|  |
| --- |
| ­  Verizon  ***Verizon <Project Name>***  ***Fortinet vNFV API Specifications***  Version X.Y  Issue date: MM/DD/YYYY |
| **Verizon and Ericsson Confidential and Proprietary** |

## Revision History:

|  |  |  |
| --- | --- | --- |
| **Revision** | **Date** | **Comment** |
|  |  |  |

Contents

[Revision History: 2](#_Toc446670112)

[1 Introduction 5](#_Toc446670113)

[1.1 Overview 5](#_Toc446670114)

[1.2 References 5](#_Toc446670115)

[1.2.1 Industry Standards Specifications 5](#_Toc446670116)

[1.2.2 Verizon Specifications 5](#_Toc446670117)

[2 Assumptions 6](#_Toc446670118)

[3 API Specifications 7](#_Toc446670119)

[3.1 API Name 7](#_Toc446670120)

[4 Attachments 7](#_Toc446670121)

Table of Figures

List of Tables

# Introduction

## Overview

This document contains the API Specifications for the FortiGate VNF used in a VNS deployment. This doc does not cover all API commands for the FortiGate, but rather a subset of commands that will be used repeatedly within VNS.

## References

### Industry Standards Specifications

1. <to be provided>

### Verizon Specifications

1. XXX.

# Assumptions

This section shall contain the list of relevant assumptions.

# API Specifications

FortiOS Rest APIs support the following HTTP methods:

|  |
| --- |
| **HTTP Method Description** |
| **GET** Retrieve a resource or collection of resources. |
| **POST** Create a resource or execute actions. |
| **PUT** Update a resource. |
| **DELETE** Delete a resource or collection of resources. |

FortiOS APIs use well-defined HTTP status codes to indicate query results to the API. Following are some of the HTTP status codes used

|  |
| --- |
| **HTTP Response Code Description** |
| 200- OK Request returns successful. |
| 400 - Bad Request Request cannot be processed by the API. |
| 401 - Not Authorized Request without successful login session. |
| 403 - Forbidden Request is missing CSRF token or administrator is missing access profile permissions. |
| 404 - Not Found Unable to find the specified resource. |
| 405 - Method Not Allowed Specified HTTP method is not allowed for this resource. |
| 413 - Request Entity Too Request cannot be processed due to large entity. Large |
| 424 - Failed Dependency Fail dependency can be duplicate resource, missing required parameter, missing required attribute, invalid attribute value. |
| 500 - Internal Server Error Internal error when processing the request. |

## FortiGate Login

1. This API is used to login to the FortiGate and is to be done before any other API calls can be made. ***The result of this API call provides the APSCOOKIE that needs to be passed to the FortiGate for every API call and csrftoken that is used for any WRITE requests to the FortiGate.***
   1. To setup an authenticated session
      1. **URL**: https://<FGT IP>/logincheck
      2. **Request Method**: POST
      3. **Input Form Data Fields**: username=<user>, secretkey=<password>
         1. Form data is x-www-form-urlencoded
   2. The URL and Request Method and Form Data Encoding are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url": "https://<FGT IP>/logincheck/",

"headers": "Content-Type: application/x-www-form-urlencoded\n",

"method": "POST",

"data": [

{

"key": "username",

"value": "admin",

"type": "text",

"enabled": true

},

{

"key": "secretkey",

"value": "admin",

"type": "text",

"enabled": true

}

],

"dataMode": "urlencoded"

**Request Response**

*Response Code*: 200 OK

*Response Headers:*

Connection: Keep-Alive

Content-Security-Policy: frame-ancestors 'self'

Content-Type: text/html; charset=utf-8

Date: Wed, 26 Oct 2016 17:59:25 GMT

Keep-Alive: timeout=10, max=20

Server: xxxxxxxx-xxxxx

Transfer-Encoding: chunked

X-Frame-Options: SAMEORIGIN

X-UA-Compatible: IE=Edge

*Response Body:*

<script language="javascript">

document.location="/ng/prompt?viewOnly&redir=%2Fng%2F";

</script>

*Response Cookies:*

Name: APSCOOKIE\_9538334086002207541

Value:"Era%3D1%26Payload%3DNBfONTCtQUhKHO7DndLD%2FeHbrr9snjxecLnodtQKCAKVvDi0dGZQDrJ9yKA%2F5dvg%0ADgW0gc6bCpPPLMTVnoLpke4nd781JvPjFuuCDDKscW33bhbqUSsUYdvIU28KNn8P%0AhZETFtGjUIsJZVNG2cdqoOTPmXtgbCpBny%2FCup0KF+k%3D%0A%26AuthHash%3DnnzCkJWxN6ENeOilgwhHS+H%2FLdwA%0A"

Domain: 192.168.186.130

Path:/

Expires:Never

HTTP:true

Secure:false

Name: ccsrftoken\_9538334086002207541

Value: 6337AFB78B3A93A46865F837FE320A3

Domain: 192.168.186.130

Path:/

Expires:Never

HTTP:false

Secure:false

Name: ccsrftoken

Value: 6337AFB78B3A93A46865F837FE320A3

Domain: 192.168.186.130

Path:/

Expires:Never

HTTP:false

Secure:false

1. For all input/output parameters:
   1. Format/syntax rules
   2. Required/Optional
   3. Valid values
   4. Default values if applicable
   5. Parameters dependencies
   6. Identification of if a value is sourced as an output from a other API
   7. Data Types with applicable restrictions for example alpha-numeric/ascii string with size and allowable characters etc.
   8. Default Values if applicable

## FortiGate Logout

1. This API is used to logout of the FortiGate and is to be done after all API calls are completed so no admin sessions are tied up.
   1. To logout of an authenticated session
      1. **URL**: https://<FGT IP>/logout
      2. **Request Method**: POST
      3. **Input Data**: none needed
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url": "https://<FGT IP>/logout",

"method": "POST",

**Request Response Example**

*Response Code*: 200 OK

*Response Headers:*

Connection: Keep-Alive

Content-Security-Policy: frame-ancestors 'self'

Content-Type: text/html; charset=utf-8

Date: Wed, 26 Oct 2016 18:20:22 GMT

Keep-Alive: timeout=10, max=20

Server: xxxxxxxx-xxxxx

Transfer-Encoding: chunked

X-Frame-Options: SAMEORIGIN

X-UA-Compatible: IE=Edge

*Response Body:*

<script language="javascript">

top.location="/login";

</script>

1. For all input/output parameters:
   1. Format/syntax rules
   2. Required/Optional
   3. Valid values
   4. Default values if applicable
   5. Parameters dependencies
   6. Identification of if a value is sourced as an output from a other API
   7. Data Types with applicable restrictions for example alpha-numeric/ascii string with size and allowable characters etc.
   8. Default Values if applicable

## FortiGate Resource Usage Check

1. This API is used to get the number of CPUs, CPU Usage, Average Memory Usage, Session information and Dish Usage from a FortiGate VM.
   1. To get the resource usage of an authenticated session
      1. **URL**: https://<FGT IP>/api/v2/monitor/system/resource
      2. **Request Method**: GET
      3. **Input Data**: none needed
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful

**Request Example**

"url":

“https://<FGT IP>/api/v2/monitor/system/resource",

"method": "GET",

“cookie”: “APSCOOKIE VALUE”

**Request Response Example**

*Response Code*: 200 OK

*Response Headers:*

Cache-Control: no-cache, must-revalidate

Connection: Keep-Alive

Content-Encoding: gzip

Content-Length: 400

Content-Security-Policy: frame-ancestors 'self'

Content-Type: application/json

Date: Wed, 26 Oct 2016 20:04:35 GMT

Keep-Alive: timeout=10, max=20

Server: xxxxxxxx-xxxxx

X-Frame-Options: SAMEORIGIN

X-UA-Compatible: IE=Edge

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists various parameters regarding the usage of the box. This API call is to be used to ensure that the number of CPUs the VM sees is the same as the one the openstack provisioning specified.

The resulting JSON message will consist of a table called results under which one should examing the field “id”: “cpu.<value>” to obtain the number of CPUs the VM is using. This field is highlighted below in the expected response and is marked in red. Other interesting values are highlighted but not marked in red.

