PAN-OS API for NGFW and Panorama

Performance and Capacity Testing

POC PAN-OS API / JAPAC POC

Jun 16, 2025

Initiative of the piece of work

- Develop expertise for advanced POC's.
- Support unique PAN solutions for the field, with expertise of the team.
- PAN-OS feature showcase to enable customer demos and evaluations.
- Train and help understand new and unique features of our products.
- Proven setups for linked demos and sandboxes.
- Establishment for further proof of functions, performance and capacity.
- Documentation to promote proven, sized and repeatable setups.
- Last but not least, establishment for further development and possibilities.

Documentation as additional references

POC Team is consistently developing technical content to accelerate testing and POC cycles. Here this is the collection of documents for various use cases. We welcome your feedback to help enhance our productivity with detailed and efficient documentation.

https://drive.google.com/drive/folders/1azplLToTYzfwynAF4gNlmC1W9ViaL9Xq?usp=drive_link

Note 9 - Subscriber-ID
Note 8 - PAN-OS API with Python
Note 7 - Explicit Web proxy w/ Network Packet Broker
Note 6 - PAN-OS SD-WAN
Note 5 - KVM w/ OVS-DPDK
Note 4 - Prisma SD-WAN POC setup v2.0
Note 3 - MFA
Note 2 - GlobalProtect w/ AD
Note 1 - User-ID

Developed tools on Github for POC and system admin https://github.com/telee0

Revision history

20250616 - v2.3 to support single-stage execution model, aka push-on-the-fly

20250116 - Updates

20230616 - Initial release

POC References - MPOV requests with VM series

POC number	API Target	Test cases
POC13663	Panorama	Capacity testing w/ 1 GB configuration

