NP6lite processors.

```
config system npu
  set fastpath disable
end
```
Optimizing NP6 performance by distributing traffic to XAUI links

On most FortiGate units with NP6 processors, the FortiGate interfaces are switch ports that connect to the NP6 processors with XAUI links. Packets pass from the interfaces to the NP6 processor over the XAUI links. Each NP6 processor has a 40 Gigabit bandwidth capacity. The four XAUI links each have a 10 Gigabit capacity for a total of 40 Gigabits.

On many FortiGate units with NP6 processors, the NP6 processors and the XAUI links are over-subscribed. Since the NP6 processors are connected by an Integrated Switch Fabric, you do not have control over how traffic is distributed to them. In fact traffic is distributed evenly by the ISF.

However, you can control how traffic is distributed to the XAUI links and you can optimize performance by distributing traffic evenly among the XAUI links. For example, if you have a very high amount of traffic passing between two networks, you can connect each network to interfaces connected to different XAUI links to distribute the traffic for each network to a different XAUI link.

For example, on a FortiGate-3200D (See FortiGate-3200D fast path architecture on page 78), there are 48 10-Gigabit interfaces that send and receive traffic for two NP6 processors over a total of eight 10-Gigabit XAUI links. Each XAUI link gets traffic from six 10-Gigabit FortiGate interfaces. The amount of traffic that the FortiGate-3200D can offload is limited by the number of NP6 processors and the number of XAUI links. You can optimize the amount of traffic that the FortiGate-3200D can process by distributing it evenly among the XAUI links and the NP6 processors.

You can see the Ethernet interface, XAUI link, and NP6 configuration by entering the `get hardware npu np6 port-list` command. For the FortiGate-3200D the output is:

```
get hardware npu np6 port-list
Chip XAUI Ports Max Cross-chip Speed offloading
        ------ ---- -------- ------- -------
np6_0   0 port1  10G  Yes
        0 port5  10G  Yes
        0 port10 10G Yes
        0 port13 10G Yes
        0 port17 10G Yes
        0 port22 10G Yes
        1 port2  10G Yes
        1 port6  10G Yes
        1 port9  10G Yes
        1 port14 10G Yes
        1 port18 10G Yes
        1 port21 10G Yes
        2 port3  10G Yes
        2 port7  10G Yes
        2 port12 10G Yes
        2 port15 10G Yes
        2 port19 10G Yes
        2 port24 10G Yes
        3 port4  10G Yes
        3 port8  10G Yes
        3 port11 10G Yes
        3 port16 10G Yes
        3 port20 10G Yes
        3 port23 10G Yes

np6_1   0 port26 10G Yes
```
In this command output you can see that each NP6 has for four XAUI links (0 to 3) and that each XAUI link is connected to six 10-gigabit Ethernet interfaces. To optimize throughput you should keep the amount of traffic being processed by each XAUI port to under 10 Gbps. So for example, if you want to offload traffic from four 10-gigabit networks you can connect these networks to Ethernet interfaces 1, 2, 3 and 4. This distributes the traffic from each 10-Gigabit network to a different XAUI link. Also, if you wanted to offload traffic from four more 10-Gigabit networks you could connect them to Ethernet ports 26, 25, 28, and 27. As a result each 10-Gigabit network would be connected to a different XAUI link.

Enabling bandwidth control between the ISF and NP6 XAUI ports

In some cases, the Internal Switch Fabric (ISF) buffer size may be larger than the buffer size of an NP6 XAUI port that receives traffic from the ISF. If this happens, burst traffic from the ISF may exceed the capacity of an XAUI port and sessions may be dropped.

You can use the following command to configure bandwidth control between the ISF and XAUI ports. Enabling bandwidth control can smooth burst traffic and keep the XAUI ports from getting overwhelmed and dropping sessions.

Use the following command to enable bandwidth control:

```fortios
config system npu
  set sw-np-bandwidth {0G | 2G | 4G | 5G | 6G}
end
```

The default setting is 0G which means no bandwidth control. The other options limit the bandwidth to 2Gbps, 4Gbps and so on.
Increasing NP6 offloading capacity using link aggregation groups (LAGs)

NP6 processors can offload sessions received by interfaces in link aggregation groups (LAGs) (IEEE 802.3ad). A 802.3ad Link Aggregation and its management protocol, Link Aggregation Control Protocol (LACP) LAG combines more than one physical interface into a group that functions like a single interface with a higher capacity than a single physical interface. For example, you could use a LAG if you want to offload sessions on a 30 Gbps link by adding three 10-Gbps interfaces to the same LAG.

All offloaded traffic types are supported by LAGs, including IPsec VPN traffic. Just like with normal interfaces, traffic accepted by a LAG is offloaded by the NP6 processor connected to the interfaces in the LAG that receive the traffic to be offloaded. If all interfaces in a LAG are connected to the same NP6 processor, traffic received by that LAG is offloaded by that NP6 processor. The amount of traffic that can be offloaded is limited by the capacity of the NP6 processor.

If a FortiGate has two or more NP6 processors connected by an integrated switch fabric (ISF), you can use LAGs to increase offloading by sharing the traffic load across multiple NP6 processors. You do this by adding physical interfaces connected to different NP6 processors to the same LAG.

Adding a second NP6 processor to a LAG effectively doubles the offloading capacity of the LAG. Adding a third further increases offloading. The actual increase in offloading capacity may not actually be doubled by adding a second NP6 or tripled by adding a third. Traffic and load conditions and other factors may limit the actual offloading result.

The increase in offloading capacity offered by LAGs and multiple NP6s is supported by the integrated switch fabric (ISF) that allows multiple NP6 processors to share session information. Most FortiGate units with multiple NP6 processors also have an ISF. However, FortiGate models such as the 1000D, 2000E, and 2500E do not have an ISF. If you attempt to add interfaces connected to different NP6 processors to a LAG the system displays an error message.

There are also a few limitations to LAG NP6 offloading support for IPsec VPN:

- IPsec VPN anti-replay protection cannot be used if IPSec is configured on a LAG that has interfaces connected to multiple NP6 processors.
- Because the encrypted traffic for one IPsec VPN tunnel has the same 5-tuple, the traffic from one tunnel can only can be balanced to one interface in a LAG. This limits the maximum throughput for one IPsec VPN tunnel in an NP6 LAG group to 10Gbps.

NP6 processors and redundant interfaces

NP6 processors can offload sessions received by interfaces that are part of a redundant interface. You can combine two or more physical interfaces into a redundant interface to provide link redundancy. Redundant interfaces ensure connectivity if one physical interface, or the equipment on that interface, fails. In a redundant interface, traffic travels only over one interface at a time. This differs from an aggregated interface where traffic travels over all interfaces for distribution of increased bandwidth.

All offloaded traffic types are supported by redundant interfaces, including IPsec VPN traffic. Just like with normal interfaces, traffic accepted by a redundant interface is offloaded by the NP6 processor connected to the interfaces in the redundant interface that receive the traffic to be offloaded. If all interfaces in a redundant interface are connected to the same NP6 processor, traffic received by that redundant interface is offloaded by that NP6 processor. The amount of traffic that can be offloaded is limited by the capacity of the NP6 processor.
If a FortiGate has two or more NP6 processors connected by an integrated switch fabric (ISF), you can create redundant interfaces that include physical interfaces connected to different NP6 processors. However, with a redundant interface, only one of the physical interfaces is processing traffic at any given time. So you cannot use redundant interfaces to increase performance in the same way as you can with aggregate interfaces.

The ability to add redundant interfaces connected to multiple NP6s is supported by the integrated switch fabric (ISF) that allows multiple NP6 processors to share session information. Most FortiGate units with multiple NP6 processors also have an ISF. However, FortiGate models such as the 1000D, 2000E, and 2500E do not have an ISF. If you attempt to add interfaces connected to different NP6 processors to a redundant interface the system displays an error message.

### Configuring inter-VDOM link acceleration with NP6 processors

FortiGate units with NP6 processors include inter-VDOM links that can be used to accelerate inter-VDOM link traffic.

- For a FortiGate unit with two NP6 processors there are two accelerated inter-VDOM links, each with two interfaces:
  - `npu0_vlink`:
    - `npu0_vlink0`
    - `npu0_vlink1`
  - `npu1_vlink`:
    - `npu1_vlink0`
    - `npu1_vlink1`

These interfaces are visible from the GUI and CLI. For a FortiGate unit with NP6 interfaces, enter the following CLI command to display the NP6-accelerated inter-VDOM links:

```fortiOS
get system interface
...
== [ npu0_vlink0 ]
name: npu0_vlink0 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable type:
  physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy: disable
  mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-fragment:
  disable

== [ npu0_vlink1 ]
name: npu0_vlink1 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable type:
  physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy: disable
  mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-fragment:
  disable

== [ npu1_vlink0 ]
name: npu1_vlink0 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable type:
  physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy: disable
  mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-fragment:
  disable

== [ npu1_vlink1 ]
name: npu1_vlink1 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable type:
  physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy: disable
  mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-fragment:
  disable
...
```

By default the interfaces in each inter-VDOM link are assigned to the root VDOM. To use these interfaces to accelerate inter-VDOM link traffic, assign each interface in the pair to the VDOMs that you want to offload traffic between. For
example, if you have added a VDOM named New-VDOM to a FortiGate unit with NP4 processors, you can go to **System > Network > Interfaces** and edit the npu0-vlink1 interface and set the Virtual Domain to New-VDOM. This results in an accelerated inter-VDOM link between root and New-VDOM. You can also do this from the CLI:

```bash
config system interface
edit npu0-vlink1
  set vdom New-VDOM
end
```

### Using VLANs to add more accelerated inter-VDOM links

You can add VLAN interfaces to the accelerated inter-VDOM links to create inter-VDOM links between more VDOMs. For the links to work, the VLAN interfaces must be added to the same inter-VDOM link, must be on the same subnet, and must have the same VLAN ID.

For example, to accelerate inter-VDOM link traffic between VDOMs named Marketing and Engineering using VLANs with VLAN ID 100 go to **System > Network > Interfaces** and select Create New to create the VLAN interface associated with the Marketing VDOM:

<table>
<thead>
<tr>
<th>Name</th>
<th>Marketing-link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>VLAN</td>
</tr>
<tr>
<td>Interface</td>
<td>npu0_vlink0</td>
</tr>
<tr>
<td>VLAN ID</td>
<td>100</td>
</tr>
<tr>
<td>Virtual Domain</td>
<td>Marketing</td>
</tr>
<tr>
<td>IP/Network Mask</td>
<td>172.20.120.12/24</td>
</tr>
</tbody>
</table>

Create the inter-VDOM link associated with Engineering VDOM:

<table>
<thead>
<tr>
<th>Name</th>
<th>Engineering-link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>VLAN</td>
</tr>
<tr>
<td>Interface</td>
<td>npu0_vlink1</td>
</tr>
<tr>
<td>VLAN ID</td>
<td>100</td>
</tr>
<tr>
<td>Virtual Domain</td>
<td>Engineering</td>
</tr>
<tr>
<td>IP/Network Mask</td>
<td>172.20.120.22/24</td>
</tr>
</tbody>
</table>

Or do the same from the CLI:

```bash
config system interface
edit Marketing-link
  set vdom Marketing
  set ip 172.20.120.12/24
  set interface npu0_vlink0
  set vlanid 100
next
edit Engineering-link
  set vdom Engineering
  set ip 172.20.120.22/24
  set interface npu0_vlink1
```
 Confirm that the traffic is accelerated

Use the following CLI commands to obtain the interface index and then correlate them with the session entries. In the following example traffic was flowing between new accelerated inter-VDOM links and physical ports port1 and port 2 also attached to the NP6 processor.

```
set vlanid 100
```

```
**diagnose ip address list**
IP=172.31.17.76->172.31.17.76/255.255.255.255 index=5 devname=port1
IP=10.74.1.76->10.74.1.76/255.255.255.255 index=6 devname=port2
IP=172.20.120.12->172.20.120.12/255.255.255.255 index=55 devname=IVL-VLAN1_ROOT
IP=172.20.120.22->172.20.120.22/255.255.255.255 index=56 devname=IVL-VLAN1_VDOM1
```

```
**diagnose sys session list**
session info: proto=1 proto_state=00 duration=282 expire=24 timeout=0 session info: proto=1
  proto_state=00 duration=124 expire=59 timeout=0 flags=00000000 sockflag=00000000
  sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
origin->sink: org pre->post, reply pre->post dev=55->5/5->55 gwy=172.31.19.254/172.20.120.22
hook=post dir=org act=snat 10.74.2.87:768->10.2.2.2:8(172.31.17.76:62464)
hook=pre dir=reply act=dnat 10.2.2.2:62464->172.31.17.76:0(10.74.2.87:768)
misc=0 policy_id=4 id_policy_id=0 auth_info=0 chk_client_info=0 vd=0
serial=0000004e tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=160/218, ipid=218/160, vlan=32769/0
```

```
session info: proto=1 proto_state=00 duration=124 expire=20 timeout=0 flags=00000000 sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
origin->sink: org pre->post, reply pre->post dev=6->56/56->6 gwy=172.20.120.12/10.74.2.87
hook=pre dir=org act=snat 10.74.2.87:768->10.2.2.2:8(0.0.0.0:0)
hook=post dir=reply act=dnat 10.2.2.2:62464->10.74.2.87:768(0.0.0.0:0)
misc=0 policy_id=3 id_policy_id=0 auth_info=0 chk_client_info=0 vd=1
serial=0000004d tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=219/161, ipid=161/219, vlan=0/32769
```

```
total session 2
```
Disabling offloading IPsec Diffie-Hellman key exchange

You can use the following command to disable using ASIC offloading to accelerate IPsec Diffie-Hellman key exchange for IPsec ESP traffic. By default hardware offloading is used. For debugging purposes or other reasons you may want this function to be processed by software.

Use the following command to disable using ASIC offloading for IPsec Diffie-Hellman key exchange:

```
config system global
   set ipsec-asic-offload disable
end
```

Access control lists (ACLs)

Access Control Lists (ACLs) use NP6 offloading to 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. This feature is available on FortiGates with NP6 processors and is not supported by FortiGates with NP6lite processors.

The ACL feature is available only 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.

Use the following command to configure IPv4 ACL lists:

```
config firewall acl
   edit 0
      set status enable
      set interface <interface-name>
      set srcaddr <firewall-address>
      set dstaddr <firewall-address>
      set service <firewall-service>
   end
```

Use the following command to configure IPv6 ACL lists:

```
config firewall acl6
   edit 0
      set status enable
      set interface <interface-name>
      set srcaddr <firewall-address6>
      set dstaddr <firewall-address6>
      set service <firewall-service>
   end
```

Where:

- `<interface-name>` is the interface on which to apply the ACL. 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 ACL function is only supported on switch fabric driven interfaces.