*Response Body:*

{

"http\_method": "GET",

"results": [

{

"id": "cpu.1",

"name": "CPU 1",

"type": "cpu",

"value": 0,

"status": "normal",

"index": "1"

},

{

"id": "cpu.average",

"name": "Average CPU Usage",

"type": "cpu",

"value": 0,

"status": "normal"

},

{

"id": "mem.usage",

"name": "Memory Usage",

"type": "mem",

"value": 31,

"status": "normal"

},

{

"id": "disk.virtual-disk",

"name": "Virtual-Disk",

"type": "disk",

"value": 0,

"status": "normal",

"size": 30235,

"used": 175

},

{

"id": "disk.total",

"name": "Total Disk Usage",

"type": "disk",

"value": 0,

"status": "normal",

"size": 30235,

"used": 175

},

{

"id": "session.total",

"name": "Total Sessions",

"type": "session",

"value": 11,

"status": "normal"

},

{

"id": "session.new",

"name": "New Sessions per Second",

"type": "session",

"value": 1,

"status": "normal"

},

{

"id": "lograte.disk",

"name": "Disk",

"type": "lograte",

"value": 0,

"status": "normal"

}

],

"vdom": "root",

"path": "system",

"name": "resource",

"action": "select",

"status": "success",

"serial": "FGVM010000069826",

"version": "v5.4.1",

"build": 1064

}

1. For all input/output parameters:
   1. Format/syntax rules
   2. Required/Optional
   3. Valid values
   4. Default values if applicable
   5. Parameters dependencies
   6. Identification of if a value is sourced as an output from a other API
   7. Data Types with applicable restrictions for example alpha-numeric/ascii string with size and allowable characters etc.
   8. Default Values if applicable

## FortiGate List Interfaces

1. This API is used to get the interfaces configured on the FortiGate VM and can be used after the VM is provisioned to make sure the IP addresses are correct and the link is up on the VM
   1. To list the interfaces
      1. **URL**: https://<FGT IP>/api/v2/monitor/system/available-interfaces
      2. **Request Method**: GET
      3. **Input Data**: none needed
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url":

“https://<FGT IP>/api/v2/monitor/system/available-interfaces",

"method": "GET",

“cookie”: “APSCOOKIE VALUE”

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists the interfaces on the VM. This API call is to be used to ensure the management, IDN, Internet and PIP interfaces are all configured.

The expected message is detailed below. The VM should have a total of 5 interfaces with IPs configured and the link status should be “UP”.

The example below is based on a FGT VM that has a few interfaces with the link down which means they were not connected but status up which means that their administrative status was UP, and the internet interface is a VLAN this can be ignored.

There are 2 other interfaces that show up ssl.root and virtual-wan which can be ignored as well. The interesting fields and responses are highlighted and marked in red below

**Request Response Example**

REQUEST:

GET https://192.168.186.130/api/v2/monitor/system/available-interfaces

RESPONSE:

GET https://192.168.186.130/api/v2/monitor/system/available-interfaces

{

"vdom" : "root",

"name" : "available-interfaces",

"results" : [

{

"valid\_in\_policy" : true,

"name" : "any"

},

{

"type" : "physical",

"link" : "up",

"alias" : "Management",

"valid\_in\_policy" : true,

"is\_physical" : true,

"status" : "up",

"speed" : 1000,

"supports\_device\_id" : true,

"is\_ipsecable" : true,

"duplex" : "full",

"is\_explicit\_proxyable" : true,

"ipv4\_addresses" : [

{

"cidr\_netmask" : 24,

"ip" : "192.168.186.130",

"netmask" : "255.255.255.0"

}

],

"name" : "port1"

},

{

"name" : "port2",

"is\_explicit\_proxyable" : true,

"is\_ipsecable" : true,

"duplex" : "half",

"supports\_device\_id" : true,

"is\_physical" : true,

"status" : "up",

"valid\_in\_policy" : true,

"link" : "down",

"type" : "physical"

},

{

"vlan\_interface" : "port2",

"is\_ipsecable" : true,

"duplex" : "half",

"supports\_device\_id" : true,

"ipv4\_addresses" : [

{

"netmask" : "255.255.255.252",

"ip" : "192.168.100.1",

"cidr\_netmask" : 30

}

],

"is\_explicit\_proxyable" : true,

"name" : "port2.v100",

"vlan\_id" : 100,

"is\_vlan" : true,

"type" : "vlan",

"link" : "down",

"status" : "up",

"valid\_in\_policy" : true

},

{

"type" : "physical",

"link" : "down",

"is\_physical" : true,

"status" : "up",

"valid\_in\_policy" : true,

"supports\_device\_id" : true,

"is\_ipsecable" : true,

"duplex" : "half",

"is\_explicit\_proxyable" : true,

"name" : "port3"

},

{

"valid\_in\_policy" : true,

"status" : "up",

"link" : "down",

"type" : "vlan",

"name" : "port3.v100",

"vlan\_id" : 100,

"is\_vlan" : true,

"ipv4\_addresses" : [

{

"ip" : "192.168.100.5",

"cidr\_netmask" : 30,

"netmask" : "255.255.255.252"

}

],

"is\_explicit\_proxyable" : true,

"is\_ipsecable" : true,

"duplex" : "half",

"supports\_device\_id" : true,

"vlan\_interface" : "port3"

},

{

"is\_ipsecable" : true,

"duplex" : "half",

"supports\_device\_id" : true,

"ipv4\_addresses" : [

{

"netmask" : "255.255.255.0",

"cidr\_netmask" : 24,

"ip" : "172.16.100.100"

}

],

"is\_explicit\_proxyable" : true,

"name" : "port4",

"type" : "physical",

"link" : "down",

"alias" : "IDN",

"is\_physical" : true,

"valid\_in\_policy" : true,

"status" : "up"

},

{

"vlan\_interface" : "port3",

"supports\_device\_id" : true,

"is\_ipsecable" : true,

"duplex" : "half",

"is\_explicit\_proxyable" : true,

"ipv4\_addresses" : [

{

"cidr\_netmask" : 24,

"ip" : "1.2.3.4",

"netmask" : "255.255.255.0"

}

],

"is\_vlan" : true,

"vlan\_id" : 10,

"name" : "port5",

"type" : "vlan",

"link" : "down",

"alias" : "Internet",

"valid\_in\_policy" : true,

"status" : "up"

},

{

"status" : "up",

"name" : "ssl.root",

"is\_explicit\_proxyable" : true,

"alias" : "SSL VPN interface",

"is\_tunnel" : true,

"is\_sslvpn" : true,

"type" : "tunnel"

},

{

"link" : "down",

"type" : "virtual-wan",

"is\_virtual\_wan\_link" : true,

"status" : "down",

"name" : "virtual-wan-link",

"members" : []

}

],

"path" : "system",

"http\_method" : "GET",

"version" : "v5.4.1",

"status" : "success",

"serial" : "FGVM010000069826",

"action" : "select",

"build" : 1064

}

1. For all input/output parameters:
   1. Format/syntax rules
   2. Required/Optional
   3. Valid values
   4. Default values if applicable
   5. Parameters dependencies
   6. Identification of if a value is sourced as an output from a other API
   7. Data Types with applicable restrictions for example alpha-numeric/ascii string with size and allowable characters etc.

Default Values if applicable

## FortiGate Show Configured Static Routes

1. This API is used to get the routing table from a FortiGate VM. This is to ensure that the static routes configured to the Management and Public IP networks are present as well as the BGP routes
   1. To get the routing table of an authenticated session
      1. **URL**: https://<FGT IP>/api/v2/cmdb/router/static
      2. **Request Method**: GET
      3. **Input Data**: None
   2. The URL and Request Method are static values. Count as a query parameter is used to get the number of routes to be displayed
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url":

“https://<FGT IP>/api/v2/monitor/router/ipv4?count=2",

"method": "GET",

“cookie”: “APSCOOKIE VALUE”

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists the active routes in the routing table of the FortiGate VM. This API call can be used to make sure that the static routes to the management, IDN and Internet interfaces are configured.

The expected message is detailed below. The VM should have at a minimum 3 routes configured: to the internet, to the management gateway and to the IDN gateway. The expected fields and interesting responses are highlighted and marked in red below.

The below example has a default route (dst is the field name) to the management and internet routes. This can be ingored.