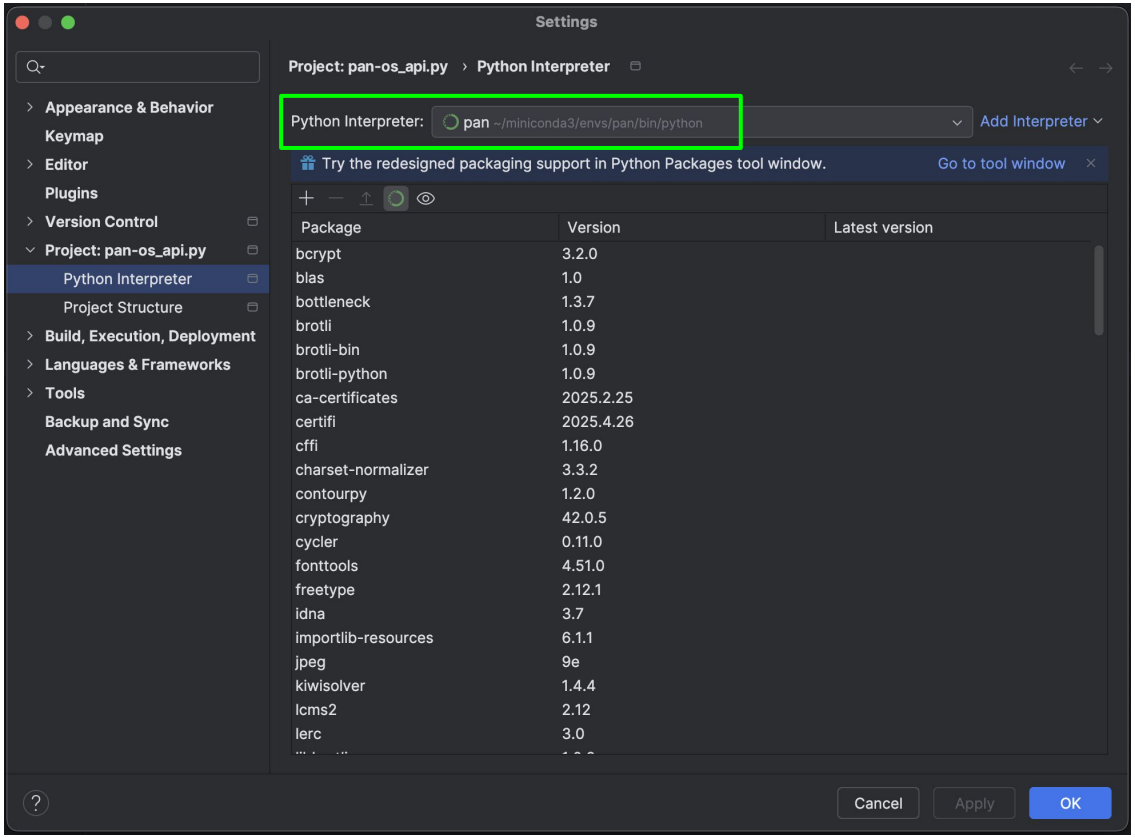
PAN-OS API

- Tools shared through GitHub https://github.com/telee0/pan-os_api.py
- Scripts rewritten in Python to generate PAN XML config and apply it through REST API.

```
$ python3 pan.py -c conf/pa.py -h <PA1_host> -u admin -v
```

- Tested on Windows, MacOS, and Linux (CentOS, Ubuntu and Ubuntu/WSL)
- Used preferably with IDE (PyCharm) so executions and configuration files (conf/*.py) can be managed easily.
- Access password may be specified in an environment variable, so that it will not be hardcoded in the configuration file.

Pycharm project settings



- Dedicated virtual runtime environment is recommended.
- This ensures a stable environment for the script from broken by package upgrade for other projects.
- Virtual runtime environment can be created with venv, minicond, etc.

Pycharm run configuration

- The main script is `pan.py`
- Script parameters:

  `-c conf/pa.py` config file

  `-h 192.168.1.251` target host

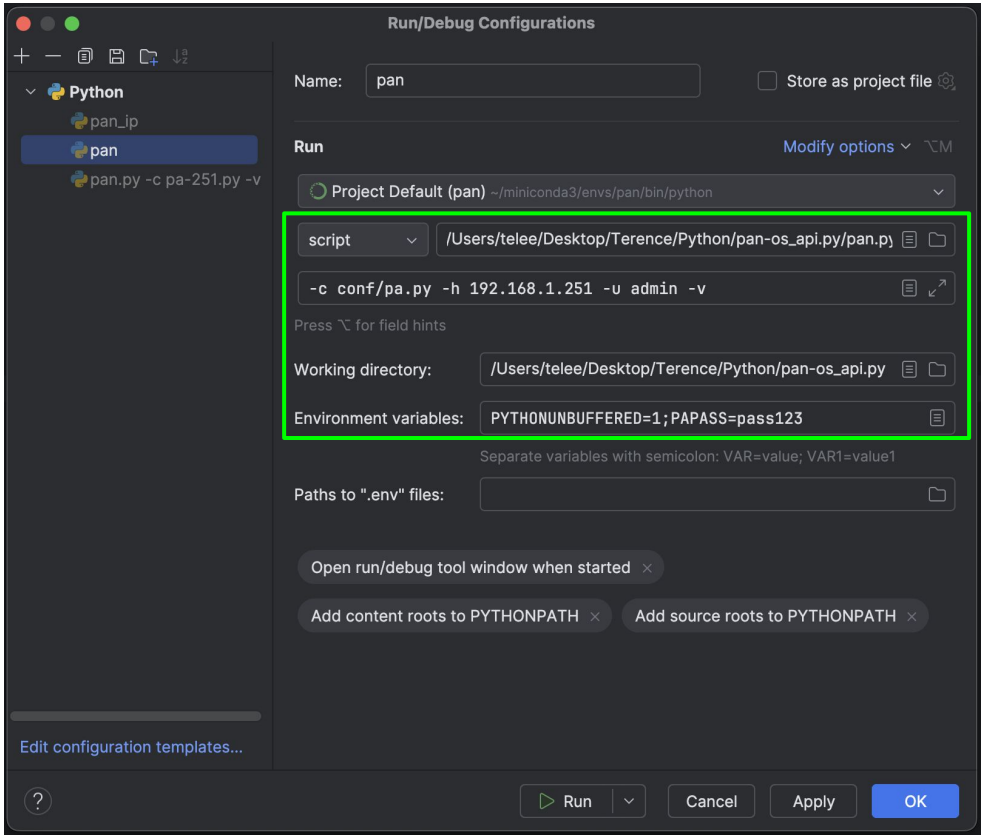
  `-u admin` login user name

  `-v` verbose mode for more details

- Environment variable for the password:

  `PAPASS=pass123` login password

  So you do not need to keep your passwords in the files before saving them on shared drives.