- `<firewall-address>` `<firewall-address6>` can be any of the address types used by the FortiGate, including address ranges. 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. 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.
<firewall-service> the firewall service to block. Use ALL to block all services.

Configuring individual NP6 processors

You can use the config system np6 command to configure a wide range of settings for each of the NP6 processors in your FortiGate unit including enabling session accounting and adjusting session timeouts. As well you can set anomaly checking for IPv4 and IPv6 traffic.

You can also enable and adjust Host Protection Engine (HPE) to protect networks from DoS attacks by categorizing incoming packets based on packet rate and processing cost and applying packet shaping to packets that can cause DoS attacks.

The settings that you configure for an NP6 processor with the config system np6 command apply to traffic processed by all interfaces connected to that NP6 processor. This includes the physical interfaces connected to the NP6 processor as well as all subinterfaces, VLAN interfaces, IPsec interfaces, LAGs and so on associated with the physical interfaces connected to the NP6 processor.

Some of the options for this command apply anomaly checking for NP6 sessions in the same way as the command described in applies anomaly checking for NP4 sessions.

```
config system np6
edit <np6-processor-name>
  set low-latency-mode {disable | enable}
  set per-session-accounting {all-enable | disable | enable-by-log}
  set session-timeout-random-range <range>
  set garbage-session-collector {disable | enable}
  set session-collector-interval <range>
  set session-timeout-interval <range>
  set session-timeout-random-range <range>
  set session-timeout-fixed {disable | enable}
config hpe
  set tcpsyn-max <packets-per-second>
  set tcp-max <packets-per-second>
  set udp-max <packets-per-second>
  set icmp-max <packets-per-second>
  set sctp-max <packets-per-second>
  set esp-max <packets-per-second>
  set ip-frag-max <packets-per-second>
  set ip-others-max <packets-per-second>
  set arp-max <packets-per-second>
  set 12-others-max <packets-per-second>
  set enable-shaper {disable | enable}
```
config fp-anomaly

    set tcp-syn-fin {allow | drop | trap-to-host}
    set tcp_fin_noack {allow | drop | trap-to-host}
    set tcp_fin_only {allow | drop | trap-to-host}
    set tcp_no_flag {allow | drop | trap-to-host}
    set tcp_syn_data {allow | drop | trap-to-host}
    set tcp-winnuke {allow | drop | trap-to-host}
    set tcp-land {allow | drop | trap-to-host}
    set udp-land {allow | drop | trap-to-host}
    set icmp-land {allow | drop | trap-to-host}
    set icmp-frag {allow | drop | trap-to-host}
    set ipv4-land {allow | drop | trap-to-host}
    set ipv4-proto-err {allow | drop | trap-to-host}
    set ipv4-unknopt {allow | drop | trap-to-host}
    set ipv4-optrr {allow | drop | trap-to-host}
    set ipv4-optssrr {allow | drop | trap-to-host}
    set ipv4-optlsrr {allow | drop | trap-to-host}
    set ipv4-optstream {allow | drop | trap-to-host}
    set ipv4-optsecurity {allow | drop | trap-to-host}
    set ipv4-opptimestamp {allow | drop | trap-to-host}
    set ipv4-csum-err {drop | trap-to-host}
    set tcp-csum-err {drop | trap-to-host}
    set udp-csum-err {drop | trap-to-host}
    set icmp-csum-err {drop | trap-to-host}
    set ipv6-land {allow | drop | trap-to-host}
    set ipv6-proto-err {allow | drop | trap-to-host}
    set ipv6-unknopt {allow | drop | trap-to-host}
    set ipv6-saddr-err {allow | drop | trap-to-host}
    set ipv6-daddr-err {allow | drop | trap-to-host}
    set ipv6-optralert {allow | drop | trap-to-host}
    set ipv6-optjumbo {allow | drop | trap-to-host}
    set ipv6-opttunnel {allow | drop | trap-to-host}
    set ipv6-opthomeaddr {allow | drop | trap-to-host}
    set ipv6-optnsap {allow | drop | trap-to-host}
    set ipv6-optendpid {allow | drop | trap-to-host}
    set ipv6-optinvid {allow | drop | trap-to-host}

end

### Command syntax

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>low-latency-mode</td>
<td>Enable low-latency mode. In low latency mode the integrated switch fabric is bypassed. Low latency mode requires that packet enter and exit using the same NP6 processor. This option is only available for NP6 processors that can operate in low-latency mode, currently only np6_0 and np6_1 on the FortiGate-3700D and DX.</td>
<td>disable</td>
</tr>
<tr>
<td>(disable</td>
<td>enable)</td>
<td></td>
</tr>
<tr>
<td>per-session-accounting</td>
<td>Disable NP6 per-session accounting or enable it and control how it works. If set to enable-by-log (the default) NP6 per-session accounting is only enabled if firewall policies accepting offloaded</td>
<td>enable-by-log</td>
</tr>
<tr>
<td>(all-enable</td>
<td>disable</td>
<td>enable-by-log)</td>
</tr>
</tbody>
</table>
### Command Description Default

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>traffic have traffic logging enabled. If set to all-enable, NP6 per-session accounting is always enabled for all traffic offloaded by the NP6 processor.</td>
<td>Enabling per-session accounting can affect performance.</td>
<td></td>
</tr>
<tr>
<td>garbage-session-collector {disable</td>
<td>Enable deleting expired or garbage sessions.</td>
<td>disable</td>
</tr>
<tr>
<td>enable}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>session-collector-interval &lt;range&gt;</td>
<td>Set the expired or garbage session collector time interval in seconds. The range is 1 to 100 seconds.</td>
<td>64</td>
</tr>
<tr>
<td>session-timeout-interval &lt;range&gt;</td>
<td>Set the timeout for checking for and removing inactive NP6 sessions. The range is 0 to 1000 seconds.</td>
<td>40</td>
</tr>
<tr>
<td>session-timeout-random-range &lt;range&gt;</td>
<td>Set the random timeout for checking and removing inactive NP6 sessions. The range is 0 to 1000 seconds.</td>
<td>8</td>
</tr>
<tr>
<td>session-timeout-fixed {disable</td>
<td>Enable to force checking for and removing inactive NP6 sessions at the session-timeout-interval time interval. Set to disable (the default) to check for and remove inactive NP6 sessions at random time intervals.</td>
<td>disable</td>
</tr>
<tr>
<td>enable}</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### config hpe

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>hpe</td>
<td>Use the following options to use HPE to apply DDoS protection at the NP6 processor by limiting the number packets per second received for various packet types by each NP6 processor. This rate limiting is applied very efficiently because it is done in hardware by the NP6 processor.</td>
<td></td>
</tr>
<tr>
<td>enable-shaper {disable</td>
<td>Enable or disable HPE DDoS protection.</td>
<td>disable</td>
</tr>
<tr>
<td>enable}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tcpsyn-max</td>
<td>Limit the maximum number of TCP SYN packets received per second. The range is 10,000 to 4,000,000,000 pps. The default limits the number of packets per second to 5,000,000 pps.</td>
<td>5000000</td>
</tr>
<tr>
<td>tcp-max</td>
<td>Limit the maximum number of TCP packets received per second. The range is 10,000 to 4,000,000,000 pps. The default limits the number of packets per second to 5,000,000 pps.</td>
<td>5000000</td>
</tr>
<tr>
<td>udp-max</td>
<td>Limit the maximum number of UDP packets received per second. The range is 10,000 to</td>
<td>5000000</td>
</tr>
</tbody>
</table>
### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4,000,000,000 pps. The default limits the number of packets per second to 5,000,000 pps.</td>
<td></td>
</tr>
<tr>
<td>icmp-max</td>
<td>Limit the maximum number of ICMP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>sctp-max</td>
<td>Limit the maximum number of SCTP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>esp-max</td>
<td>Limit the maximum number of ESP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>ip-frag-max</td>
<td>Limit the maximum number of fragmented IP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>ip-others-max</td>
<td>Limit the maximum number of other types of IP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>arp-max</td>
<td>Limit the maximum number of ARP packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
<tr>
<td>l2-others-max</td>
<td>Limit the maximum number of other layer-2 packets received. The range is 10,000 to 4,000,000,000 pps. The default is 100,000 pps.</td>
<td>100000</td>
</tr>
</tbody>
</table>

### config fp-anomaly

**fp-anomaly**

Configure how the NP6 processor does traffic anomaly protection. In most cases you can configure the NP6 processor to allow or drop the packets associated with an attack and forward the packets that are associated with the attack to FortiOS (called trap-to-host). Selecting trap-to-host turns off NP6 anomaly protection for that anomaly. If you require anomaly protection but don't want to use the NP6 processor, you can select trap-to-host and enable anomaly protection with a DoS policy.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>tcp-syn-fin {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>tcp_fin_noack {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>tcp_fin_only {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>tcp_no_flag {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td>Default</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>tcp_syn_data {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>tcp-winnuke {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>tcp-land {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>udp-land {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>icmp-land {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>icmp-frag {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-land {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4 proto-err {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-unknopt {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-optrr {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-optssrr {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-optlsrr {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-optstream {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-optsecurity {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-opttimestamp {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv4-csum-err {drop</td>
<td>trap-to-host}</td>
<td>Detects IPv4 checksum errors.</td>
</tr>
<tr>
<td>tcp-csum-err {drop</td>
<td>trap-to-host}</td>
<td>Detects TCP checksum errors.</td>
</tr>
<tr>
<td>udp-csum-err {drop</td>
<td>trap-to-host}</td>
<td>Detects UDP checksum errors.</td>
</tr>
<tr>
<td>icmp-csum-err {drop</td>
<td>trap-to-host}</td>
<td>Detects ICMP checksum errors.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td>Default</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>trap-to-host</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ipv6-land {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-unknopt {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-saddr-err {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-daddr_err {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-optralert {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-optjumbo {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-opttunnel {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-opthomeaddr {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-optnsap {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-optendpid {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
<tr>
<td>ipv6-optinvid {allow</td>
<td>drop</td>
<td>trap-to-host}</td>
</tr>
</tbody>
</table>

**Enabling per-session accounting for offloaded NP6 and NP6lite sessions**

Per-session accounting is a logging feature that allows the FortiGate to report the correct bytes/pkt numbers per session for sessions offloaded to an NP6 or NP6lite processor. This information appears in traffic log messages as well as in FortiView. The following example shows the Sessions dashboard widget tracking SPU and nTurbo sessions. **Current sessions** shows the total number of sessions, **SPU** shows the percentage of these sessions that are SPU sessions and **Nturbo** shows the percentage that are nTurbo sessions.
You can hover over the SPU icon to see some information about the offloaded sessions.

You configure per-session accounting for each NP6 processor. For example, use the following command to enable per-session accounting for NP6_0 and NP6_1:

```
config system np6
  edit np6_0
    set per-session-accounting enable-by-log
  next
edit np6_1
  set per-session-accounting enable-by-log
end
```

If your FortiGate has NP6lite processors, you can use the following command to enable per-session accounting for all of the NP6lite processors in the FortiGate unit:

```
config system npu
  set per-session-accounting enable-by-log
end
```

The option, `enable-by-log` enables per-session accounting for offloaded sessions with traffic logging enabled and `all-enable` enables per-session accounting for all offloaded sessions.

By default, `per-session-accounting` is set to `enable-by-log`, which results in per-session accounting being turned on when you enable traffic logging in a policy.

Per-session accounting can affect offloading performance. So you should only enable per-session accounting if you need the accounting information.

Enabling per-session accounting does not provide traffic flow data for sFlow or NetFlow.

**Configuring NP6 session timeouts**

For NP6 traffic, FortiOS refreshes an NP6 session's lifetime when it receives a session update message from the NP6 processor. To avoid session update message congestion, these NP6 session checks are performed all at once after a
random time interval and all of the update messages are sent from the NP6 processor to FortiOS at once. This can result in fewer messages being sent because they are only sent at random time intervals instead of every time a session times out.

In fact, if your NP6 processor is processing a lot of short lived sessions, it is recommended that you use the default setting of random checking every 8 seconds to avoid very bursty session updates. If the time between session updates is very long and very many sessions have been expired between updates a large number of updates will need to be done all at once.

You can use the following command to set the random time range.

```fortigate
config system np6
edit <np6-processor-name>
    set session-timeout-fixed disable
    set session-timeout-random-range 8
end
```

This is the default configuration. The random timeout range is 1 to 1000 seconds and the default range is 8. So, by default, NP6 sessions are checked at random time intervals of between 1 and 8 seconds. So sessions can be inactive for up to 8 seconds before they are removed from the FortiOS session table.

If you want to reduce the amount of checking you can increase the session-timeout-random-range. This could result in inactive sessions being kept in the session table longer. But if most of your NP6 sessions are relatively long this shouldn't be a problem.

You can also change this session checking to a fixed time interval and set a fixed timeout:

```fortigate
config system np6
edit <np6-processor-name>
    set session-timeout-fixed enable
    set session-timeout-fixed 40
end
```

The fixed timeout default is every 40 seconds and the range is 1 to 1000 seconds. Using a fixed interval further reduces the amount of checking that occurs.

You can select random or fixed updates and adjust the time intervals to minimize the refreshing that occurs while still making sure inactive sessions are deleted regularly. For example, if an NP6 processor is processing sessions with long lifetimes you can reduce checking by setting a relatively long fixed timeout.

### Configure the number of IPsec engines NP6 processors use

NP6 processors use multiple IPsec engines to accelerate IPsec encryption and decryption. In some cases out of order ESP packets can cause problems if multiple IPsec engines are running. To resolve this problem you can configure all of the NP6 processors to use fewer IPsec engines.

Use the following command to change the number of IPsec engines used for decryption (ipsec-dec-subengine-mask) and encryption (ipsec-enc-subengine-mask). These settings are applied to all of the NP6 processors in the FortiGate unit.

```fortigate
config system npu
    set ipsec-dec-subengine-mask <engine-mask>
    set ipsec-enc-subengine-mask <engine-mask>
end
```
NP6 and NP6lite acceleration

<engine-mask> is a hexadecimal number in the range 0x01 to 0xff where each bit represents one IPsec engine. The default <engine-mask> for both options is 0xff which means all IPsec engines are used. Add a lower <engine-mask> to use fewer engines. You can configure different engine masks for encryption and decryption.

**Stripping clear text padding and IPsec session ESP padding**

In some situations, when clear text or ESP packets in IPsec sessions may have large amounts of layer 2 padding, the NP6 IPsec engine may not be able to process them and the session may be blocked.