REQUEST:

GET https://192.168.186.130/api/v2/cmdb/router/static

RESPONSE:

GET https://192.168.186.130/api/v2/cmdb/router/static

{

"version" : "v5.4.1",

"path" : "router",

"build" : 1064,

"http\_method" : "GET",

"http\_status" : 200,

"name" : "static",

"status" : "success",

"results" : [

{

"weight" : 0,

"priority" : 5,

"dst" : "0.0.0.0 0.0.0.0",

"distance" : 10,

"seq-num" : 1,

"comment" : "Management Route",

"dstaddr" : "",

"device" : "port1",

"internet-service-custom" : "",

"status" : "enable",

"gateway" : "192.168.186.2",

"internet-service" : 0,

"dynamic-gateway" : "disable",

"virtual-wan-link" : "disable",

"q\_origin\_key" : "1",

"blackhole" : "disable"

},

{

"seq-num" : 2,

"distance" : 10,

"dstaddr" : "",

"device" : "port4",

"comment" : "IDN Route",

"weight" : 0,

"dst" : "172.16.100.0 255.255.255.0",

"priority" : 0,

"virtual-wan-link" : "disable",

"dynamic-gateway" : "disable",

"blackhole" : "disable",

"q\_origin\_key" : "2",

"gateway" : "172.16.100.1",

"status" : "enable",

"internet-service-custom" : "",

"internet-service" : 0

},

{

"dynamic-gateway" : "disable",

"virtual-wan-link" : "disable",

"blackhole" : "disable",

"q\_origin\_key" : "3",

"gateway" : "1.2.3.1",

"status" : "enable",

"internet-service-custom" : "",

"internet-service" : 0,

"seq-num" : 3,

"distance" : 10,

"dstaddr" : "",

"device" : "port5",

"comment" : "Default Route",

"weight" : 0,

"dst" : "0.0.0.0 0.0.0.0",

"priority" : 0

}

],

"vdom" : "root",

"serial" : "FGVM010000069826"

}

## FortiGate Get Routing Table

1. This API is used to get the routing table from a FortiGate VM. This is to ensure that the static routes configured to the Management and Public IP networks are present as well as the BGP routes
   1. To get the routing table of an authenticated session
      1. **URL**: https://<FGT IP>/api/v2/cmdb/router/ipv4?count=<no of routes to display>
      2. **Request Method**: GET
      3. **Input Data**: count as a query parameter is required.
   2. The URL and Request Method are static values. Count as a query parameter is used to get the number of routes to be displayed
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url":

“https://<FGT IP>/api/v2/monitor/router/ipv4?count=2",

"method": "GET",

“cookie”: “APSCOOKIE VALUE”

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists the active routes in the routing table of the FortiGate VM. This API call can be used to make sure that the static routes to the management, IDN and Internet interfaces configured are present in the routing table ablong with the BGP routes from the PIP network. The difference between this call and the one from 3.5 is that this API call shows that the route exists in the routing table as opposed to just being configured.

The expected message is detailed below. The VM should have at a minimum 3 routes configured and available on the routing table: to the internet, to the management gateway and to the IDN gateway.

The below example has a default route (dst is the field name) to the management and internet routes. This can be ingored. The example below shows the management, IDN networks as connected routes which are highlighted. The expected fields and interesting responses are highlighted and marked in red below.

**Request Response Example**

*Response Code*: 200 OK

*Response Headers:*

Cache-Control: no-cache, must-revalidate

Connection: Keep-Alive

Content-Encoding: gzip

Content-Length: 286

Content-Security-Policy: frame-ancestors 'self'

Content-Type: application/json

Date: Wed, 26 Oct 2016 23:22:35 GMT

Keep-Alive: timeout=10, max=20

Server: xxxxxxxx-xxxxx

X-Frame-Options: SAMEORIGIN

X-UA-Compatible: IE=Edge

*Response Body:*

REQUEST:

GET https://192.168.186.130/api/v2/monitor/router/ipv4?count=10

RESPONSE:

GET https://192.168.186.130/api/v2/monitor/router/ipv4?count=10

{

"http\_method" : "GET",

"action" : "select",

"status" : "success",

"path" : "router",

"results" : [

{

"type" : "static",

"distance" : 10,

"gateway" : "192.168.186.2",

"ip\_version" : 4,

"interface" : "port1",

"metric" : 0,

"ip\_mask" : "0.0.0.0/0"

},

{

"type" : "connect",

"gateway" : "0.0.0.0",

"distance" : 0,

"ip\_version" : 4,

"interface" : "port5",

"metric" : 0,

"ip\_mask" : "1.2.3.0/24"

},

{

"gateway" : "0.0.0.0",

"distance" : 0,

"interface" : "port4",

"ip\_version" : 4,

"metric" : 0,

"type" : "connect",

"ip\_mask" : "172.16.100.0/24"

},

{

"type" : "connect",

"distance" : 0,

"gateway" : "0.0.0.0",

"interface" : "port2.v100",

"metric" : 0,

"ip\_version" : 4,

"ip\_mask" : "192.168.100.0/30"

},

{

"gateway" : "0.0.0.0",

"distance" : 0,

"interface" : "port3.v100",

"ip\_version" : 4,

"metric" : 0,

"type" : "connect",

"ip\_mask" : "192.168.100.4/30"

},

{

"type" : "connect",

"interface" : "port1",

"metric" : 0,

"ip\_version" : 4,

"gateway" : "0.0.0.0",

"distance" : 0,

"ip\_mask" : "192.168.186.0/24"

}

],

"version" : "v5.4.1",

"serial" : "FGVM010000069826",

"build" : 1064,

"vdom" : "root",

"name" : "ipv4"

}

## FortiGate List Firewall Policies

1. This API is used to get all the firewall policies configured on the FortiGate.
   1. To list firewall policies configured on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy
      2. **Request Method**: GET
      3. **Input Data**: none needed
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

"url":

“https://<FGT IP>/api/v2/cmdb/firewall/policy",

"method": "GET",

“cookie”: “APSCOOKIE VALUE”

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists the configured firewall policies on the FortiGate VM. This API call can be used to make sure that at a minimum 2 firewall policies exist that allow traffic from the PIP networks to the internet. The entire response is below and the fields that are mandatory and expected are detailed as well.

**Request Response Example**

*Response Code*: 200 OK

RESPONSE:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

{

"http\_status" : 200,

"vdom" : "root",

"build" : 1064,

"path" : "firewall",

"http\_method" : "GET",

"status" : "success",

"serial" : "FGVM010000069826",

"results" : [

{

"ips-sensor" : "",

"schedule-timeout" : "disable",

"logtraffic" : "utm",

"action" : "accept",

"vlan-cos-fwd" : 255,

"match-vip" : "disable",

"permit-any-host" : "disable",

"custom-log-fields" : [],

"disclaimer" : "disable",

"status" : "enable",

"auth-redirect-addr" : "",

"wanopt-detection" : "active",

"wsso" : "enable",

"redirect-url" : "",

"ssl-ssh-profile" : "",

"name" : "PIP1\_to\_Internet\_Essential",

"devices" : [],

"ntlm" : "disable",

"dnsfilter-profile" : "",

"utm-status" : "disable",

"timeout-send-rst" : "disable",

"icap-profile" : "",

"diffserv-reverse" : "disable",

"service-negate" : "disable",

"rtp-addr" : [],

"wanopt-profile" : "",

"ntlm-guest" : "disable",

"rtp-nat" : "disable",

"capture-packet" : "disable",

"diffservcode-rev" : "000000",

"auth-cert" : "",

"dstaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"webcache-https" : "disable",

"ssl-mirror-intf" : [],

"webfilter-profile" : "",

"dlp-sensor" : "",

"rsso" : "disable",

"ssl-mirror" : "disable",

"diffservcode-forward" : "000000",

"delay-tcp-npu-sessoin" : "disable",

"tags" : [],

"permit-stun-host" : "disable",

"natinbound" : "disable",

"traffic-shaper-reverse" : "",

"session-ttl" : 0,

"comments" : "",

"inbound" : "disable",

"webcache" : "disable",

"srcaddr-negate" : "disable",

"per-ip-shaper" : "",

"av-profile" : "",

"learning-mode" : "disable",

"fsso-agent-for-ntlm" : "",

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"waf-profile" : "",

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"voip-profile" : "",

"ippool" : "disable",

"wanopt-peer" : "",

"profile-group" : "",

"tcp-mss-sender" : 0,

"wccp" : "disable",

"wanopt" : "disable",

"wanopt-passive-opt" : "default",

"identity-based-route" : "",

"outbound" : "disable",

"vpntunnel" : "",

"profile-type" : "single",

"casi-profile" : "",

"dstaddr-negate" : "disable",

"scan-botnet-connections" : "disable",

"ntlm-enabled-browsers" : [],

"application-list" : "",

"policyid" : 1,

"traffic-shaper" : "",

"schedule" : "always",

"srcintf" : [

{

"name" : "port2.v100",

"q\_origin\_key" : "port2.v100"