If you notice dropped IPsec sessions, you could try using the following CLI options to cause the NP6 processor to strip clear text padding and ESP padding before send the packets to the IPsec engine. With padding stripped, the session can be processed normally by the IPsec engine.

Use the following command to strip ESP padding:

```config system npu
    set strip-esp-padding enable
    set strip-clear-text-padding enable
end
```

Stripping clear text and ESP padding are both disabled by default.

**Disable NP6 CAPWAP offloading**

By default and where possible, managed FortiAP and FortiLink CAPWAP sessions are offloaded to NP6 processors. You can use the following command to disable CAWAP session offloading:

```config system npu
    set capwap-offload disable
end
```

**Optionally disable NP6 offloading of traffic passing between 10Gbps and 1Gbps interfaces**

Due to NP6 internal packet buffer limitations, some offloaded packets received at a 10Gbps interface and destined for a 1Gbps interface can be dropped, reducing performance for TCP and IP tunnel traffic. If you experience this performance reduction, you can use the following command to disable offloading sessions passing from 10Gbps interfaces to 1Gbps interfaces:

```config system npu
    set host-shortcut-mode host-shortcut
end
```

Select host-shortcut to stop offloading TCP and IP tunnel packets passing from 10Gbps interfaces to 1Gbps interfaces. TCP and IP tunnel packets passing from 1Gbps interfaces to 10Gbps interfaces are still offloaded as normal.

If host-shortcut is set to the default bi-directional setting, packets in both directions are offloaded.
This option is only available if your FortiGate has 10G and 1G interfaces accelerated by NP6 processors.

**Offloading RDP traffic**

FortiOS supports NP6 offloading of Reliable Data Protocol (RDP) traffic. RDP is a network transport protocol that optimizes remote loading, debugging, and bulk transfer of images and data. RDP traffic uses Assigned Internet Protocol number 27 and is defined in RFC 908 and updated in RFC 1151. If your network is processing a lot of RDP traffic, offloading it can improve overall network performance.

You can use the following command to enable or disable NP6 RDP offloading. RDP offloading is enabled by default.

```
config system npu
  set rdp-offload {disable | enable}
end
```

**NP6 session drift**

In some cases, sessions processed by NP6 processors may fail to be deleted leading to a large number of idle sessions. This is called session drift. You can use SNMP to be alerted when the number of idle sessions becomes high. SNMP also allows you to see which NP6 processor has the abnormal number of idle sessions and you can use a diagnose command to delete them.

The following MIB fields allow you to use SNMP to monitor session table information for NP6 processors including drift for each NP6 processor:

```
FORTINET-FORTIGATE-MIB::fgNPUNumber.0 = INTEGER: 2
FORTINET-FORTIGATE-MIB::fgNPUName.0 = STRING: NP6
FORTINET-FORTIGATE-MIB::fgNPUDrvDriftSum.0 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fgNPUIIndex.0 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fgNPUIIndex.1 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fgNPUSessionTblSize.0 = Gauge32: 33554432
FORTINET-FORTIGATE-MIB::fgNPUSessionTblSize.1 = Gauge32: 33554432
FORTINET-FORTIGATE-MIB::fgNPUSessionCount.0 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fgNPUSessionCount.1 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fgNPUDrvDrift.0 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fgNPUDrvDrift.1 = INTEGER: 0
```

You can also use the following diagnose command to determine of drift is occurring:

```
diagnose npu np6 sse-drift-summary
NPU    drv-drift
------- -------
np6_0  0
np6_1  0
------- -------
Sum    0
------- -------
```
The command output shows a drift summary for all the NP6 processors in the system, and shows the total drift. Normally the sum is 0. The previous command output, from a FortiGate-1500D, shows that the 1500D's two NP6 processors are not experiencing any drift.

If the sum is not zero, then extra idle sessions may be accumulating. You can use the following command to delete those sessions:

```bash
diagnose npu np6 sse-purge-drift <np6_id> [<time>]
```

Where `<np6_id>` is the number (starting with NP6_0 with a np6_id of 0) of the NP6 processor for which to delete idle sessions in. `<time>` is the age in seconds of the idle sessions to be deleted. All idle sessions this age and older are deleted. The default time is 300 seconds.

The `diagnose npu np6 sse-stats <np6_id>` command output also includes a `drv-drift` field that shows the total drift for one NP6 processor.

### Optimizing FortiGate-3960E and 3980E IPsec VPN performance

You can use the following command to configure outbound hashing to improve IPsec VPN performance for the FortiGate-3960E and 3980E. If you change these settings, to make sure they take affect, you should reboot your device.

```bash
config system np6
edit np6_0
  set ipsec-outbound-hash {disable | enable}
  set ipsec-ob-hash-function {switch-group-hash | global- hash | global-hash-weighted | round-robin-switch-group | round-robin-global}
end
```

Where:

- `ipsec-outbound-hash` is disabled by default. If you enable it you can set `ipsec-ob-hash-function` as follows:

  - `switch-group-hash` (the default) distribute outbound IPsec Security Association (SA) traffic to NP6 processors connected to the same switch as the interfaces that received the incoming traffic. This option, keeps all traffic on one switch and the NP6 processors connected to that switch, to improve performance.

  - `global-hash` distribute outbound IPsec SA traffic among all NP6 processors.

  - `global-hash-weighted` distribute outbound IPsec SA traffic from switch 1 among all NP6 processors with more sessions going to the NP6s connected to switch 0. This options is only recommended for the FortiGate-3980E because it is designed to weigh switch 0 hider to send more sessions to switch 0 which on the FortiGate-3980E has more NP6 processors connected to it. On the FortiGate-3960E both switches have the same number of NP6s so for best performance one switch shouldn't have a higher weight.

  - `round-robin-switch-group` round-robin distribution of outbound IPsec SA traffic among the NP6 processors connected to the same switch.

  - `round-robin-global` round-robin distribution of outbound IPsec SA traffic among all NP6 processors.
FortiGate-3960E and 3980E support for high throughput traffic streams

FortiGate devices with multiple NP6 processors support high throughput by distributing sessions to multiple NP6 processors. However, default ISF hash-based load balancing has some limitations for single traffic streams or flows that use more than 10Gbps of bandwidth. Normally, the ISF sends all of the packets in a single traffic stream over the same 10Gbps interface to an NP6 processor. If a single traffic stream is larger than 10Gbps, packets are also sent to 10Gbps interfaces that may be connected to the same NP6 or to other NP6s. Because the ISF uses hash-based load balancing, this can lead to packets being processed out of order and other potential drawbacks.

You can configure the FortiGate-3960E and 3980E to support single traffic flows that are larger than 10Gbps. To enable this feature, you can assign interfaces to round robin groups using the following configuration. If you assign an interface to a Round Robin group, the ISF uses round-robin load balancing to distribute incoming traffic from one stream to multiple NP6 processors. Round-robin load balancing prevents the potential problems associated with hash-based load balancing of packets from a single stream.

```plaintext
config system npu
    config port-npu-map
        edit <interface>
            set npu-group-index <npu-group>
        end
    end

<interface> is the name of an interface that receives or sends large traffic streams.

<npu-group> is the number of an NPU group. To enable round-robin load balancing select a round-robin NPU group. Use ? to see the list of NPU groups. The output shows which groups support round robin load balancing. For example, the following output shows that NPU group 30 supports round robin load balancing to NP6 0 to 7.

set npu-group-index ?
index: npu group
  0 : NP#0-7
  2 : NP#0
  3 : NP#1
  4 : NP#2
  5 : NP#3
  6 : NP#4
  7 : NP#5
  8 : NP#6
  9 : NP#7
 10 : NP#0-1
 11 : NP#2-3
 12 : NP#4-5
 13 : NP#6-7
 14 : NP#0-3
 15 : NP#4-7
30 : NP#0-7 - Round Robin

For example, use the following command to assign port1, port2, port17 and port18 to NPU group 30.

config system npu
    config port-npu-map
        edit port1
            set npu-group-index 30
        next
        edit port2
```
Recalculating packet checksums if the iph.reserved bit is set to 0

NP6 and the NP6lite processors clear the iph.flags.reserved bit. This results in the packet checksum becoming incorrect because by default the packet is changed but the checksum is not recalculated. Since the checksum is incorrect these packets may be dropped by the network stack. You can enable this option to cause the system to re-calculate the checksum. Enabling this option may cause a minor performance reduction. This option is disabled by default.

To enabled checksum recalculation for packets with the iph.flags.reserved header:

```
config system npu
    set iph-rsvd-re-cksum enable
end
```

NP6 IPsec engine status monitoring

Use the following command to configure NP6 IPsec engine status monitoring.

```
config monitoring np6-ipsec-engine
    set status enable
    set interval 5
    set threshold 10 10 8 8 6 6 4 4
end
```

Use this command to configure NP6 IPsec engine status monitoring. NP6 IPsec engine status monitoring writes a system event log message if the IPsec engines in an NP6 processor become locked after receiving malformed packets.

If an IPsec engine becomes locked, that particular engine can no longer process IPsec traffic, reducing the capacity of the NP6 processor. The only way to recover from a locked IPsec engine is to restart the FortiGate device. If you notice an IPsec performance reduction over time on your NP6 accelerated FortiGate device, you could enable NP6 IPsec engine monitoring and check log messages to determine if your NP6 IPsec engines are becoming locked.

To configure IPsec engine status monitoring you set status to enable and then configure the following options:

```
interval
```

Set the IPsec engine status check time interval in seconds (range 1 to 60 seconds, default = 1).

```
threshold <np6_0-threshold> <np6_1-threshold>...<np6_7-threshold>
```
Set engine status check thresholds. An NP6 processor has eight IPsec engines and you can set a threshold for each engine. NP6 IPsec engine status monitoring regularly checks the status of all eight engines in all NP6 processors in the FortiGate device.

Each threshold can be an integer between 1 and 255 and represents the number of times the NP6 IPsec engine status check detects that the NP6 processor is busy before generating a log message.

The default thresholds are 15 15 12 12 8 8 5 5. Any IPsec engine exceeding its threshold triggers the event log message. The default interval and thresholds have been set to work for most network topologies based on a balance of timely reporting a lock-up and accuracy and on how NP6 processors distribute sessions to their IPsec engines. The default settings mean:

- If engine 1 or 2 are busy for 15 checks (15 seconds) trigger an event log message.
- If engine 3 or 4 are busy for 12 checks (15 seconds) trigger an event log message.
- And so on.

NP6 IPsec engine monitoring writes three levels of log messages:

- Information if an IPsec engine is found to be busy.
- Warning if an IPsec engine exceeds a threshold.
- Critical if a lockup is detected, meaning an IPsec engine continues to exceed its threshold.

The log messages include the NP6 processor and engine affected.
FortiGate NP6 architectures

This chapter shows the NP6 architecture for the all FortiGate models that include NP6 processors.

FortiGate-300D fast path architecture

The FortiGate-300D includes one NP6 processor connected to four 1Gb RJ-45 Ethernet ports (port1-4) and four 1Gb SFP interfaces (port5-port8).

You can use the following get command to display the FortiGate-300D NP6 configuration. The command output shows one NP6 named NP6_0 and the interfaces (ports) connected to it. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
Chip  XAUI Ports  Max  Cross-chip
      Speed offloading
------- ---- -------- ------ ----------
np6_0  0            1G  Yes
1  port5  1G  Yes
1  port7  1G  Yes
1  port8  1G  Yes
1  port6  1G  Yes
1  port3  1G  Yes
1  port4  1G  Yes
1  port1  1G  Yes
1  port2  1G  Yes
2
1
3
------- ---- -------- ------ ----------
```
FortiGate-300E and 301E fast path architecture

The FortiGate-300E and 301E each include one NP6 processor. All front panel data interfaces (port1 to 28, S1, S2, VW1, and VW2) connect to the NP6 processor. So all supported traffic passing between any two data interfaces can be offloaded.

The following diagram also shows the XAUI and QSGMII port connections between the NP6 processor and the front panel interfaces.

You can use the following get command to display the FortiGate-300E or 301E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td>0 port1</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port2</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port3</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port4</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port5</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port6</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port7</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 port8</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port9</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port10</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port11</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port12</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port13</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port14</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port15</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 port16</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 port17</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 port18</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 port19</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 port20</td>
<td>1G</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
FortiGate NP6 architectures

The FortiGate-400D includes one NP6 processor connected to eight 1Gb SFP interfaces (port1-port8) and eight 1Gb RJ-45 Ethernet ports (port9-16).

You can use the following get command to display the FortiGate-400D NP6 configuration. The command output shows one NP6 named NP6_0 and the interfaces (ports) connected to it. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip    XAUl Ports Max Cross-chip Speed offloading
-------- ------ ------- ------ -------- -------- --------
np6_0     0
1 port10  1G     Yes  1 port9  1G     Yes  1 port12  1G     Yes  1 port11  1G     Yes
1 port14  1G     Yes  1 port13  1G     Yes  1 port16  1G     Yes  1 port15  1G     Yes
```
FortiGate NP6 architectures

The FortiGate-400E and 401E each include one NP6 processor. All front panel data interfaces (port1 to 28, S1, S2, VW1, and VW2) connect to the NP6 processor. So all supported traffic passing between any two data interfaces can be offloaded.

The following diagram also shows the XAUI and QSGMII port connections between the NP6 processor and the front panel interfaces.

You can use the following get command to display the FortiGate-400E or 401E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip Offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td>0</td>
<td>port1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port2</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port3</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port4</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port5</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port6</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port7</td>
<td>1G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

FortiOS Handbook - Hardware Acceleration
FortiGate-500D fast path architecture

The FortiGate-500D includes one NP6 processor connected to eight 1Gb SFP interfaces (port1-port8) and eight 1Gb RJ-45 Ethernet ports (port9-16).

You can use the following get command to display the FortiGate-500D NP6 configuration. The command output shows one NP6 named NP6_0 and the interfaces (ports) connected to it. You can also use the `diagnose npu np6 port-list` command to display this information.
### FortiGate-500E and 501E fast path architecture

The FortiGate-500E and 501E each include one NP6 processor. All front panel data interfaces (port 1 to 12, S1, S2, VW1, VW2, X1, and X2) connect to the NP6 processor. So all supported traffic passing between any two data interfaces can be offloaded.

The following diagram also shows the QSGMII and XAUI port connections between the NP6 processor and the front panel interfaces.

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port10</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port9</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port12</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port11</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port14</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port13</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port16</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port15</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port5</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port7</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port8</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port6</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port3</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port4</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port2</td>
<td>1G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

2

---

get hardware npu np6 port-list

**Diagram:**

- Port 1-8: 2x QSGMII
- Port 9-12, S1, S2, VW1, VW2: 2x QSGMII
- X1 and X2: 2x XAUI
- System Bus
- NP6_0
- CP9
- CPU
- CP9

FortiOS Handbook - Hardware Acceleration

Fortinet Technologies Inc.
You can use the following get command to display the FortiGate-500E or 501E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td>x1</td>
<td>0</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port3</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port4</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port5</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port6</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port7</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port8</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port9</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port10</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port11</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port12</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>s1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>s2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>vw1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>vw2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>x2</td>
<td>10G</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**FortiGate-600E and 601E fast path architecture**

The FortiGate-600E and 601E each include one NP6 processor. All front panel data interfaces (port 1 to 12, S1, S2, VW1, VW2, X1, and X2) connect to the NP6 processor. So all supported traffic passing between any two data interfaces can be offloaded. The FortiGate-600E and 601E have the same fast path architecture as the FortiGate-500E and 501E. The 600E/601E models include a next-generation CPU for improved performance.

The following diagram also shows the QSGMII and XAUI port connections between the NP6 processor and the front panel interfaces.
You can use the following `get` command to display the FortiGate-600E or 601E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td>x1</td>
<td>1</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port3</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port4</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port5</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port6</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port7</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port8</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port9</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port10</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port11</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>port12</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>s1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>s2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>vw1</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>vw2</td>
<td>1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>x2</td>
<td>2</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

---

**FortiGate-600D fast path architecture**

The FortiGate-600D includes one NP6 processor connected to eight 1Gb SFP interfaces (port1-port8) and eight 1Gb RJ-45 Ethernet ports (port9-16) and two 10Gb SFP+ interfaces (port17 and port18).
You can use the following `get` command to display the FortiGate-600D NP6 configuration. The command output shows one NP6 named NP6_0 and the interfaces (ports) connected to it. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip XAUI Ports Max Cross-chip Speed offloading
------- ----- -------- ----- -------
np6_0   0        1 port10 1G Yes
         1 port9 1G Yes
         1 port12 1G Yes
         1 port11 1G Yes
         1 port14 1G Yes
         1 port13 1G Yes
         1 port16 1G Yes
         1 port15 1G Yes
         1 port5 1G Yes
         1 port7 1G Yes
         1 port8 1G Yes
         1 port6 1G Yes
         1 port3 1G Yes
         1 port4 1G Yes
         1 port1 1G Yes
         1 port2 1G Yes
         2 port17 10G Yes
         3 port18 10G Yes
------- ----- -------- ----- -------
```

**FortiGate-800D fast path architecture**

The FortiGate-800D includes one NP6 processor connected through an integrated switch fabric to all of the FortiGate-800D network interfaces. This hardware configuration supports NP6-accelerated fast path offloading for sessions between any of the FortiGate-800D interfaces.
You can use the following `get` command to display the FortiGate-800D NP6 configuration. The command output shows one NP6 named NP6_0. The output also shows all of the FortiGate-800D interfaces (ports) connected to NP6_0. You can also use the `diagnose npu np6 port-list` command to display this information.

```plaintext
get hardware npu np6 port-list
Chip   XAUI Ports Max Cross-chip Speed offloading
-------- ---- -------- ------ ----------
np6_0   0  port31  1G   Yes
       1  wan1  1G   Yes
       1  port1  1G   Yes
       1  wan2  1G   Yes
       1  port2  1G   Yes
       1  port3  1G   Yes
       1  port4  1G   Yes
       1  port5  1G   Yes
       1  port6  1G   Yes
       1  port30 1G   Yes
       1  port29 1G   Yes
       1  port28 1G   Yes
       1  port27 1G   Yes
       1  port26 1G   Yes
       1  port25 1G   Yes
       1  port24 1G   Yes
       1  port23 1G   Yes
       2  port7  1G   Yes
       2  port8  1G   Yes
       2  port9  1G   Yes
       2  port10 1G   Yes
       2  port11 1G   Yes
       2  port12 1G   Yes
       2  port13 1G   Yes
       2  port14 1G   Yes
       2  port15 1G   Yes
       2  port16 1G   Yes
       2  port17 1G   Yes
       2  port18 1G   Yes
       2  port19 1G   Yes
```
FortiGate NP6 architectures

2 port20 1G Yes
2 port21 1G Yes
2 port22 1G Yes
3 port32 10G Yes

FortiGate-900D fast path architecture

The FortiGate-900D includes two NP6 processors that are not connected by an integrated switch fabric (ISF). Without an ISF, traffic through a FortiGate-900D could experience lower latency than traffic through similar hardware with an ISF. The NP6 processors are connected to network interfaces as follows:

- Eight 1Gb SFP interfaces (port17-port24), eight 1Gb RJ-45 Ethernet interfaces (port25-32) and one 10Gb SFP+ interface (portB) share connections to the first NP6 processor.
- Eight 1Gb SFP interfaces (port1-port8), eight RJ-45 Ethernet interfaces (port9-16) and one 10Gb SFP+ interface (portA) share connections to the second NP6 processor.

Because the FortiGate-900D does not have an ISF you cannot create Link Aggregation Groups (LAGs) that include interfaces connected to both NP6 processors.

- Eight 1Gb SFP interfaces (port17-port24), eight 1Gb RJ-45 Ethernet interfaces (port25-32) and one 10Gb SFP+ interface (portB) share connections to the first NP6 processor.
- Eight 1Gb SFP interfaces (port1-port8), eight RJ-45 Ethernet interfaces (port9-16) and one 10Gb SFP+ interface (portA) share connections to the second NP6 processor.

You can use the following get command to display the FortiGate-900D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
Chip XAUI Ports Max Cross-chip
Speed offloading
------- ----- -------- --------
np6_0   0
1 port17 1G Yes
1 port18 1G Yes
```
FortiGate-1000D fast path architecture

The FortiGate-1000D includes two NP6 processors that are not connected by an integrated switch fabric (ISF). The NP6 processors are connected to network interfaces as follows:

Because the FortiGate-1000D does not have an ISF you cannot create Link Aggregation Groups (LAGs) or redundant interfaces that include interfaces connected to both NP6 processors.

- Eight 1Gb SFP interfaces (port17-port24), eight 1Gb RJ-45 Ethernet interfaces (port25-32) and one 10Gb SFP+ interface (portB) share connections to the first NP6 processor.
- Eight 1Gb SFP interfaces (port1-port8), eight RJ-45 Ethernet interfaces (port9-16) and one 10Gb SFP+ interface (portA) share connections to the second NP6 processor.
You can use the following `get` command to display the FortiGate-1000D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip   XAUI Ports  Max  Cross-chip Speed offloading
------- ------ ------- ----- ------------
np6_0   0
1      port1    1G    Yes
1      port18   1G    Yes
1      port19   1G    Yes
1      port20   1G    Yes
1      port21   1G    Yes
1      port22   1G    Yes
1      port23   1G    Yes
1      port24   1G    Yes
1      port27   1G    Yes
1      port28   1G    Yes
1      port25   1G    Yes
1      port26   1G    Yes
1      port31   1G    Yes
1      port32   1G    Yes
1      port29   1G    Yes
1      port30   1G    Yes
2      portB    10G  Yes
3
------- ------ ------- ----- ------------
np6_1   0
1      port1    1G    Yes
1      port2    1G    Yes
1      port3    1G    Yes
1      port4    1G    Yes
1      port5    1G    Yes
```
<table>
<thead>
<tr>
<th>Port</th>
<th>Type</th>
<th>Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>port6</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port7</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port8</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port11</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port12</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port9</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port10</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port15</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port16</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port13</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port14</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>portA</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>port25</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port26</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port27</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port28</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port29</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port30</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port31</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port32</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>port33</td>
<td>10G</td>
<td>No</td>
</tr>
<tr>
<td>port34</td>
<td>10G</td>
<td>No</td>
</tr>
<tr>
<td>port35</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>port36</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**FortiGate-1200D fast path architecture**

The FortiGate-1200D features two NP6 processors both connected to an integrated switch fabric.

- Eight SFP 1Gb interfaces (port1-port8), eight RJ-45 Ethernet ports (port17-24) and two SFP+ 10Gb interfaces (port33 and port34) share connections to the first NP6 processor.
- Eight SFP 1Gb interfaces (port9-port16), eight RJ-45 Ethernet ports (port25-32) and two SFP+ 10Gb interfaces (port35-port36) share connections to the second NP6 processor.
You can use the following get command to display the FortiGate-1200D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
Chip  XAUI  Ports  Max  Cross-chip
      Speed offloading
------- ------ ---------- ------ ------------------------
np6_0  0     port33  10G  Yes
       1     port34  10G  Yes
       2     port1   1G   Yes
       2     port3   1G   Yes
       2     port5   1G   Yes
       2     port7   1G   Yes
       2     port17  1G   Yes
       2     port19  1G   Yes
       2     port21  1G   Yes
       2     port23  1G   Yes
       3     port2   1G   Yes
       3     port4   1G   Yes
       3     port6   1G   Yes
       3     port8   1G   Yes
       3     port18  1G   Yes
       3     port20  1G   Yes
       3     port22  1G   Yes
       3     port24  1G   Yes
np6_1  0     port35  10G  Yes
       1     port36  10G  Yes
       2     port9   1G   Yes
       2     port11  1G   Yes
       2     port13  1G   Yes
       2     port15  1G   Yes
       2     port25  1G   Yes
       2     port27  1G   Yes
       2     port29  1G   Yes
       2     port31  1G   Yes
       3     port10  1G   Yes
       3     port12  1G   Yes
       3     port14  1G   Yes
       3     port16  1G   Yes
       3     port26  1G   Yes
       3     port28  1G   Yes
       3     port30  1G   Yes
       3     port32  1G   Yes
------- ------ ---------- ------ ------------------------
```

**FortiGate-1500D fast path architecture**

The FortiGate-1500D features two NP6 processors both connected to an integrated switch fabric.
- Eight SFP 1Gb interfaces (port1-port8), eight RJ-45 1Gb Ethernet interfaces (port17-24) and four SFP+ 10Gb interfaces (port33-port36) share connections to the first NP6 processor.

- Eight SFP 1Gb interfaces (port9-port16), eight RJ-45 1Gb Ethernet interfaces (port25-32) and four SFP+ 10Gb interfaces (port37-port40) share connections to the second NP6 processor.

You can use the following `get` command to display the FortiGate-1500D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip   XAUI Ports  Max Speed Cross-chip Speed offloading
------- ---- ------- ------ ---------------------
np6_0  0    port1  1G  Yes
       0    port5  1G  Yes
       0    port17 1G  Yes
       0    port21 1G  Yes
       0    port33 10G Yes
       1    port2  1G  Yes
       1    port6  1G  Yes
       1    port18 1G  Yes
       1    port22 1G  Yes
       1    port34 10G Yes
       2    port3  1G  Yes
       2    port7  1G  Yes
       2    port19 1G  Yes
       2    port23 1G  Yes
       2    port35 10G Yes
```
### FortiGate-1500DT fast path architecture

The FortiGate-1500DT features two NP6 processors both connected to an integrated switch fabric. The FortiGate-1500DT has the same hardware configuration as the FortiGate-1500D, but with the addition of newer CPUs and a slightly different interface configuration.

The FortiGate-1500DT includes the following interfaces and NP6 processors:

- Eight SFP 1Gb interfaces (port1-port8), eight RJ-45 1Gb Ethernet interfaces (port17-24) and four RJ-45 10Gb Ethernet interfaces (port33-port36) share connections to the first NP6 processor.
- Eight SFP 1Gb interfaces (port9-port16), eight RJ-45 1Gb Ethernet ports (port25-32) and four SFP+ 10Gb interfaces (port37-port40) share connections to the second NP6 processor.

<table>
<thead>
<tr>
<th>Port</th>
<th>Type</th>
<th>Speed</th>
<th>Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>port4</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>port8</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>20</td>
<td>port20</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>24</td>
<td>port24</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>36</td>
<td>port36</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>port9</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>13</td>
<td>port13</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>25</td>
<td>port25</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>29</td>
<td>port29</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>37</td>
<td>port37</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>port10</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>14</td>
<td>port14</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>26</td>
<td>port26</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>30</td>
<td>port30</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>38</td>
<td>port38</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>port11</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>port15</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>27</td>
<td>port27</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>31</td>
<td>port31</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>39</td>
<td>port39</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>port12</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>16</td>
<td>port16</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>28</td>
<td>port28</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>32</td>
<td>port32</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>40</td>
<td>port40</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>
You can use the following get command to display the FortiGate-1500DT NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>port1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>port5</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>port17</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>port21</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>port33</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port2</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port6</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port18</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port22</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>port34</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>port3</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>port7</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>port19</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>port23</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>port35</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>port4</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>port8</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>port20</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>port24</td>
<td>1G</td>
<td>Yes</td>
</tr>
</tbody>
</table>
FortiGate NP6 architectures

<table>
<thead>
<tr>
<th>np6</th>
<th>port</th>
<th>speed</th>
<th>status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>9</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>13</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>25</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>29</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>0</td>
<td>37</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>26</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>30</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>38</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>27</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>31</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>39</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>16</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>28</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

FortiGate-2000E fast path architecture

The FortiGate-2000E includes three NP6 processors in an NP Direct configuration. The NP6 processors connected to the 10GigE ports are also in a low latency NP Direct configuration. Because of NP Direct, you cannot create Link Aggregation Groups (LAGs) or redundant interfaces between interfaces connected to different NP6s. As well, traffic will only be offloaded if it enters and exits the FortiGate on interfaces connected to the same NP6.

The NP6s are connected to network interfaces as follows:

- NP6_0 is connected to four 10GigE SFP+ interfaces (port33 to port36) in a low latency configuration.
- NP6_1 is connected to thirty-two 10/100/1000BASE-T interfaces (port1 to port32).
- NP6_2 is connected to two 10GigE SFP+ (port37 and port38) in a low latency configuration.

The following diagram also shows the XAUI and QSGMII port connections between the NP6 processors and the front panel interfaces and the aggregate switch for the thirty-two 10/100/1000BASE-T interfaces.
You can use the following get command to display the FortiGate-2000E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

get hardware npu np6 port-list

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip Offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_1</td>
<td>port1 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port5 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port9 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port13 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port17 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port21 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port25 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port29 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port2 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port6 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port10 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port14 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port18 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port22 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port26 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port30 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port3 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port7 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port11 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port15 1G</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>port19 1G</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
FortiGate-2500E fast path architecture

The FortiGate-2500E includes four NP6 processors in an NP Direct configuration. The NP6 processors connected to the 10GigE ports are also in a low latency NP Direct configuration. Because of NP Direct, you cannot create Link Aggregation Groups (LAGs) or redundant interfaces between interfaces connected to different NP6s. As well, traffic will only be offloaded if it enters and exits the FortiGate on interfaces connected to the same NP6.

The NP6s are connected to network interfaces as follows:

- NP6_0 is connected to four 10GigE SFP+ interfaces (port37 to port40) in a low latency configuration.
- NP6_1 is connected to thirty-two 10/100/1000BASE-T interfaces (port1 to port32).
- NP6_2 is connected to two 10GigE SFP+ interfaces (port41 and port42) and two 10Gig fiber bypass interfaces (port43 and port44) in a low latency configuration.
- NP6_3 is connected to four 10GigE SFP+ interfaces (port33 to port36) in a low latency configuration.

The following diagram also shows the XAUI and QSGMII port connections between the NP6 processors and the front panel interfaces and the aggregate switch for the thirty-two 10/100/1000BASE-T interfaces.
You can use the following get command to display the FortiGate-2500E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip   XAUI Ports  Max  Cross-chip
       Speed  offloading
--------  ------  ---------  ----
np6_1    0  port1  1G  No
         0  port5  1G  No
         0  port9  1G  No
         0  port13 1G  No
         0  port17 1G  No
         0  port21 1G  No
         0  port25 1G  No
         0  port29 1G  No
         1  port2  1G  No
         1  port6  1G  No
         1  port10 1G  No
         1  port14 1G  No
         1  port18 1G  No
         1  port22 1G  No
         1  port26 1G  No
         1  port30 1G  No
         2  port3  1G  No
         2  port7  1G  No
         2  port11 1G  No
         2  port15 1G  No
         2  port19 1G  No
         2  port23 1G  No
         2  port27 1G  No
```
FortiGate NP6 architectures