}

],

"natoutbound" : "disable",

"q\_origin\_key" : "1",

"poolname" : [],

"profile-protocol-options" : "",

"captive-portal-exempt" : "disable",

"dsri" : "disable",

"spamfilter-profile" : "",

"replacemsg-override-group" : "",

"users" : [],

"natip" : "0.0.0.0 0.0.0.0",

"uuid" : "b874d658-9c92-51e6-b6e8-687e361c10cd",

"service" : [

{

"q\_origin\_key" : "ALL",

"name" : "ALL"

}

],

"auth-path" : "disable",

"global-label" : "",

"groups" : [],

"fsso" : "disable",

"nat" : "enable",

"vlan-cos-rev" : 255,

"firewall-session-dirty" : "check-all",

"label" : "",

"diffserv-forward" : "disable",

"block-notification" : "disable",

"tcp-mss-receiver" : 0,

"fixedport" : "disable",

"logtraffic-start" : "disable",

"send-deny-packet" : "disable"

},

{

"match-vip" : "disable",

"vlan-cos-fwd" : 255,

"permit-any-host" : "disable",

"disclaimer" : "disable",

"custom-log-fields" : [],

"ips-sensor" : "",

"logtraffic" : "utm",

"action" : "accept",

"schedule-timeout" : "disable",

"dnsfilter-profile" : "",

"ntlm" : "disable",

"utm-status" : "disable",

"timeout-send-rst" : "disable",

"icap-profile" : "",

"diffserv-reverse" : "disable",

"wanopt-detection" : "active",

"auth-redirect-addr" : "",

"status" : "enable",

"wsso" : "enable",

"name" : "PIP2\_to\_Internet\_Essential",

"devices" : [],

"ssl-ssh-profile" : "",

"redirect-url" : "",

"rtp-nat" : "disable",

"capture-packet" : "disable",

"webcache-https" : "disable",

"dstaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"auth-cert" : "",

"diffservcode-rev" : "000000",

"service-negate" : "disable",

"rtp-addr" : [],

"wanopt-profile" : "",

"ntlm-guest" : "disable",

"session-ttl" : 0,

"comments" : "",

"webcache" : "disable",

"inbound" : "disable",

"srcaddr-negate" : "disable",

"webfilter-profile" : "",

"ssl-mirror-intf" : [],

"rsso" : "disable",

"dlp-sensor" : "",

"diffservcode-forward" : "000000",

"ssl-mirror" : "disable",

"tags" : [],

"traffic-shaper-reverse" : "",

"natinbound" : "disable",

"permit-stun-host" : "disable",

"delay-tcp-npu-sessoin" : "disable",

"ippool" : "disable",

"profile-group" : "",

"wanopt-peer" : "",

"wccp" : "disable",

"tcp-mss-sender" : 0,

"wanopt-passive-opt" : "default",

"wanopt" : "disable",

"per-ip-shaper" : "",

"fsso-agent-for-ntlm" : "",

"learning-mode" : "disable",

"av-profile" : "",

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"voip-profile" : "",

"waf-profile" : "",

"dstintf" : [

{

"q\_origin\_key" : "port5",

"name" : "port5"

}

],

"application-list" : "",

"ntlm-enabled-browsers" : [],

"policyid" : 2,

"traffic-shaper" : "",

"schedule" : "always",

"outbound" : "disable",

"identity-based-route" : "",

"dstaddr-negate" : "disable",

"casi-profile" : "",

"vpntunnel" : "",

"profile-type" : "single",

"scan-botnet-connections" : "disable",

"spamfilter-profile" : "",

"replacemsg-override-group" : "",

"uuid" : "9449d5d8-9c8a-51e6-77d7-2da1f31c99c0",

"users" : [],

"natip" : "0.0.0.0 0.0.0.0",

"srcintf" : [

{

"q\_origin\_key" : "port3.v100",

"name" : "port3.v100"

}

],

"natoutbound" : "disable",

"q\_origin\_key" : "2",

"profile-protocol-options" : "",

"poolname" : [],

"dsri" : "disable",

"captive-portal-exempt" : "disable",

"diffserv-forward" : "disable",

"block-notification" : "disable",

"label" : "",

"tcp-mss-receiver" : 0,

"logtraffic-start" : "disable",

"fixedport" : "disable",

"send-deny-packet" : "disable",

"service" : [

{

"q\_origin\_key" : "ALL",

"name" : "ALL"

}

],

"auth-path" : "disable",

"global-label" : "",

"fsso" : "disable",

"groups" : [],

"vlan-cos-rev" : 255,

"firewall-session-dirty" : "check-all",

"nat" : "enable"

}

],

"name" : "policy",

"version" : "v5.4.1"

}

**Example Expected CLI Configuration – Essential**

config firewall policy

edit 1

set name "PIP1\_to\_Internet\_Essential"

set uuid b874d658-9c92-51e6-b6e8-687e361c10cd

set srcintf "port2.v100"

set dstintf "port5"

set srcaddr "all"

set dstaddr "all"

set action accept

set schedule "always"

set service "ALL"

set nat enable

next

edit 2

set name "PIP2\_to\_Internet\_Essential"

set uuid 9449d5d8-9c8a-51e6-77d7-2da1f31c99c0

set srcintf "port3.v100"

set dstintf "port5"

set srcaddr "all"

set dstaddr "all"

set action accept

set schedule "always"

set service "ALL"

set nat enable

next

end

## FortiGate Link Monitor

1. This API is used to retrieve the availability of the next hop to the FortiGate for all the interfaces being used.
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/monitor/system/link-monitor
      2. **Request Method**: GET
      3. **Input Data**: none needed
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example**

REQUEST:

GET https://192.168.186.130/api/v2/monitor/system/link-monitor

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that lists the availability of the next hop to the FortiGate VMs interfaces.

The expected message is that the status is UP as shown for the management interface below. The VM should have at a minimum 4 interfaces always UP and available: to the internet, to the management gateway and to the IDN gateway and to one of the PIP gateways.

The below example has a few interfaces which are down due to the VM setup. This can be ingored. The example below shows the management, as UP. The expected fields and interesting responses are highlighted and marked in red below.

**Response Example**

RESPONSE:

GET https://192.168.186.130/api/v2/monitor/system/link-monitor

{

"action" : "select",

"results" : {

"port3-pip-monitor" : {

"port3.v100" : {

"status" : "down"

}

},

"port2-pip-monitor" : {

"port2.v100" : {

"status" : "down"

}

},

"mgmt-intf-monitor" : {

"port1" : {

"rx\_bandwidth" : 801,

"session" : 66,

"packet\_loss" : 0,

"latency" : 0.166433,

"status" : "up",

"jitter" : 0.21,

"tx\_bandwidth" : 607

}

},

"internet-monitor" : {

"port5" : {

"status" : "down"

}

},

"idn-monitor" : {

"port4" : {

"status" : "down"

}

}

},

"status" : "success", 🡪 API call was succesful

"vdom" : "root",

"version" : "v5.4.1",

"serial" : "FGVM010000069826",

"name" : "link-monitor",

"http\_method" : "GET",

"build" : 1064,

"path" : "system"

}

1. For all input/output parameters:
   1. Format/syntax rules
   2. Required/Optional
   3. Valid values
   4. Default values if applicable
   5. Parameters dependencies
   6. Identification of if a value is sourced as an output from a other API
   7. Data Types with applicable restrictions for example alpha-numeric/ascii string with size and allowable characters etc.
   8. Default Values if applicable

## FortiGate Editing Firewall Policy – Converting from Essential to Core Service Package

1. This API is used to add security profiles to the 2 existing firewall policies and convert the firewall configuration to the Core service package level
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy/<policy id>
         1. Policy ID for the example configurations are 1 and 2, but can be obtained by using the API call for checking all firewall policies and searching for the policyid
      2. **Request Method**: PUT
      3. **Input Data**: JSON Values in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that shows the firewall policyID that was modified and if it was successful and FortiGate VM.

To ensure that the firewall policy was configured appropriately you will need to run the API call under 3.7 which lists the firewall policy, again.

**Request Example – There are 2 requests made, 1 for each firewall policy that needs to be modified**

**REQUEST 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"profile-protocol-options" : "default",

"ips-sensor" : "default",

"ssl-ssh-profile" : "certificate-inspection",

"utm-status" : "enable",

"application-list" : "default"

}

**RESPONSE 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"results" : {

"mkey" : "1"

},

"http\_status" : 200,

"name" : "policy",

"http\_method" : "PUT",

"vdom" : "root",

"path" : "firewall",

"mkey" : "1",

"serial" : "FGVM010000069826",

"build" : 1064,

"status" : "success",

"version" : "v5.4.1"

}

**REQUEST 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"application-list" : "default",

"ssl-ssh-profile" : "certificate-inspection",

"ips-sensor" : "default",

"profile-protocol-options" : "default",

"utm-status" : "enable"

}

**RESPONSE 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"http\_method" : "PUT",

"build" : 1064,

"path" : "firewall",

"results" : {

"mkey" : "2"

},

"http\_status" : 200,

"status" : "success",

"serial" : "FGVM010000069826",

"name" : "policy",

"mkey" : "2",

"vdom" : "root",

"version" : "v5.4.1"

}

**Confirming the Firewall policy change was successful**

The API call from 3.7 is run again and the fields changed (compared to 3.7) are marked below in red

REQUEST:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

RESPONSE:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

{

"vdom" : "root",

"status" : "success", 🡪 API call was succesful

"name" : "policy",

"build" : 1064,

"path" : "firewall",

"http\_method" : "GET",

"http\_status" : 200,

"version" : "v5.4.1",

"results" : [

{

"status" : "enable",

"icap-profile" : "",

"uuid" : "b874d658-9c92-51e6-b6e8-687e361c10cd",

"srcaddr-negate" : "disable",

"tcp-mss-sender" : 0,

"permit-any-host" : "disable",

"send-deny-packet" : "disable",

"wanopt-detection" : "active",

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"schedule" : "always",

"ntlm-guest" : "disable",

"application-list" : "default",

"schedule-timeout" : "disable",

"wanopt-profile" : "",

"ssl-mirror-intf" : [],

"webcache" : "disable",

"disclaimer" : "disable",

"auth-redirect-addr" : "",

"ntlm-enabled-browsers" : [],

"profile-group" : "",

"comments" : "",

"match-vip" : "disable",

"delay-tcp-npu-sessoin" : "disable",

"timeout-send-rst" : "disable",

"q\_origin\_key" : "1",

"traffic-shaper" : "",

"redirect-url" : "",

"block-notification" : "disable",

"learning-mode" : "disable",

"outbound" : "disable",

"fixedport" : "disable",

"dstaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"groups" : [],

"voip-profile" : "",

"fsso" : "disable",

"casi-profile" : "",

"dnsfilter-profile" : "",

"nat" : "enable",

"logtraffic" : "utm",

"name" : "PIP1\_to\_Internet\_Essential",

"dstaddr-negate" : "disable",

"dsri" : "disable",

"users" : [],

"fsso-agent-for-ntlm" : "",

"wsso" : "enable",

"per-ip-shaper" : "",

"policyid" : 1,

"global-label" : "",

"waf-profile" : "",

"tcp-mss-receiver" : 0,

"session-ttl" : 0,

"diffserv-reverse" : "disable",

"identity-based-route" : "",

"webcache-https" : "disable",

"profile-protocol-options" : "default",

"action" : "accept",

"tags" : [],

"spamfilter-profile" : "",

"diffserv-forward" : "disable",

"replacemsg-override-group" : "",

"profile-type" : "single",

"diffservcode-rev" : "000000",

"auth-path" : "disable",

"wanopt-passive-opt" : "default",

"custom-log-fields" : [],

"firewall-session-dirty" : "check-all",

"wanopt-peer" : "",

"poolname" : [],

"service-negate" : "disable",

"scan-botnet-connections" : "disable",

"srcintf" : [

{

"q\_origin\_key" : "port2.v100",

"name" : "port2.v100"

}

],

"rtp-nat" : "disable",

"ssl-mirror" : "disable",

"rsso" : "disable",

"label" : "",

"diffservcode-forward" : "000000",

"av-profile" : "",

"captive-portal-exempt" : "disable",

"ips-sensor" : "default",

"webfilter-profile" : "",

"auth-cert" : "",

"permit-stun-host" : "disable",

"dlp-sensor" : "",

"natinbound" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"capture-packet" : "disable",

"natip" : "0.0.0.0 0.0.0.0",

"vpntunnel" : "",

"wccp" : "disable",

"natoutbound" : "disable",

"rtp-addr" : [],

"ippool" : "disable",

"utm-status" : "enable",

"traffic-shaper-reverse" : "",

"inbound" : "disable",

"wanopt" : "disable",

"devices" : [],

"vlan-cos-fwd" : 255,

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"vlan-cos-rev" : 255,

"service" : [

{

"q\_origin\_key" : "ALL",

"name" : "ALL"

}

],

"logtraffic-start" : "disable",

"ntlm" : "disable"

},

{

"nat" : "enable",

"logtraffic" : "utm",

"dnsfilter-profile" : "",

"groups" : [],

"fsso" : "disable",

"casi-profile" : "",

"voip-profile" : "",

"dstaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"fixedport" : "disable",

"outbound" : "disable",

"learning-mode" : "disable",

"waf-profile" : "",

"global-label" : "",

"policyid" : 2,

"per-ip-shaper" : "",

"wsso" : "enable",

"fsso-agent-for-ntlm" : "",

"dsri" : "disable",

"users" : [],

"dstaddr-negate" : "disable",

"name" : "PIP2\_to\_Internet\_Essential",

"application-list" : "default",

"schedule-timeout" : "disable",

"ntlm-guest" : "disable",

"schedule" : "always",

"srcaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"wanopt-detection" : "active",

"send-deny-packet" : "disable",

"permit-any-host" : "disable",

"tcp-mss-sender" : 0,

"srcaddr-negate" : "disable",

"uuid" : "9449d5d8-9c8a-51e6-77d7-2da1f31c99c0",

"status" : "enable",

"icap-profile" : "",

"block-notification" : "disable",

"redirect-url" : "",

"traffic-shaper" : "",

"q\_origin\_key" : "2",

"timeout-send-rst" : "disable",

"delay-tcp-npu-sessoin" : "disable",

"match-vip" : "disable",

"comments" : "",

"profile-group" : "",

"ntlm-enabled-browsers" : [],

"auth-redirect-addr" : "",

"webcache" : "disable",

"disclaimer" : "disable",

"wanopt-profile" : "",

"ssl-mirror-intf" : [],

"inbound" : "disable",

"traffic-shaper-reverse" : "",

"utm-status" : "enable",

"ippool" : "disable",

"rtp-addr" : [],

"natoutbound" : "disable",

"wccp" : "disable",

"vpntunnel" : "",

"capture-packet" : "disable",

"natip" : "0.0.0.0 0.0.0.0",

"natinbound" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"dlp-sensor" : "",

"ntlm" : "disable",

"logtraffic-start" : "disable",

"service" : [

{

"name" : "ALL",

"q\_origin\_key" : "ALL"

}

],

"vlan-cos-rev" : 255,

"dstintf" : [

{

"q\_origin\_key" : "port5",

"name" : "port5"

}

],

"vlan-cos-fwd" : 255,

"wanopt" : "disable",

"devices" : [],

"replacemsg-override-group" : "",

"diffserv-forward" : "disable",

"spamfilter-profile" : "",

"tags" : [],

"action" : "accept",

"profile-protocol-options" : "default",

"webcache-https" : "disable",

"identity-based-route" : "",

"diffserv-reverse" : "disable",

"session-ttl" : 0,

"tcp-mss-receiver" : 0,

"auth-cert" : "",

"permit-stun-host" : "disable",

"webfilter-profile" : "",

"captive-portal-exempt" : "disable",

"ips-sensor" : "default",

"av-profile" : "",

"diffservcode-forward" : "000000",

"label" : "",

"ssl-mirror" : "disable",

"rtp-nat" : "disable",

"rsso" : "disable",

"srcintf" : [

{

"name" : "port3.v100",

"q\_origin\_key" : "port3.v100"