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>port31</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port4</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port8</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port12</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port16</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port20</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port24</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port28</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>port32</td>
<td>1G</td>
<td>No</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>np6_0</td>
<td>0</td>
<td>port37</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port38</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port39</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port40</td>
<td>10G</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>np6_2</td>
<td>0</td>
<td>port43</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port44</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port41</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port42</td>
<td>10G</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>np6_3</td>
<td>0</td>
<td>port33</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port34</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port35</td>
<td>10G</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port36</td>
<td>10G</td>
</tr>
</tbody>
</table>

FortiGate-3000D fast path architecture

The FortiGate-3000D features 16 front panel SFP+ 10Gb interfaces connected to two NP6 processors through an Integrated Switch Fabric (ISF). The FortiGate-3000D has the following fastpath architecture:

- 8 SFP+ 10Gb interfaces, port1 through port8 share connections to the first NP6 processor (np6_0).
- 8 SFP+ 10Gb interfaces, port9 through port16 share connections to the second NP6 processor (np6_1).
You can use the following get command to display the FortiGate-3000D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1 and the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip  XAUI Ports  Max  Cross-chip
     Speed     offloading
-------  -------  ------  -------------
np6_0     0  port1  10G  Yes
          0  port6  10G  Yes
          1  port2  10G  Yes
          1  port5  10G  Yes
          2  port3  10G  Yes
          2  port8  10G  Yes
          3  port4  10G  Yes
          3  port7  10G  Yes

np6_1     0  port10 10G  Yes
          0  port13 10G  Yes
          1  port9  10G  Yes
          1  port14 10G  Yes
          2  port12 10G  Yes
```
FortiGate-3100D fast path architecture

The FortiGate-3100D features 32 SFP+ 10Gb interfaces connected to two NP6 processors through an Integrated Switch Fabric (ISF). The FortiGate-3100D has the following fastpath architecture:

- 16 SFP+ 10Gb interfaces, port1 through port16 share connections to the first NP6 processor (np6_0).
- 16 SFP+ 10Gb interfaces, port27 through port32 share connections to the second NP6 processor (np6_1).

You can use the following get command to display the FortiGate-3100D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1 and the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
```

Chip XAUI Ports Max Cross-chip
## FortiGate-3200D Fast Path Architecture

The FortiGate-3200D features two NP6 processors connected to an Integrated Switch Fabric (ISF). The FortiGate-3200D has the following fastpath architecture:

- 24 SFP+ 10Gb interfaces, port1 through port24 share connections to the first NP6 processor (np6_0).
- 24 SFP+ 10Gb interfaces, port25 through port48 share connections to the second NP6 processor (np6_1).

### Speed Offloading

<table>
<thead>
<tr>
<th>np6_0</th>
<th></th>
<th>port1</th>
<th>10G</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>port6</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port10</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port13</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port2</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port5</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port9</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port14</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port3</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port8</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port12</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port15</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port4</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port7</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port11</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port16</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>np6_1</th>
<th></th>
<th>port17</th>
<th>10G</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>port21</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port25</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port29</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port18</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port22</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port30</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port19</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port23</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port27</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port31</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port20</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port24</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port28</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>port32</td>
<td>10G</td>
<td>Yes</td>
</tr>
</tbody>
</table>
You can use the following get command to display the FortiGate-3200D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1 and the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
Chip    XAUI Ports  Max   Cross-chip
        Speed   offloading
-------    -------   ------   ----------
np6_0     0     port1  10G   Yes
          0     port5  10G   Yes
          0     port10 10G   Yes
          0    port13  10G   Yes
          0    port17  10G   Yes
          0    port22  10G   Yes
          1    port2   10G   Yes
          1    port6   10G   Yes
          1    port9   10G   Yes
          1    port14  10G   Yes
          1    port18  10G   Yes
          1    port21  10G   Yes
          2    port3   10G   Yes
          2    port7   10G   Yes
```
### FortiGate-3400E and 3401E fast path architecture

The FortiGate-3400E and 3401E models feature 24 front panel 10GigE SFP+ interfaces (HA1, HA2, port 1 to 22) and four 100GigE QSFP28 interfaces (port 23 to 26) connected to six NP6 processors (NP0 to NP5) through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fast path offloading or aggregate interfaces.

<table>
<thead>
<tr>
<th>Port</th>
<th>Type</th>
<th>Description</th>
<th></th>
<th>---</th>
<th>---</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_1</td>
<td>0</td>
<td>port26</td>
<td>10G</td>
<td>Yes</td>
<td>0</td>
<td>port29</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port33</td>
<td>10G</td>
<td>Yes</td>
<td>0</td>
<td>port37</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port41</td>
<td>10G</td>
<td>Yes</td>
<td>0</td>
<td>port45</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port25</td>
<td>10G</td>
<td>Yes</td>
<td>1</td>
<td>port30</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port34</td>
<td>10G</td>
<td>Yes</td>
<td>1</td>
<td>port38</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port42</td>
<td>10G</td>
<td>Yes</td>
<td>1</td>
<td>port46</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port28</td>
<td>10G</td>
<td>Yes</td>
<td>2</td>
<td>port31</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port35</td>
<td>10G</td>
<td>Yes</td>
<td>2</td>
<td>port39</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port43</td>
<td>10G</td>
<td>Yes</td>
<td>2</td>
<td>port47</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port27</td>
<td>10G</td>
<td>Yes</td>
<td>3</td>
<td>port32</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port36</td>
<td>10G</td>
<td>Yes</td>
<td>3</td>
<td>port40</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port44</td>
<td>10G</td>
<td>Yes</td>
<td>3</td>
<td>port48</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Port</th>
<th>Type</th>
<th>Description</th>
<th></th>
<th>---</th>
<th>---</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---
You can use the following `get` command to display the FortiGate-3400E or 3401E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAU1</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port1</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port2</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port3</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port4</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port5</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port6</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port7</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port8</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port9</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port10</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port11</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port12</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port13</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port14</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port15</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port16</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port17</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port18</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port19</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port20</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port21</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port22</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-5</td>
<td>0-3</td>
<td>port23</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
</tbody>
</table>
FortiGate-3600E and 3601E fast path architecture

The FortiGate-3600E and 3601E models feature 32 front panel 10GigE SFP+ interfaces (HA1, HA2, port 1 to 30) and six 100GigE QSFP28 interfaces (port 31 to 36) connected to six NP6 processors (NP0 to NP5) through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fast path offloading or aggregate interfaces.

You can use the following get command to display the FortiGate-3600E or 3601E NP6 configuration. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip       XAUI Ports  Max Speed  Cross-chip offloading
-------------------  ------  --------  -------------------
NP#0-5      0-3 port1  100000M  Yes
NP#0-5      0-3 port2  100000M  Yes
NP#0-5      0-3 port3  100000M  Yes
NP#0-5      0-3 port4  100000M  Yes
NP#0-5      0-3 port5  100000M  Yes
NP#0-5      0-3 port6  100000M  Yes
```
FortiGate-3700D fast path architecture

The FortiGate-3700D features four NP6 processors. The first two NP6 processors (np6_0 and np6_1) can be configured for low latency operation. The low latency configuration changes the FortiGate-3700D fast path architecture.

FortiGate-3700D low latency fast path architecture

Ports 25 to 32 can be used for low latency offloading. As long as traffic enters and exits the FortiGate-3700D through ports connected to the same NP6 processor and using these low latency ports the traffic will be offloaded and have lower latency that other NP6 offloaded traffic. Latency is reduced by bypassing the integrated switch fabric (ISF).

You can use the following command to turn on low latency mode for np6_0 and np6_1:

```
config system np6
   edit np6_0
      set low-latency-mode enable
   next
   edit np6_1
      set low-latency-mode enable
```
Although you do not have to turn on low latency to both np6_0 and np6_1, if you turn on low latency for just one NP6, the other NP6 will still be mapped according to the normal latency configuration.

With low latency enabled for both np6_0 and np6_1, the FortiGate-3700D has the following fastpath architecture:

- Four SFP+ 10Gb interfaces, port25 to port28, share connections to the first NP6 processor (np6_0) so sessions entering one of these ports and exiting through another will experience low latency.
- Four SFP+ 10Gb interfaces, port29 to port32, share connections to the second NP6 processor (np6_1) so sessions entering one of these ports and exiting through another will experience low latency.
- Ten SFP+ 10Gb interfaces, port5 to port14, and two 40Gb QSFP interfaces, port1 and port2, share connections to the third NP6 processor (np6_2).
- Ten SFP+ 10Gb interfaces, port15 to port24, and two 40Gb QSFP interfaces, port3 and port4, share connections to the fourth NP6 processor (np6_3).

You can use the following get command to display the FortiGate-3700D NP6 configuration. In this output example, the first two NP6s (np6_0 and np6_1) are configured for low latency. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.
You can use the following command to turn off low latency mode for np6_0 and np6_1:

```bash
config system np6
edit np6_0
    set low-latency-mode disable
next
edit np6_1
    set low-latency-mode disable
end
```
You do not have to turn off low latency to both np6_0 and np6_1. If you turn off low latency to just one NP6, the other NP6 will still be mapped according to the normal configuration.

In addition to turning off low latency, entering these commands also changes how ports are mapped to NP6s. Port1 is now mapped to np6_0 and port 3 is not mapped to np6_1. The FortiGate-3700D has the following fastpath architecture:

- One 40Gb QSFP interface, port1, and four SFP+ 10Gb interfaces, port25 to port28 share connections to the first NP6 processor (np6_0).
- One 40Gb QSFP interface, port3, and four SFP+ 10Gb interfaces, port29 to port32 share connections to the second NP6 processor (np6_1).
- One 40Gb QSFP interface, port2 and ten SFP+ 10Gb interfaces, port5 to port14 share connections to the third NP6 processor (np6_2).
- One 40Gb QSFP interface, port4, and ten SFP+ 10Gb interfaces, port15 to port24 share connections to the fourth NP6 processor (np6_3).
You can use the following get command to display the FortiGate-3700D NP6 configuration with low latency turned off for np6_0 and np6_1. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```plaintext
get hardware npu np6 port-list
Chip  XAUI Ports Max Speed Cross-chip offloading
------- ----- ------- ---- -------
np6_0 0 port26 10G Yes
1 port25 10G Yes
2 port28 10G Yes
3 port27 10G Yes
0-3 port1 40G Yes
------- ----- ------- ---- -------
np6_1 0 port30 10G Yes
1 port29 10G Yes
2 port32 10G Yes
3 port31 10G Yes
0-3 port3 40G Yes
------- ----- ------- ---- -------
np6_2 0 port5 10G Yes
0 port9 10G Yes
0 port13 10G Yes
1 port6 10G Yes
1 port10 10G Yes
1 port14 10G Yes
2 port7 10G Yes
2 port11 10G Yes
3 port8 10G Yes
3 port12 10G Yes
0-3 port2 40G Yes
------- ----- ------- ---- -------
np6_3 0 port15 10G Yes
0 port19 10G Yes
0 port23 10G Yes
1 port16 10G Yes
1 port20 10G Yes
1 port24 10G Yes
2 port17 10G Yes
2 port21 10G Yes
3 port18 10G Yes
3 port22 10G Yes
0-3 port4 40G Yes
------- ----- ------- ---- -------
```

**FortiGate-3700DX fast path architecture**

The FortiGate-3700DX features four NP6 processors. The first two NP6 processors (np6_0 and np6_1) can be configured for low latency operation. The low latency configuration changes the FortiGate-3700D fast path architecture. The FortiGate-3700DX also includes two TP2 cards that offload GTPu sessions.
FortiGate-3700DX low latency fast path architecture

Ports 25 to 32 can be used for low latency offloading. As long as traffic enters and exits the FortiGate-3700D through ports connected to the same NP6 processor and using these low latency ports the traffic will be offloaded and have lower latency that other NP6 offloaded traffic. Latency is reduced by bypassing the integrated switch fabric (ISF).

You can use the following command to turn on low latency mode for np6_0 and np6_1:

```plaintext
cfg system np6
   edit np6_0
      set low-latency-mode enable
   next
   edit np6_1
      set low-latency-mode enable
end
```

You do not have to turn on low latency to both np6_0 and np6_1. If you turn on low latency for just one NP6, the other NP6 will still be mapped according to the normal latency configuration.

With low latency enabled for both np6_0 and np6_1 the FortiGate-3700D has the following fastpath architecture:

- Four SFP+ 10Gb interfaces, port25 to port28, share connections to the first NP6 processor (np6_0) so sessions entering one of these ports and exiting through another will experience low latency
- Four SFP+ 10Gb interfaces, port29 to port32, share connections to the second NP6 processor (np6_1) so sessions entering one of these ports and exiting through another will experience low latency
- Ten SFP+ 10Gb interfaces, port5 to port14, and two 40Gb QSFP interfaces, port1 and port2, share connections to the third NP6 processor (np6_2).
- Ten SFP+ 10Gb interfaces, port15 to port24, and two 40Gb QSFP interfaces, port3 and port4, share connections to the fourth NP6 processor (np6_3).
You can use the following get command to display the FortiGate-3700D NP6 configuration. In this output example, the first two NP6s (np6_0 and np6_1) are configured for low latency. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip   XAUI Ports Max Cross-chip Speed offloading
-------- ------- -------- ------ ----- ---------------
np6_2   0      port5  10G  Yes
         0      port9  10G  Yes
         0      port13 10G  Yes
         1      port6  10G  Yes
         1      port10 10G  Yes
         1      port14 10G  Yes
         2      port7  10G  Yes
         2      port11 10G  Yes
```
FortiGate-3700D normal latency fast path architecture

You can use the following command to turn off low latency mode for np6_0 and np6_1:

```plaintext
config system np6
    edit np6_0
        set low-latency-mode disable
    next
    edit np6_1
        set low-latency-mode disable
end
```

You do not have to turn off low latency to both np6_0 and np6_1. If you turn off low latency to just one NP6, the other NP6 will still be mapped according to the normal configuration.

In addition to turning off low latency, entering these commands also changes how ports are mapped to NP6s. Port1 is now mapped to np6_0 and port 3 is not mapped to np6_1. The FortiGate-3700D has the following fastpath architecture:

- One 40Gb QSFP interface, port1, and four SFP+ 10Gb interfaces, port25 to port28 share connections to the first NP6 processor (np6_0).
- One 40Gb QSFP interface, port3, and four SFP+ 10Gb interfaces, port29 to port32 share connections to the second NP6 processor (np6_1).
- One 40Gb QSFP interface, port2 and ten SFP+ 10Gb interfaces, port5 to port14 share connections to the third NP6 processor (np6_2).
- One 40Gb QSFP interface, port4, and ten SFP+ 10Gb interfaces, port15 to port24 share connections to the fourth NP6 processor (np6_3).

You can use the following get command to display the FortiGate-3700D NP6 configuration with low latency turned off for np6_0 and np6_1. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
Chip   XAUI Ports   Max Speed Cross-chip Speed offloading
------- ----- ------ ------ ------- ------
np6_0  0 port26     10G   Yes
       1 port25     10G   Yes
       2 port28     10G   Yes
       3 port27     10G   Yes
       0-3 port1    40G   Yes
np6_1  0 port30     10G   Yes
       1 port29     10G   Yes
       2 port32     10G   Yes
       3 port31     10G   Yes
```
### FortiGate-3800D fast path architecture

The FortiGate-3800D features four front panel 100GigE CFP2 interfaces, four 40GigE QSFP+ interfaces, and eight 10GigE SFP+ interfaces connected to eight NP6 processors through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fastpath offloading or aggregate interfaces.
You can use the following get command to display the FortiGate-3800D NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port. You can also use the diagnose npu np6 port-list command to display this information.

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP#0-7</td>
<td>0-3 port1</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port2</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port3</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port4</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port5</td>
<td>4000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port6</td>
<td>4000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port7</td>
<td>4000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port8</td>
<td>4000000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port9</td>
<td>1000000M</td>
<td>Yes</td>
</tr>
</tbody>
</table>
FortiGate NP6 architectures

```
NP#0-7   0-3  port10  10000M  Yes
NP#0-7   0-3  port11  10000M  Yes
NP#0-7   0-3  port12  10000M  Yes
NP#0-7   0-3  port13  10000M  Yes
NP#0-7   0-3  port14  10000M  Yes
NP#0-7   0-3  port15  10000M  Yes
NP#0-7   0-3  port16  10000M  Yes
-----     ----  ------  ------  ------
```

**FortiGate-3810D fast path architecture**

The FortiGate-3810D features six front panel 100GigE CFP2 interfaces connected to eight NP6 processors through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fastpath offloading or aggregate interfaces.
You can use the following get command to display the FortiGate-3810D NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip Offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>0-3 port1</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 port2</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 port3</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 port4</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 port5</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 port6</td>
<td>100000M</td>
<td>Yes</td>
</tr>
</tbody>
</table>
```
FortiGate-3815D fast path architecture

The FortiGate-3815D features four front panel 100GigE CFP2 interfaces and eight 10GigE SFP+ interfaces connected to eight NP6 processors through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fastpath offloading or aggregate interfaces.

You can use the following get command to display the FortiGate-3815D NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port. You can also use the `diagnose npu np6 port-list` command to display this information.

```
get hardware npu np6 port-list
Chip  XAUI Ports  Max  Cross-chip
```
FortiGate-3960E fast path architecture

The FortiGate-3960E features sixteen front panel 10GigE SFP+ interfaces and six 100GigE QSFP+ interfaces connected to sixteen NP6 processors through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fastpath offloading or aggregate interfaces.

<table>
<thead>
<tr>
<th>Speed</th>
<th>offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>all 0-3 port1</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port2</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port3</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port4</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port11</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port12</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port13</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port14</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port10</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port9</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port8</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port7</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port5</td>
<td>100000M Yes</td>
</tr>
<tr>
<td>all 0-3 port6</td>
<td>100000M Yes</td>
</tr>
</tbody>
</table>
You can use the following get command to display the FortiGate-3960E NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port. You can also use the diagnose npu np6 port-list command to display this information.

get hardware npu np6 port-list

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip Speed</th>
<th>Offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP#0-7</td>
<td>0-3 port1</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>NP#2</td>
<td>0-3 port2</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port3</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>NP#0-7</td>
<td>0-3 port4</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
### FortiGate-3980E fast path architecture

The FortiGate-3980E features sixteen front panel 10GigE SFP+ interfaces and ten 100GigE QSFP28 interfaces connected to twenty-eight NP6 processors through an Integrated Switch Fabric (ISF). Individual interfaces are not mapped to NP6 processors because of the integrated switch fabric. No special mapping is required for fastpath offloading or aggregate interfaces.
You can use the following get command to display the FortiGate-3980E NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port. You can also use the `diagnose npu np6 port-list` command to display this information.

```
diagnose npu np6 port-list
Chip  XAUI Ports  Max  Cross-chip
      Speed  offloading
---------  -------  -------  -----------
NP#0-7    0-3  port1  10000M  Yes
NP#0-7    0-3  port2  10000M  Yes
NP#0-7    0-3  port3  10000M  Yes
NP#0-7    0-3  port4  10000M  Yes
```
FortiGate NP6 architectures

<table>
<thead>
<tr>
<th>NP#</th>
<th>0-3 port</th>
<th>10000M</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP#0-7</td>
<td>port5</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port6</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port7</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port8</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port9</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port10</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port11</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port12</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port13</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port14</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port15</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port16</td>
<td>10000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port17</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#0-7</td>
<td>port18</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port19</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port20</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port21</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port22</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port23</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port24</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port25</td>
<td>100000M</td>
<td>Yes</td>
</tr>
<tr>
<td>NP#8-27</td>
<td>port26</td>
<td>100000M</td>
<td>Yes</td>
</tr>
</tbody>
</table>

For information about optimizing FortiGate-3980E IPsec VPN performance, see Optimizing FortiGate-3960E and 3980E IPsec VPN performance on page 49.

For information about supporting large traffic streams, see FortiGate-3960E and 3980E support for high throughput traffic streams on page 50.

**FortiGate-5001D fast path architecture**

The FortiGate5001D features two NP6 processors.

- port1, port3, fabric1 and base1 share connections to the first NP6 processor.
- port2, port4, fabric2 and base2 share connections to the second NP6 processor.
NP6 default interface mapping

You can use the following get command to display the FortiGate-5001D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```
get hardware npu np6 port-list
```

```
Chip XAUI Ports Max Speed Cross-chip offloading
------- ---- -------- ------ ---------------
np6_0 0 port3    10G     Yes
1
2 base1     1G     Yes
3
0-3 port1  40G     Yes
0-3 fabric1 40G    Yes
0-3 fabric3 40G    Yes
0-3 fabric5 40G    Yes
------- ---- -------- ------ ---------------
np6_1 0 port4    10G     Yes
1
2 base2     1G     Yes
3
0-3 port2  40G     Yes
0-3 fabric2 40G    Yes
```
NP6 interface mapping with split ports

If you use the following CLI command to split port1:

```
config system global
    set split-port port1
end
```

The new split ports (port1/1 to port1/4) are mapped to the same NP6 as the port1 interface:

```
diagnose npu np6 port-list
```

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI</th>
<th>Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>np6_0</td>
<td>0</td>
<td>port3</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>port1/1</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>port1/2</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>base1</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port1/3</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>port1/4</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0-3</td>
<td>fabric1</td>
<td>40G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0-3</td>
<td>fabric3</td>
<td>40G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0-3</td>
<td>fabric5</td>
<td>40G</td>
<td>Yes</td>
</tr>
<tr>
<td>np6_1</td>
<td>0</td>
<td>port4</td>
<td>10G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>base2</td>
<td>1G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>port2</td>
<td>40G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0-3</td>
<td>fabric2</td>
<td>40G</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>0-3</td>
<td>fabric4</td>
<td>40G</td>
<td>Yes</td>
</tr>
</tbody>
</table>

FortiGate-5001E and 5001E1 fast path architecture

The FortiGate-5001E and 5001E1 features two NP6 processors and an integrated switch fabric. The integrated switch fabric allows you to configure aggregate interfaces between interfaces connected to different NP6s and supports offloading between for traffic entering and exiting from any interfaces.

The NP6s are connected to network interfaces as follows:

- NP6_0 is connected to port1, port3, fabric1, and base1.
- NP6_1 is connected to port2, port4, fabric2, and base2.

The following diagram also shows the XAUI port connections between the NP6 processors and the front panel interfaces and the integrated switch fabric.
NP6 default interface mapping

You can use the following get command to display the FortiGate-5001E NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6. You can also use the diagnose npu np6 port-list command to display this information.

```plaintext
get hardware npu np6 port-list
Chip   XAUI Ports  Max Speed  Cross-chip Speed offloading
------- ------ ------- -------  ---------------
np6_0  0     port3  10G    Yes
       1
       2    base1  1G    Yes
       3
np6_1  0     port4  10G    Yes
       1
       2
```
NP6 interface mapping with split ports

If you use the following CLI command to split port1:

```
config system global
  set split-port port1
end
```

The new split ports (port1/1 to port 1/4) are mapped to the same NP6 as the port1 interface:

```
diagnose npu np6 port-list
Chip   XAUI Ports    Max  Cross-chip Speed  offloading
       ------ ------- ------ -------- -------- --------
np6_0   0     port3    10G  Yes  0
         0   port1/1    10G  Yes  0
         1   port1/2    10G  Yes  0
         2     base1     1G  Yes  0
         2   port1/3    10G  Yes  0
         3   port1/4    10G  Yes  0
np6_1   0     port4    10G  Yes  0
         1
         2
         3     base2     1G  Yes  0
         0     port2     40G  Yes  0
         0     port2     40G  Yes  0
         0     fabric2   40G  Yes  0
         0     fabric4   40G  Yes  0
```

FortiGate-6000 series

The FortiGate-6000 series includes the FortiGate-6300F, 6301F, 6500F, and 6501F. All of these models have the same hardware architecture. FortiGate-6000 models have separate data and management planes. The data plane handles all traffic and security processing functionality. The management plane handles management functions such as administrator logins, configuration and session synchronization, SNMP and other monitoring, HA heartbeat communication, and remote and (if supported) local disk logging. Separating these two planes means that resources used for traffic and security processing are not compromised by management activities.
In the data plane, two DP3 load balancers use session-aware load balancing to distribute sessions from the front panel interfaces (port1 to 28) to Fortinet Processor Cards (FPCs). The DP3 processors communicate with the FPCs across the 3.2Tbps integrated switch fabric. Each FPC processes sessions load balanced to it. The FPCs send outgoing sessions back to the integrated switch fabric and then out the network interfaces to their destinations.

The NP6 processor in each FPC enhances network performance with fastpath acceleration that offloads communication sessions from the FPC CPU. The NP6 processor can also handle some CPU intensive tasks, like IPsec VPN encryption/decryption. The NP6 processor in each FPC connects to the integrated switch fabric over four XAUI ports.

The CP9 processors in each FPC accelerate many common resource intensive security related processes such as SSL VPN, Antivirus, Application Control, and IPS.

The management plane includes the management board, base backplane, management interfaces, and HA heartbeat interfaces. Configuration and session synchronization between FPCs in a FortiGate-6000F occurs over the base backplane. In an HA configuration, configuration and session synchronization between the FortiGate-6000s in the
cluster takes place over the HA1 and HA2 interfaces. Administrator logins, SNMP monitoring, remote logging to one or more FortiAnalyzers or syslog servers, and other management functions use the MGMT1, MGMT2, and MGMT3 interfaces. You can use the 10Gbps MGMT3 interface for additional bandwidth that might be useful for high bandwidth activities such as remote logging.

From the management board, you can use the `diagnose npu np6 port-list` command to display the FortiGate-6000 NP6 configuration. The command output shows the NP6 configuration for all of the FPCs. You can see the same information for individual FPCs by logging into each FPC (for example by using the `execute system console-server connect <slot-number>` command) and using the same `diagnose` command or the `get hardware npu np6 port-list` command.

As shown in the example below for the FPC in slot 1, all of the FortiGate-6000 front panel interfaces and the fabric backplane (elbc-ctrl) connect to the NP6 processor in each FPC.