}

],

"scan-botnet-connections" : "disable",

"service-negate" : "disable",

"poolname" : [],

"custom-log-fields" : [],

"firewall-session-dirty" : "check-all",

"wanopt-passive-opt" : "default",

"wanopt-peer" : "",

"auth-path" : "disable",

"diffservcode-rev" : "000000",

"profile-type" : "single"

}

],

"serial" : "FGVM010000069826"

}

## FortiGate Editing Firewall Policy – Converting from Core to Complete Service Package

1. This API is used to add security profiles to the 2 existing firewall policies and convert the firewall configuration to the Complete service package level
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy/<policy id>
         1. Policy ID for the example configurations are 1 and 2, but can be obtained by using the API call for checking all firewall policies and searching for the policyid
      2. **Request Method**: PUT
      3. **Input Data**: JSON Values in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – There are 2 requests made, 1 for wach firewall policy that needs to be modified**

**REQUEST 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"av-profile" : "default",

"dnsfilter-profile" : "default",

"webfilter-profile" : "default",

"spamfilter-profile" : "default"

}

**Expected Response Details**

The expected response when this API call is made is a set of JSON values that shows the firewall policyID that was modified and if it was successful and FortiGate VM.

To ensure that the firewall policy was configured appropriately you will need to run the API call under 3.7 which lists the firewall policy, again.

**RESPONSE 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"status" : "success",

"http\_status" : 200,

"results" : {

"mkey" : "1"

},

"serial" : "FGVM010000069826",

"mkey" : "1",

"http\_method" : "PUT",

"path" : "firewall",

"vdom" : "root",

"version" : "v5.4.1",

"name" : "policy",

"build" : 1064

}

**REQUEST 2**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"av-profile" : "default",

"spamfilter-profile" : "default",

"webfilter-profile" : "default",

"dnsfilter-profile" : "default"

}

**RESPONSE 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"vdom" : "root",

"name" : "policy",

"path" : "firewall",

"status" : "success",

"results" : {

"mkey" : "2"

},

"build" : 1064,

"version" : "v5.4.1",

"http\_status" : 200,

"mkey" : "2",

"http\_method" : "PUT",

"serial" : "FGVM010000069826"

}

**Confirming the Firewall policy change was successful**

The API call from 3.7 is run again and the fields changed (compared to 3.9) are marked below in red

REQUEST:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

RESPONSE:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

{

"results" : [

{

"service" : [

{

"q\_origin\_key" : "ALL",

"name" : "ALL"

}

],

"wanopt" : "disable",

"poolname" : [],

"tcp-mss-sender" : 0,

"utm-status" : "enable",

"permit-any-host" : "disable",

"webfilter-profile" : "default",

"profile-protocol-options" : "default",

"ssl-mirror-intf" : [],

"ntlm-enabled-browsers" : [],

"firewall-session-dirty" : "check-all",

"diffserv-forward" : "disable",

"wanopt-peer" : "",

"natip" : "0.0.0.0 0.0.0.0",

"casi-profile" : "",

"rsso" : "disable",

"waf-profile" : "",

"ippool" : "disable",

"icap-profile" : "",

"av-profile" : "default",

"nat" : "enable",

"wanopt-passive-opt" : "default",

"wanopt-profile" : "",

"learning-mode" : "disable",

"fixedport" : "disable",

"session-ttl" : 0,

"profile-group" : "",

"srcaddr-negate" : "disable",

"fsso" : "disable",

"custom-log-fields" : [],

"diffserv-reverse" : "disable",

"logtraffic" : "utm",

"schedule-timeout" : "disable",

"action" : "accept",

"policyid" : 1,

"capture-packet" : "disable",

"captive-portal-exempt" : "disable",

"redirect-url" : "",

"vpntunnel" : "",

"logtraffic-start" : "disable",

"webcache" : "disable",

"per-ip-shaper" : "",

"profile-type" : "single",

"dstaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"auth-path" : "disable",

"block-notification" : "disable",

"auth-cert" : "",

"dstaddr-negate" : "disable",

"wccp" : "disable",

"vlan-cos-rev" : 255,

"send-deny-packet" : "disable",

"global-label" : "",

"schedule" : "always",

"outbound" : "disable",

"q\_origin\_key" : "1",

"diffservcode-rev" : "000000",

"auth-redirect-addr" : "",

"service-negate" : "disable",

"ssl-mirror" : "disable",

"identity-based-route" : "",

"diffservcode-forward" : "000000",

"fsso-agent-for-ntlm" : "",

"traffic-shaper" : "",

"ntlm" : "disable",

"tcp-mss-receiver" : 0,

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"dnsfilter-profile" : "default",

"rtp-addr" : [],

"scan-botnet-connections" : "disable",

"natinbound" : "disable",

"dsri" : "disable",

"users" : [],

"srcintf" : [

{

"q\_origin\_key" : "port2.v100",

"name" : "port2.v100"

}

],

"permit-stun-host" : "disable",

"application-list" : "default",

"dlp-sensor" : "",

"devices" : [],

"webcache-https" : "disable",

"tags" : [],

"voip-profile" : "",

"groups" : [],

"status" : "enable",

"timeout-send-rst" : "disable",

"vlan-cos-fwd" : 255,

"rtp-nat" : "disable",

"disclaimer" : "disable",

"match-vip" : "disable",

"inbound" : "disable",

"traffic-shaper-reverse" : "",

"ips-sensor" : "default",

"uuid" : "b874d658-9c92-51e6-b6e8-687e361c10cd",

"spamfilter-profile" : "default",

"replacemsg-override-group" : "",

"ntlm-guest" : "disable",

"wanopt-detection" : "active",

"label" : "",

"name" : "PIP1\_to\_Internet\_Essential",

"delay-tcp-npu-sessoin" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"natoutbound" : "disable",

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"comments" : "",

"wsso" : "enable"

},

{

"wanopt" : "disable",

"tcp-mss-sender" : 0,

"poolname" : [],

"service" : [

{

"name" : "ALL",

"q\_origin\_key" : "ALL"

}

],

"firewall-session-dirty" : "check-all",

"diffserv-forward" : "disable",

"wanopt-peer" : "",

"natip" : "0.0.0.0 0.0.0.0",

"casi-profile" : "",

"waf-profile" : "",

"rsso" : "disable",

"utm-status" : "enable",

"permit-any-host" : "disable",

"webfilter-profile" : "default",

"ssl-mirror-intf" : [],

"profile-protocol-options" : "default",

"ntlm-enabled-browsers" : [],

"icap-profile" : "",

"nat" : "enable",

"av-profile" : "default",

"ippool" : "disable",

"session-ttl" : 0,

"profile-group" : "",

"fsso" : "disable",

"srcaddr-negate" : "disable",

"wanopt-passive-opt" : "default",

"wanopt-profile" : "",

"fixedport" : "disable",

"learning-mode" : "disable",

"schedule-timeout" : "disable",

"policyid" : 2,

"action" : "accept",

"custom-log-fields" : [],

"diffserv-reverse" : "disable",

"logtraffic" : "utm",

"captive-portal-exempt" : "disable",

"redirect-url" : "",

"vpntunnel" : "",

"logtraffic-start" : "disable",

"webcache" : "disable",

"capture-packet" : "disable",

"block-notification" : "disable",

"auth-cert" : "",

"profile-type" : "single",

"per-ip-shaper" : "",

"auth-path" : "disable",

"dstaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"vlan-cos-rev" : 255,

"global-label" : "",

"send-deny-packet" : "disable",

"outbound" : "disable",

"schedule" : "always",

"dstaddr-negate" : "disable",

"wccp" : "disable",

"ssl-mirror" : "disable",

"diffservcode-forward" : "000000",

"identity-based-route" : "",

"traffic-shaper" : "",

"fsso-agent-for-ntlm" : "",

"q\_origin\_key" : "2",

"diffservcode-rev" : "000000",

"auth-redirect-addr" : "",

"service-negate" : "disable",

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"rtp-addr" : [],

"dnsfilter-profile" : "default",

"scan-botnet-connections" : "disable",

"natinbound" : "disable",

"dsri" : "disable",

"ntlm" : "disable",

"tcp-mss-receiver" : 0,

"application-list" : "default",

"users" : [],

"srcintf" : [

{

"q\_origin\_key" : "port3.v100",

"name" : "port3.v100"

}

],

"permit-stun-host" : "disable",

"tags" : [],

"webcache-https" : "disable",

"voip-profile" : "",

"groups" : [],

"dlp-sensor" : "",

"devices" : [],

"vlan-cos-fwd" : 255,

"rtp-nat" : "disable",

"disclaimer" : "disable",

"match-vip" : "disable",

"status" : "enable",

"timeout-send-rst" : "disable",

"uuid" : "9449d5d8-9c8a-51e6-77d7-2da1f31c99c0",

"ips-sensor" : "default",

"spamfilter-profile" : "default",

"inbound" : "disable",

"traffic-shaper-reverse" : "",

"wanopt-detection" : "active",

"label" : "",

"name" : "PIP2\_to\_Internet\_Essential",

"delay-tcp-npu-sessoin" : "disable",

"replacemsg-override-group" : "",

"ntlm-guest" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"natoutbound" : "disable",

"comments" : "",

"wsso" : "enable"

}

],

"vdom" : "root",

"http\_status" : 200,

"path" : "firewall",

"version" : "v5.4.1",

"serial" : "FGVM010000069826",

"status" : "success",

"http\_method" : "GET",

"build" : 1064,

"name" : "policy"

}

## FortiGate Editing Firewall Policy – Converting from Essential to Complete Service Package

1. This API is used to add security profiles to the 2 existing firewall policies and convert the firewall configuration to the Complete service package level
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy/<policy id>
         1. Policy ID for the example configurations are 1 and 2, but can be obtained by using the API call for checking all firewall policies and searching for the policyid
      2. **Request Method**: PUT
      3. **Input Data**: JSON Values in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – There are 2 requests made, 1 for wach firewall policy that needs to be modified**

**REQUEST 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"spamfilter-profile" : "default",

"av-profile" : "default",

"application-list" : "default",

"dnsfilter-profile" : "default",

"ssl-ssh-profile" : "certificate-inspection",

"webfilter-profile" : "default",

"ips-sensor" : "default",

"profile-protocol-options" : "default",

"utm-status" : "enable"

}

**RESPONSE 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"http\_status" : 200,

"version" : "v5.4.1",

"name" : "policy",

"results" : {

"mkey" : "1"

},

"path" : "firewall",

"http\_method" : "PUT",

"vdom" : "root",

"serial" : "FGVM010000069826",

"build" : 1064,

"status" : "success",

"mkey" : "1"

}

**REQUEST 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"av-profile" : "default",

"application-list" : "default",

"spamfilter-profile" : "default",

"ssl-ssh-profile" : "certificate-inspection",

"utm-status" : "enable",

"ips-sensor" : "default",

"dnsfilter-profile" : "default",

"profile-protocol-options" : "default",

"webfilter-profile" : "default"

}

**RESPONSE 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"status" : "success",

"version" : "v5.4.1",

"name" : "policy",

"mkey" : "2",

"results" : {

"mkey" : "2"

},

"http\_status" : 200,

"build" : 1064,

"http\_method" : "PUT",

"serial" : "FGVM010000069826",

"path" : "firewall",

"vdom" : "root"

}

**Confirming the Firewall policy change was successful**

The API call from 3.7 is run again and the fields changed (compared to 3.7) are marked below in red

REQUEST:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

RESPONSE:

GET https://192.168.186.130/api/v2/cmdb/firewall/policy

{

"results" : [

{

"service" : [

{

"q\_origin\_key" : "ALL",

"name" : "ALL"

}

],

"wanopt" : "disable",

"poolname" : [],

"tcp-mss-sender" : 0,

"utm-status" : "enable",

"permit-any-host" : "disable",

"webfilter-profile" : "default",

"profile-protocol-options" : "default",

"ssl-mirror-intf" : [],

"ntlm-enabled-browsers" : [],

"firewall-session-dirty" : "check-all",

"diffserv-forward" : "disable",

"wanopt-peer" : "",

"natip" : "0.0.0.0 0.0.0.0",

"casi-profile" : "",

"rsso" : "disable",

"waf-profile" : "",

"ippool" : "disable",

"icap-profile" : "",

"av-profile" : "default",

"nat" : "enable",

"wanopt-passive-opt" : "default",

"wanopt-profile" : "",

"learning-mode" : "disable",

"fixedport" : "disable",

"session-ttl" : 0,

"profile-group" : "",

"srcaddr-negate" : "disable",

"fsso" : "disable",

"custom-log-fields" : [],

"diffserv-reverse" : "disable",

"logtraffic" : "utm",

"schedule-timeout" : "disable",

"action" : "accept",

"policyid" : 1,

"capture-packet" : "disable",

"captive-portal-exempt" : "disable",

"redirect-url" : "",

"vpntunnel" : "",

"logtraffic-start" : "disable",

"webcache" : "disable",

"per-ip-shaper" : "",

"profile-type" : "single",

"dstaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"auth-path" : "disable",

"block-notification" : "disable",

"auth-cert" : "",

"dstaddr-negate" : "disable",

"wccp" : "disable",

"vlan-cos-rev" : 255,

"send-deny-packet" : "disable",

"global-label" : "",

"schedule" : "always",

"outbound" : "disable",

"q\_origin\_key" : "1",

"diffservcode-rev" : "000000",

"auth-redirect-addr" : "",

"service-negate" : "disable",

"ssl-mirror" : "disable",

"identity-based-route" : "",

"diffservcode-forward" : "000000",

"fsso-agent-for-ntlm" : "",

"traffic-shaper" : "",

"ntlm" : "disable",

"tcp-mss-receiver" : 0,

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"dnsfilter-profile" : "default",

"rtp-addr" : [],

"scan-botnet-connections" : "disable",

"natinbound" : "disable",

"dsri" : "disable",

"users" : [],

"srcintf" : [

{

"q\_origin\_key" : "port2.v100",

"name" : "port2.v100"

}

],

"permit-stun-host" : "disable",

"application-list" : "default",

"dlp-sensor" : "",

"devices" : [],

"webcache-https" : "disable",

"tags" : [],

"voip-profile" : "",

"groups" : [],

"status" : "enable",

"timeout-send-rst" : "disable",

"vlan-cos-fwd" : 255,

"rtp-nat" : "disable",

"disclaimer" : "disable",

"match-vip" : "disable",

"inbound" : "disable",

"traffic-shaper-reverse" : "",

"ips-sensor" : "default",

"uuid" : "b874d658-9c92-51e6-b6e8-687e361c10cd",

"spamfilter-profile" : "default",

"replacemsg-override-group" : "",

"ntlm-guest" : "disable",

"wanopt-detection" : "active",

"label" : "",

"name" : "PIP1\_to\_Internet\_Essential",

"delay-tcp-npu-sessoin" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"natoutbound" : "disable",

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"comments" : "",

"wsso" : "enable"

},

{

"wanopt" : "disable",

"tcp-mss-sender" : 0,

"poolname" : [],

"service" : [

{

"name" : "ALL",

"q\_origin\_key" : "ALL"

}

],

"firewall-session-dirty" : "check-all",

"diffserv-forward" : "disable",

"wanopt-peer" : "",

"natip" : "0.0.0.0 0.0.0.0",

"casi-profile" : "",

"waf-profile" : "",

"rsso" : "disable",

"utm-status" : "enable",

"permit-any-host" : "disable",

"webfilter-profile" : "default",

"ssl-mirror-intf" : [],

"profile-protocol-options" : "default",

"ntlm-enabled-browsers" : [],

"icap-profile" : "",

"nat" : "enable",

"av-profile" : "default",

"ippool" : "disable",

"session-ttl" : 0,

"profile-group" : "",

"fsso" : "disable",

"srcaddr-negate" : "disable",

"wanopt-passive-opt" : "default",

"wanopt-profile" : "",

"fixedport" : "disable",

"learning-mode" : "disable",

"schedule-timeout" : "disable",

"policyid" : 2,

"action" : "accept",

"custom-log-fields" : [],

"diffserv-reverse" : "disable",

"logtraffic" : "utm",

"captive-portal-exempt" : "disable",

"redirect-url" : "",

"vpntunnel" : "",

"logtraffic-start" : "disable",

"webcache" : "disable",

"capture-packet" : "disable",

"block-notification" : "disable",

"auth-cert" : "",

"profile-type" : "single",

"per-ip-shaper" : "",

"auth-path" : "disable",

"dstaddr" : [

{

"name" : "all",

"q\_origin\_key" : "all"