```
diagnose npu np6 port-list
==========================================================================
Slot: 1  Module SN: FCC6KFT018900347
Chip  XAUI Ports  Max  Cross-chip
       Speed  offloading
        ----  -------  ------  --------
   all  0-3  elbc-ctrl/110G  Yes
   all  0-3  port1  25G  Yes
   all  0-3  port2  25G  Yes
   all  0-3  port3  25G  Yes
   all  0-3  port4  25G  Yes
   all  0-3  port5  25G  Yes
   all  0-3  port6  25G  Yes
   all  0-3  port7  25G  Yes
   all  0-3  port8  25G  Yes
   all  0-3  port9  25G  Yes
   all  0-3  port10  25G  Yes
   all  0-3  port11  25G  Yes
   all  0-3  port12  25G  Yes
   all  0-3  port13  25G  Yes
   all  0-3  port14  25G  Yes
   all  0-3  port15  25G  Yes
   all  0-3  port16  25G  Yes
   all  0-3  port17  25G  Yes
   all  0-3  port18  25G  Yes
   all  0-3  port19  25G  Yes
   all  0-3  port20  25G  Yes
   all  0-3  port21  25G  Yes
   all  0-3  port22  25G  Yes
   all  0-3  port23  25G  Yes
   all  0-3  port24  25G  Yes
   all  0-3  port25 100G  Yes
   all  0-3  port26 100G  Yes
   all  0-3  port27 100G  Yes
   all  0-3  port28 100G  Yes
```

FortiController-5902D fast path architecture

The FortiController-5902D NP6 network processors and integrated switch fabric (ISF) provide hardware acceleration by offloading load balancing from the primary FortiController-5902D CPU. Network processors are especially useful for accelerating load balancing of TCP and UDP sessions.

The first packet of every new session is received by the primary FortiController-5902D and the primary FortiController-5902D uses its load balancing schedule to select the worker that will process the new session. This information is passed back to an NP6 network processor and all subsequent packets of the same sessions are offloaded to an NP6 network processor which sends the packet directly to a subordinate unit. Load balancing is effectively offloaded from the primary unit to the NP6 network processors resulting in a faster and more stable active-active cluster.

Traffic accepted by the FortiController-5902D F1 to F4 interfaces is that is processed by the primary FortiController-5902D and is also be offloaded to the NP6 processors.

Individual FortiController-5902D interfaces are not mapped to NP6 processors. Instead an Aggregator connects the all fabric interfaces to the ISF and no special mapping is required for fastpath offloading.

NP6 content clustering mode interface mapping

FortiController-5902Ds run in content clustering mode and load balance sessions to FortiGate-5001D workers. Use the following command to enable content clustering:
config system elbc
    set mode content-cluster
    set inter-chassis-support enable
end

You can use the following get command to display the content clustering FortiController-5902D NP6 configuration. The output shows that all ports are mapped to all NP6 processors. You can also use the `diagnose npu np6 port-list` command to display this information.

get hardware npu np6 port-list

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>all 0-3 f1</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f2</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f3</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f4</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 np6_0_4</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 np6_0_5</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/1-2</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/3</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/4</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/5</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/6</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/7</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/8</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/9</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/10</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/11</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/12</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/13</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 elbc-ctrl/14</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

**NP6 default interface mapping**

You can use the following command to display the default FortiController-5902D NP6 configuration.

`diagnose npu np6 port-list`

<table>
<thead>
<tr>
<th>Chip</th>
<th>XAUI Ports</th>
<th>Max Speed</th>
<th>Cross-chip offloading</th>
</tr>
</thead>
<tbody>
<tr>
<td>all 0-3 f1</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f2</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f3</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 f4</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 np6_0_4</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 np6_0_5</td>
<td>10000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric1/2</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric3</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric4</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric5</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric6</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric7</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>all 0-3 fabric8</td>
<td>40000M</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
FortiGate NP6 architectures

<table>
<thead>
<tr>
<th>all</th>
<th>0-3 fabric9</th>
<th>40000M</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>0-3 fabric10</td>
<td>40000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 fabric11</td>
<td>40000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 fabric12</td>
<td>40000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 fabric13</td>
<td>40000M</td>
<td>Yes</td>
</tr>
<tr>
<td>all</td>
<td>0-3 fabric14</td>
<td>40000M</td>
<td>Yes</td>
</tr>
</tbody>
</table>

FortiGate-7030E fast path architecture

The FortiGate-7030E chassis schematic below shows the communication channels between chassis components including the management module (MGMT), the FIM (called FIM1) and the FPMs (FPM3 and FPM4).

![Chassis Schematic](image)

The management module (MGMT, with Intelligent Platform Management Bus (IPMB) address 0x20) communicates with all modules in the chassis over the base backplane. Each module, including the management module includes a Shelf Management Controller (SMC). These SMCs support IPMB communication between the management module and the FIM and FPMs for storing and sharing sensor data that the management module uses to control chassis cooling and power distribution. The base backplane also supports serial communications to allow console access from the management module to all modules, and 1Gbps Ethernet communication for management and heartbeat communication between modules.

FIM1 (IPMB address 0x82) is the FIM in slot 1. The interfaces of this module connect the chassis to data networks and can be used for Ethernet management access to chassis components. The FIM includes DP2 processors that distribute sessions over the Integrated Switch Fabric (ISF) to the NP6 processors in the FPMs. Data sessions are communicated to the FPMs over the 80Gbps chassis fabric backplane.

FPM3 and FPM4 (IPMB addresses 0x86 and 0x88) are the FPM processor modules in slots 3 and 4. These worker modules process sessions distributed to them by the FIM. FPMs include NP6 processors to offload sessions from the FPM CPU and CP9 processors that accelerate content processing.
FortiGate-7040E fast path architecture

The FortiGate-7040E chassis schematic below shows the communication channels between chassis components including the management module (MGMT), the FIMs (called FIM1 and FIM2) and the FPMs (FPM3 and FPM4).

The management module (MGMT, with Intelligent Platform Management Bus (IPMB) address 0x20) communicates with all modules in the chassis over the base backplane. Each module, including the management module, includes a Shelf Management Controller (SMC). These SMCs support IPMB communication between the management module and the FIM and FPMs for storing and sharing sensor data that the management module uses to control chassis cooling and power distribution. The base backplane also supports serial communications to allow console access from the management module to all modules, and 1Gbps Ethernet communication for management and heartbeat communication between modules.

FIM1 and FIM2 (IPMB addresses 0x82 and 0x84) are the FIMs in slots 1 and 2. The interfaces of these modules connect the chassis to data networks and can be used for Ethernet management access to chassis components. The FIMs include DP2 processors that distribute sessions over the Integrated Switch Fabric (ISF) to the NP6 processors in the FPMs. Data sessions are communicated to the FPMs over the 80Gbps chassis fabric backplane.

FPM3 and FPM4 (IPMB addresses 0x86 and 0x88) are the FPM processor modules in slots 3 and 4. These worker modules process sessions distributed to them by the FIMs. FPMs include NP6 processors to offload sessions from the FPM CPU and CP9 processors that accelerate content processing.

FortiGate-7060E fast path architecture

The FortiGate-7060E chassis schematic below shows the communication channels between chassis components including the management modules (MGMT), the FIMs (called FIM1 and FIM2) and the FPMs (FPM3, FPM4, FPM5, and FPM6).
By default, MGMT2 is the active management module and MGMT1 is inactive. The active management module always has the Intelligent Platform Management Bus (IPMB) address 0x20 and the inactive management module always has the IPMB address 0x22.

The active management module communicates with all modules in the chassis over the base backplane. Each module, including the management modules has a Shelf Management Controller (SMC). These SMCs support IPMB communication between the active management module and the FIM and FPMs for storing and sharing sensor data that the management module uses to control chassis cooling and power distribution. The base backplane also supports serial communications to allow console access from the management module to all modules, and 1Gbps Ethernet communication for management and heartbeat communication between modules.

FIM1 and FIM2 (IPMB addresses 0x82 and 0x84) are the FIMs in slots 1 and 2. The interfaces of these modules connect the chassis to data networks and can be used for Ethernet management access to chassis components. The FIMs include DP2 processors that distribute sessions over the Integrated Switch Fabric (ISF) to the NP6 processors in the FPMs. Data sessions are communicated to the FPMs over the 80Gbps chassis fabric backplane.

FPM03, FPM04, FPM05, and FPM06 (IPMB addresses 0x86, 0x88, 0x8A, and 0x8C) are the FPM processor modules in slots 3 to 6. These worker modules process sessions distributed to them by the FIMs. FPMs include NP6 processors to offload sessions from the FPM CPU and CP9 processors that accelerate content processing.

**FIM-7901E fast path architecture**

The FIM-7901E includes an integrated switch fabric (ISF) that connects the FIM front panel interfaces to the DP2 session-aware load balancers and to the chassis fabric and base backplanes. The ISF also allows the DP2 processors to
distribute sessions among all NP6 processors on the FortiGate Processor Modules (FPM) in the FortiGate-7000 chassis.

**FIM-7901E schematic**

**FIM-7904E fast path architecture**

The FIM-7904E includes an integrated switch fabric (ISF) that connects the front panel interfaces to the DP2 session-aware load balancers and to the chassis backplanes. The ISF also allows the DP2 processors to distribute sessions among all NP6 processors on the FortiGate Processor Modules (FPM) in the FortiGate-7000 chassis.
FIM-7904E hardware architecture

FIM-7910E fast path architecture

The FIM-7910E includes an integrated switch fabric (ISF) that connects the front panel interfaces to the DP2 session-aware load balancers and to the chassis backplanes. The ISF also allows the DP2 processors to distribute sessions among all NP6 processors on the FortiGate Processor Modules (FPM) in the FortiGate-7000 chassis.
FIM-7920E fast path architecture

The FIM-7920E includes an integrated switch fabric (ISF) that connects the front panel interfaces to the DP2 session-aware load balancers and to the chassis backplanes. The ISF also allows the DP2 processors to distribute sessions among all NP6 processors on the FortiGate Processor Modules (FPM) in the FortiGate-7000 chassis.
In a FortiGate-7000 chassis, FPM-7620E NP6 network processors combined with the FortiGate Interface Module (FIM) Integrated Switch Fabric (ISF) provide hardware acceleration by offloading sessions from the FPM-7620E CPUs. The result is enhanced network performance provided by the NP6 processors plus the removal of network processing load from the FPM-7620 CPUs. The NP6 processors can also handle some CPU-intensive tasks, like IPsec VPN encryption/decryption. Because of the ISF in each FIM, all sessions are fast-pathed and accelerated.
FPM-7620E hardware architecture
FortiGate NP6lite architectures

This chapter shows the NP6lite architecture for all FortiGate models that include NP6lite processors.

FortiGate-200E and 201E fast path architecture

The FortiGate-200E and 201E include two NP6lite processors. Because of this model, you cannot create Link Aggregation Groups (LAGs) or redundant interfaces between interfaces connected to different NP6lites. As well traffic will only be offloaded if it enters and exits the FortiGate on interfaces connected to the same NP6lite.

The NP6lites are connected to network interfaces as follows:
- NP6lite_0 is connected to six 1GE RJ-45 interfaces (port9-port14) and four 1GE SFP interfaces (port15-18).
- NP6lite_1 is connected to ten 1GE RJ45 interfaces (wan1, wan2, port1-port8).

The following diagram also shows the RGMII and QSGMII port connections between the NP6lite processors and the front panel interfaces. Both RGMII and QSGMII interfaces operate at 1000Mbps. However, QSGMII interfaces can also negotiate to operate at lower speeds: 10, 100, and 1000Mbps. To connect the FortiGate-200E to networks with speeds lower than 1000Mbps use the QSGMII interfaces (port1-8 and port11-18).

You can use the following get command to display the FortiGate-200E or 201E NP6lite configuration. You can also use the diagnose npu np6lite port-list command to display this information.

```
get hardware npu np6lite port-list
Chip XAUI Ports Max Cross-chip Speed offloading
------- ---- ------- ----- ------------
np6lite_0 2 port9 1000M NO
1 port10 1000M NO
4 port11 1000M NO
```
<table>
<thead>
<tr>
<th>Device</th>
<th>Port</th>
<th>Speed</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FortiGate NP6lite architectures</td>
<td>port12</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port13</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port14</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port15</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port16</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port17</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port18</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>wan1</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>wan2</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port1</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port2</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port3</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port4</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port5</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port6</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port7</td>
<td>1000M</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>port8</td>
<td>1000M</td>
<td>NO</td>
</tr>
</tbody>
</table>
Hardware acceleration get and diagnose commands

This section describes some get and diagnose commands you can use to display useful information about the NP6 processors sessions processed by NP6 processors.

get hardware npu np6

You can use the get hardware npu np6 command to display information about the NP6 processors in your FortiGate and the sessions they are processing. This command contains a subset of the options available from the diagnose npu np6 command. The command syntax is:

```
get hardware npu np6 {dce <np6-id> | ipsec-stats | port-list | session-stats <np6-id> | sse-stats <np6-id> | synproxy-stats}
```

- `<np6-id>` identifies the NP6 processor. 0 is np6_0, 1 is np6_1 and so on.
- `dce` show NP6 non-zero sub-engine drop counters for the selected NP6.
- `ipsec-stats` show overall NP6 IPsec offloading statistics.
- `port-list` show the mapping between the FortiGate physical interfaces and NP6 processors.
- `session-stats` show NP6 session offloading statistics counters for the selected NP6.
- `sse-stats` show hardware session statistics counters.
- `synproxy-stats` show overall NP6 synproxy statistics for TCP connections identified as being syn proxy DoS attacks.

diagnose npu np6

The diagnose npu np6 command displays extensive information about NP6 processors and the sessions that they are processing. Some of the information displayed can be useful for understanding the NP6 configuration, seeing how sessions are being processed and diagnosing problems. Some of the commands may only be useful for Fortinet software developers. The command syntax is:

```
diagnose npu np6 {options}
```

The following options are available:

- `fastpath {disable | enable} <np6-od>` enable or disable fastpath processing for a selected NP6.
- `dce` shows NP6 non-zero sub-engine drop counters for the selected NP6.
- `dce-all` show all subengine drop counters.
- `anomaly-drop` show non-zero L3/L4 anomaly check drop counters.
- `anomaly-drop-all` show all L3/L4 anomaly check drop counters.
- `hrx-drop` show non-zero host interface drop counters.
hrx-drop-all show all host interface drop counters.
session-stats show session offloading statistics counters.
session-stats-clear clear session offloading statistics counters.
sse-stats show hardware session statistics counters.
sse-stats-clear show hardware session statistics counters.
pdq show packet buffer queue counters.
xgmac-stats show XGMAC MIBs counters.
xgmac-stats-clear clear XGMAC MIBS counters.
port-list show port list.
ipsec-stats show IPsec offloading statistics.
ipsec-stats-clear clear IPsec offloading statistics.
eeprom-read read NP6 EEPROM.
npu-feature show NPU feature and status.
register show NP6 registers.
fortilink configure managed FortiSwitch.
synproxy-stats show synproxy statistics.

Using diagnose npu np6 npu-feature to verify enabled NP6 features

You can use the `diagnose npu np6 npu-feature` command to see the NP6 features that are enabled on your FortiGate and those that are not.

The following command output, from a FortiGate-1500D, shows the default NP6 configuration for most FortiGates with NP6 processors:

```
diagnose npu np6 npu-feature
___________________________  ______  ______
np_0  np_1
-------------------------------------------
Fastpath       Enabled  Enabled
HPE-type-shaping Disabled  Disabled
Standalone     No       No
IPv4 firewall  Yes      Yes
IPv6 firewall  Yes      Yes
IPv4 IPSec     Yes      Yes
IPv6 IPSec     Yes      Yes
IPv4 tunnel    Yes      Yes
IPv6 tunnel    Yes      Yes
GRE tunnel     No       No
GRE passthrough Yes      Yes
IPv4 Multicast Yes      Yes
IPv6 Multicast Yes      Yes
CAPWAP         Yes      Yes
RDP Offload    Yes      Yes
```

FortiOS Handbook - Hardware Acceleration
If you use the following command to disable fastpath:

```plaintext
config system npu
  set fastpath disable
end
```

The `npu-feature` command output shows this configuration change:

```plaintext
diagnose npu np6 npu-feature

np_0        np_1
-------------  ----------
Fastpath      Disabled  Disabled
HPE-type-shaping Disabled  Disabled
Standalone    No        No
IPv4 firewall Yes       Yes
IPv6 firewall Yes       Yes
IPv4 IPSec    Yes       Yes
IPv6 IPSec    Yes       Yes
IPv4 tunnel   Yes       Yes
IPv6 tunnel   Yes       Yes
GRE tunnel    No        No
GRE passthrough Yes      Yes
IPv4 Multicast Yes      Yes
IPv6 Multicast Yes      Yes
CAPWAP        Yes       Yes
RDP Offload   Yes       Yes
```

### Using diagnose npu np6lite npu-feature to verify enabled NP6Lite features

You can use the `diagnose npu np6lite npu-feature` command to see the NP6Lite features that are enabled on your FortiGate and those that are not.

The following command output, from a FortiGate-200E, shows the default NP6Lite configuration for most FortiGates with NP6Lite processors:

```plaintext
diagnose npu np6 npu-feature

np_0        np_1
-------------  ----------
Fastpath      Enabled  Enabled
IPv4 firewall Yes       Yes
IPv6 firewall Yes       Yes
IPv4 IPSec    Yes       Yes
IPv6 IPSec    Yes       Yes
IPv4 tunnel   Yes       Yes
IPv6 tunnel   Yes       Yes
GRE tunnel    No        No
```

If you use the following command to disable fastpath:

```plaintext
config system npu
  set fastpath disable
end
```

The `npu-feature` command output show this configuration change:
Using the diagnose sys session/session6 list command

The `diagnose sys session list` and `diagnose sys session6 list` commands list all of the current IPv4 or IPv6 sessions being processed by the FortiGate. For each session the command output includes an `npu info` line that displays NPx offloading information for the session. If a session is not offloaded the command output includes a `no_ofld_reason` line that indicates why the session was not offloaded.

Displaying NP6 offloading information for a session

The `npu info` line of the `diagnose sys session list` command includes information about the offloaded session that indicates the type of processor and whether its IPsec or regular traffic:

- `offload=8/8` for NP6 sessions.
- `flag 0x81` means regular traffic.
- `flag 0x82` means IPsec traffic.

Example offloaded IPv4 NP6 session

The following session output by the `diagnose sys session list` command shows an offloaded session. The information in the `npu info` line shows this is a regular session (flag=0x81/0x81) that is offloaded by an NP6 processor (offload=8/8).

```plaintext
diagnose npu np6 npu-feature

--- np_0 np_1
Fastpath  Disabled Disabled
IPV4 firewall  Yes Yes
IPV6 firewall  Yes Yes
IPV4 IPSec  Yes Yes
IPV6 IPSec  Yes Yes
IPV4 tunnel  Yes Yes
IPV6 tunnel  Yes Yes
GRE tunnel  No No
```
Example IPv4 session that is not offloaded

The following session, output by the diagnose sys session list command includes the `no_ofld_reason` line that indicates that the session was not offloaded because it is a local-in session.

```
session info: proto=6 proto_state=01 duration=19 expire=3597 timeout=3600
flags=00000000 sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/ vlan_cos=8/8
state=local may_dirty
statistic(bytes/packets/allow_err): org=6338/15/1 reply=7129/12/1 tuples=2
speed(Bps/kbps): 680/5
origin->sink: org pre->in, reply out->post dev=15->50/50->15 gwy=5.5.5.5/0.0.0.0
hook=pre dir=org act=noop 5.5.5.33:60567->5.5.5.5:443(0.0.0.0:0)
hook=post dir=reply act=noop 5.5.5.33:60567->5.5.5.5:443(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=00002d29
tos=ff/ff
ips_view=0 app_list=0 app=0
url_cat=0
dd_type=0 dd_mode=0
no_ofld_reason: local
```

Example IPv4 IPsec NP6 session

```
diagnose sys session list
session info: proto=6 proto_state=01 duration=34 expire=3565 timeout=3600 flags=00000000 sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/pl-vdom2
state=re may_dirty npu
statistic(bytes/packets/allow_err): org=112/2/1 reply=112/2/1 tuples=2
origin->sink: org pre->post, reply pre->post dev=57->7/7->57 gwy=10.1.100.11/11.11.11.1
hook=pre dir=org act=noop 172.16.200.55:35254->10.1.100.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.11:80->172.16.200.55:35254(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 id_policy_id=0 auth_info=0 chk_client_info=0 vd=4
serial=000002d29
tos=ff/ff
ips_view=0 app_list=0 app=0
url_cat=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=172.16.200.55, bps=260
```
npu_state=00000000
npu info: flag=0x081/0x82, offload=8/8, ips_offload=0/0, epid=1/3, ipid=3/1, vlan=32779/0

Example IPv6 NP6 session

diagnose sys session6 list
session6 info: proto=6 proto_state=01 duration=2 expire=3597 timeout=3600 flags=00000000 sock-port=0 sockflag=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0
policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=152/2/0 reply=152/2/0 tuples=2
speed(Bps/kbps): 0/0
origin->sink: org pre->post, reply pre->post dev=13->14/14->13
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0 serial=0000027a
npu_state=01040000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=137/136, ipid=136/137, vlan=0/0

Example NAT46 NP6 session

diagnose sys session list
session info: proto=6 proto_state=01 duration=19 expire=3580 timeout=3600 flags=00000000 sock-port=0 sockflag=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=npu nlb
statistic(bytes/packets/allow_err): org=112/2/1 reply=112/2/1 tuples=2
speed(Bps/kbps): 0/0
origin->sink: org nataf->post, reply pre->org dev=52->14/14->52 gwy=0.0.0.0/10.1.100.1
hook=5 dir=org act=noop 10.1.100.1:21937->10.1.100.11:80(0.0.0.0:0)
hook=6 dir=reply act=noop 10.1.100.11:80->10.1.100.1:21937(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=04051aae tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
npu_state=00000000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=1/3, ipid=0/0, serial=04051aae

Example NAT64 NP6 session

diagnose sys session6 list
session6 info: proto=6 proto_state=01 duration=36 expire=3563 timeout=3600 flags=00000000 sock-port=0 sockflag=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0
policy_dir=0 tunnel=/
state=may_dirty npu nlb
statistic(bytes/packets/allow_err): org=72/1/0 reply=152/2/0 tuples=2
speed(Bps/kbps): 0/0
orgin->sink: org pre->org, reply nataf->post dev=13->14/14->13
hook=post dir=reply act=noop 64:ff9b::a01:640b:80 ->2000:172:16:200::55:33945 (:::0)
hook=5 dir=org act=noop 10.1.100.1:21937->10.1.100.11:80 (0.0.0.0:0)
hook=6 dir=reply act=noop 10.1.100.11:80->10.1.100.1:21937 (0.0.0.0:0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0 serial=0000027b
npu_state=00000000
npu info: flag=0x00/0x81, offload=8/0, ips_offload=0/0, epid=137/0, ipid=136/0, vlan=0/0

diagnose npu np6 session-stats <np6-id> (number of NP6 IPv4 and IPv6 sessions)

You can use the diagnose npu np6 portlist command to list the NP6-ids and the interfaces that each NP6 is connected to. The <np6-id> of np6_0 is 0, the <np6-id> of np6_1 is 1 and so on. The diagnose npu np6 session-stats <np6-id> command output includes the following headings:

- ins44 installed IPv4 sessions
- ins46 installed NAT46 sessions
- del4 deleted IPv4 and NAT46 sessions
- ins64 installed NAT64 sessions
- ins66 installed IPv6 sessions
- del6 deleted IPv6 and NAT64 sessions
- e is the error counter for each session type

<table>
<thead>
<tr>
<th>qid</th>
<th>ins44</th>
<th>ins46</th>
<th>del4</th>
<th>ins64</th>
<th>ins66</th>
<th>del6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ins44_e</td>
<td>ins46_e</td>
<td>del4_e</td>
<td>ins64_e</td>
<td>ins66_e</td>
<td>del6_e</td>
</tr>
<tr>
<td>0</td>
<td>94</td>
<td>0</td>
<td>44</td>
<td>0</td>
<td>40</td>
<td>30</td>
</tr>
<tr>
<td>1</td>
<td>84</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>30</td>
<td>28</td>
</tr>
<tr>
<td>2</td>
<td>90</td>
<td>0</td>
<td>42</td>
<td>0</td>
<td>40</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>86</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>4</td>
<td>72</td>
<td>0</td>
<td>34</td>
<td>0</td>
<td>34</td>
<td>28</td>
</tr>
<tr>
<td>5</td>
<td>86</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>28</td>
<td>32</td>
</tr>
<tr>
<td>6</td>
<td>82</td>
<td>0</td>
<td>38</td>
<td>0</td>
<td>32</td>
<td>34</td>
</tr>
<tr>
<td>7</td>
<td>86</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>30</td>
<td>30</td>
</tr>
</tbody>
</table>
diagnose npu np6 ipsec-stats (NP6 IPsec statistics)

The command output includes IPv4, IPv6, and NAT46 IPsec information:

- `spi_ses4` is the IPv4 counter
- `spi_ses6` is the IPv6 counter
- `4to6_ses` is the NAT46 counter

### IPsec Virtual interface stats:

<table>
<thead>
<tr>
<th>vif_start_oid</th>
<th>vif_end_oid</th>
<th>03ed</th>
<th>03fc</th>
</tr>
</thead>
<tbody>
<tr>
<td>vif_get</td>
<td>vif_get_expired</td>
<td>000000000000</td>
<td></td>
</tr>
<tr>
<td>vif_get_fail</td>
<td>vif_get_invld</td>
<td>000000000000</td>
<td></td>
</tr>
<tr>
<td>vif_set</td>
<td>vif_set_invld</td>
<td>000000000000</td>
<td></td>
</tr>
<tr>
<td>vif_clear</td>
<td>vif_clear_invld</td>
<td>000000000000</td>
<td></td>
</tr>
</tbody>
</table>

### np6_0:

- `sa_install`  
- `sa_remove`  
- `4to6_ses_ins`  
- `4to6_ses_del`  
- `spi_ses6_ins`  
- `spi_ses6_del`  
- `spi_ses4_ins`  
- `spi_ses4_del`  
- `sa_map_alloc_fail`  
- `sa_ins_null_adapter`  
- `del_sa_mismatch`  
- `ib_chk_null_adpt`  
- `ob_chk_null_adpt`  
- `rx_vif_miss`  
- `rx_sa_miss`  
- `waiting_ib_sa`  
- `msg_miss`  

### np6_1:

- `sa_install`  
- `sa_remove`  
- `4to6_ses_ins`  
- `4to6_ses_del`  
- `spi_ses6_ins`  
- `spi_ses6_del`  
- `spi_ses4_ins`  
- `spi_ses4_del`  
- `sa_map_alloc_fail`  
- `sa_ins_null_adapter`
Hardware acceleration get and diagnose commands

```plaintext
del_sa_mismatch 00000000000 ib_chk_null_adpt 00000000000
ib_chk_null_sa 00000000000 ob_chk_null_adpt 00000000000
ob_chk_null_sa 00000000000 rx_vif_miss 00000000000
rx_sa_mismatch 00000000000 rx_mark_miss 00000000000
waiting_ib_sa 00000000000 sa_mismatch 00000000000
msg_miss 00000000000
```

diagnose npu np6 sse-stats <np6-id> (number of NP6 sessions and dropped sessions)

This command displays the total number of inserted, deleted and purged sessions processed by a selected NP6 processor. The number of dropped sessions of each type can be determined by subtracting the number of successful sessions from the total number of sessions. For example, the total number of dropped insert sessions is `insert-total-insert-success`.

```plaintext
diagnose npu np6 sse-stats 0
Counters            SSE0           SSE1           Total
-------------------  ---------------  ---------------  ---------------
active              0               0               0
insert-total        25              0               0
insert-success      25              0               0
delete-total        25              0               0
delete-success      25              0               0
purge-total         0               0               0
purge-success       0               0               0
search-total        40956           38049           79005
search-hit          37714           29867           67581
-------------------  ---------------  ---------------  ---------------
pht-size            8421376         8421376
ofst-size           8355840         8355840
ofstfree            8355839         8355839
PBA                 3001            3001
```

diagnose npu np6 dce <np6-id> (number of dropped NP6 packets)

This command displays the number of dropped packets for the selected NP6 processor.

- **IHP1_PKTCHK** number of dropped IP packets
- **IPSEC0_ENGINB0** number of dropped IPSec
- **TPE_SHAPER** number of dropped traffic sharper packets

```plaintext
diag npu np6 dce 1
IHP1_PKTCHK :0000000000000000000000000001833 [5b] IPSEC0_ENGINB0 :0000000000000000000000000003 [80]
TPE_SHAPER :00000000000000000552 [94]
```
diagnose hardware deviceinfo nic <interface-name> (number of packets dropped by an interface)

This command displays a wide variety of statistics for FortiGate interfaces. The fields Host Rx dropped and Host Tx dropped display the number of received and transmitted packets that have been dropped.

diagnose hardware deviceinfo nic port2
...

= = = = = = = Counters = = = = = = = =
Rx Pkts : 20482043
Rx Bytes : 31047522516
Tx Pkts : 19000495
Tx Bytes : 1393316953
Host Rx Pkts : 27324
Host Rx Bytes : 1602755
Host Rx dropped : 0
Host Tx Pkts : 8741
Host Tx Bytes : 5731300
Host Tx dropped : 0
sw_rx_pkts : 20482043
sw_rx_bytes : 31047522516
sw_tx_pkts : 19000495
sw_tx_bytes : 1393316953
sw_np_rx_pkts : 19000495
sw_np_rx_bytes : 1469318933
sw_np_tx_pkts : 20482042
sw_np_tx_bytes : 31129450620

diagnose npu np6 synproxy-stats (NP6 SYN-proxied sessions and unacknowledged SYNs)

This command display information about NP6 syn-proxy sessions including the total number proxied sessions. As well the Number of attacks, no ACK from client shows the total number of acknowledged SYNs.

diagnose npu np6 synproxy-stats
DoS SYN-Proxy:
Number of proxied TCP connections : 39277346
Number of working proxied TCP connections : 182860
Number of retired TCP connections : 39094486
Number of attacks, no ACK from client : 208
Copyright © 2019 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.