}

],

"vlan-cos-rev" : 255,

"global-label" : "",

"send-deny-packet" : "disable",

"outbound" : "disable",

"schedule" : "always",

"dstaddr-negate" : "disable",

"wccp" : "disable",

"ssl-mirror" : "disable",

"diffservcode-forward" : "000000",

"identity-based-route" : "",

"traffic-shaper" : "",

"fsso-agent-for-ntlm" : "",

"q\_origin\_key" : "2",

"diffservcode-rev" : "000000",

"auth-redirect-addr" : "",

"service-negate" : "disable",

"srcaddr" : [

{

"q\_origin\_key" : "all",

"name" : "all"

}

],

"rtp-addr" : [],

"dnsfilter-profile" : "default",

"scan-botnet-connections" : "disable",

"natinbound" : "disable",

"dsri" : "disable",

"ntlm" : "disable",

"tcp-mss-receiver" : 0,

"application-list" : "default",

"users" : [],

"srcintf" : [

{

"q\_origin\_key" : "port3.v100",

"name" : "port3.v100"

}

],

"permit-stun-host" : "disable",

"tags" : [],

"webcache-https" : "disable",

"voip-profile" : "",

"groups" : [],

"dlp-sensor" : "",

"devices" : [],

"vlan-cos-fwd" : 255,

"rtp-nat" : "disable",

"disclaimer" : "disable",

"match-vip" : "disable",

"status" : "enable",

"timeout-send-rst" : "disable",

"uuid" : "9449d5d8-9c8a-51e6-77d7-2da1f31c99c0",

"ips-sensor" : "default",

"spamfilter-profile" : "default",

"inbound" : "disable",

"traffic-shaper-reverse" : "",

"wanopt-detection" : "active",

"label" : "",

"name" : "PIP2\_to\_Internet\_Essential",

"delay-tcp-npu-sessoin" : "disable",

"replacemsg-override-group" : "",

"ntlm-guest" : "disable",

"ssl-ssh-profile" : "certificate-inspection",

"dstintf" : [

{

"name" : "port5",

"q\_origin\_key" : "port5"

}

],

"natoutbound" : "disable",

"comments" : "",

"wsso" : "enable"

}

],

"vdom" : "root",

"http\_status" : 200,

"path" : "firewall",

"version" : "v5.4.1",

"serial" : "FGVM010000069826",

"status" : "success",

"http\_method" : "GET",

"build" : 1064,

"name" : "policy"

}

## FortiGate Editing Firewall Policy – Converting from Complete or Core to Essential Service Package

1. This API is used to add security profiles to the 2 existing firewall policies and downgrade the firewall configuration to the Essential service package level from the Core or Complete service levels
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy/<policy id>
         1. Policy ID for the example configurations are 1 and 2, but can be obtained by using the API call for checking all firewall policies and searching for the policyid
      2. **Request Method**: PUT
      3. **Input Data**: JSON Values in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – There are 2 requests made, 1 for wach firewall policy that needs to be modified**

**REQUEST 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"utm-status" : "disable"

}

**RESPONSE 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1/

{

"http\_method" : "PUT",

"results" : {

"mkey" : "1"

},

"build" : 1064,

"version" : "v5.4.1",

"status" : "success",

"mkey" : "1",

"path" : "firewall",

"name" : "policy",

"http\_status" : 200,

"serial" : "FGVM010000069826",

"vdom" : "root"

}

**REQUEST 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"utm-status" : "disable"

}

**RESPONSE 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2/

{

"vdom" : "root",

"name" : "policy",

"build" : 1064,

"version" : "v5.4.1",

"results" : {

"mkey" : "2"

},

"serial" : "FGVM010000069826",

"status" : "success",

"http\_method" : "PUT",

"http\_status" : 200,

"path" : "firewall",

"mkey" : "2"

}

**Confirming the Firewall policy change was successful**

The API call from 3.7 is run again and the resulting response will be the same as the one under 3.7

## FortiGate Editing Firewall Policy – Converting from Complete to Core Service Package

1. This API is used to add security profiles to the 2 existing firewall policies and convert the firewall configuration to the Complete service package level
   1. To get the status of the availability of the next hop on a FortiGate
      1. **URL**: https://<FGT IP>/api/v2/cmdb/firewall/policy/<policy id>
         1. Policy ID for the example configurations are 1 and 2, but can be obtained by using the API call for checking all firewall policies and searching for the policyid
      2. **Request Method**: PUT
      3. **Input Data**: JSON Values in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – There are 2 requests made, 1 for wach firewall policy that needs to be modified**

**REQUEST 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1

{

"spamfilter-profile" : "",

"webfilter-profile" : "",

"av-profile" : "",

"dnsfilter-profile" : ""

}

**RESPONSE 1:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/1

{

"vdom" : "root",

"status" : "success",

"mkey" : "1",

"serial" : "FGVM010000069826",

"http\_status" : 200,

"name" : "policy",

"http\_method" : "PUT",

"version" : "v5.4.1",

"build" : 1064,

"path" : "firewall",

"results" : {

"mkey" : "1"

}

}

**REQUEST 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2

{

"av-profile" : "",

"dnsfilter-profile" : "",

"webfilter-profile" : "",

"spamfilter-profile" : ""

}

**RESPONSE 2:**

PUT https://192.168.186.130/api/v2/cmdb/firewall/policy/2

{

"http\_method" : "PUT",

"http\_status" : 200,

"serial" : "FGVM010000069826",

"build" : 1064,

"version" : "v5.4.1",

"results" : {

"mkey" : "2"

},

"status" : "success",

"path" : "firewall",

"vdom" : "root",

"mkey" : "2",

"name" : "policy"

}

**Confirming the Firewall policy change was successful**

The API call from 3.7 is run again and the resulting response will be the same as the one under 3.9

## FortiGate Uploading a License

1. This API is used to upload a VM license to a FortiGate
   1. To upload the license file
      1. **URL**: https://<FGT IP>/api/v2/monitor/system/vmlicense/upload
      2. **Request Method**: POST
      3. **Input Data**: Base64 encoded license as JSON value in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
      1. *The FortiGate will reboot after a license has been uploaded and will require validation from FortiGuard or a FortiManager*
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – The VM license used here has been converted to base64 and is truncated for security reasons**

REQUEST:

POST https://192.168.186.130/api/v2/monitor/system/vmlicense/upload

{

"file\_content" : "LS0tLS1CRUdJTiBGR1QgVk0gTElDRU5TRS0tLS0tDQpRQUFBQUNRaWdudFFjUHF6YUhWbk5KZjVFbWVHNzc3NGFlSFBXcU9SN0RHb1F1Qkl4UTAwbGUzZmI3QXVEZ1RDQmJhWl000000000000000000000000000000000000000”

}

RESPONSE:

POST https://192.168.186.130/api/v2/monitor/system/vmlicense/upload

{

"build" : 1064,

"serial" : "FGVM010000069826",

"path" : "system",

"http\_method" : "POST",

"version" : "v5.4.1",

"action" : "upload",

"vdom" : "root",

"name" : "vmlicense",

"status" : "success",

"http\_status" : 200

}

## FortiGate Uploading Firmware

1. This API is used to upload a new Firmware version to a FortiGate
   1. To upload the license file
      1. **URL**: https://<FGT IP>/api/v2/monitor/system/firmware/upgrade
      2. **Request Method**: POST
      3. **Input Data**: Base64 encoded firmware file as JSON value in the body
   2. The URL and Request Method are static values
   3. Success Code: 200 OK, Error Code: Any other HTTP response codes mean the call was not successful
      1. *The FortiGate will reboot after a license has been uploaded and will require validation from FortiGuard or a FortiManager*
   4. Request and Response examples applicable to the software version/VNF configuration being used for this project.

**Request Example – The firmware file is converted to base64 and is used in the file\_content json object**

REQUEST:

POST https://192.168.186.130/api/v2/monitor/system/vmlicense/upload

{

“source”: “upload”, "file\_content" : "base64 encoded file”

}

# Attachments

This section includes the lab validated template files, config files, etc. that may be needed