Required (Python) packages for the scripts

- There is a file `requirements.txt` which is recognized by PyCharm so can be used to patch the environment.
- The environment also can be checked for installed packages with `pip`.
- Use `pip install -r requirements.txt` to install the packages.

```
(pan) C:\Users\Terence>conda activate pan
(pan) C:\Users\Terence>pip list --format=freeze
Brotli==1.0.9
certifi==2025.4.26
charset-normalizer==3.3.2
idna==3.7
pip==25.1
PySocks==1.7.1
requests==2.32.3
setuptools==78.1.1
urllib3==2.3.0
wheel==0.45.1
win_inet_pton==1.1.0

(pan) C:\Users\Terence>
```

Win 11

MacOS

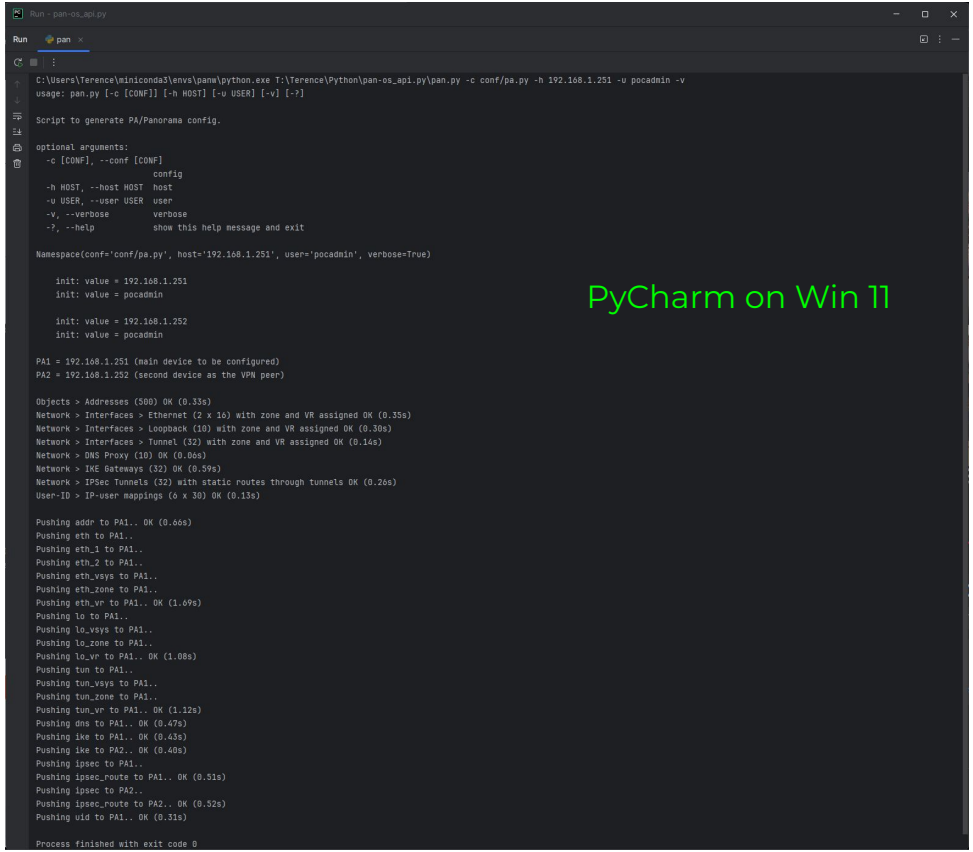
Two-stage execution model (2S)

- Python scripts to generate a set of Bash scripts, with which PA config in XML will be injected by a command-line tool through HTTP/S.
- 2S is friendly for troubleshooting, because for lab testing, DUT's various capacity limits are always touched, and those limits need to be identified and reviewed. External command-line tools has better support in this aspect with comprehensive debugging facilities.
- For example, `wget` as the HTTPS client talks to PA/PAN and posts XML through REST API.
- It can be replaced with `cURL`, or Python request module (to make it a single stage model without manual invocation of Bash scripts).

Single-stage model (1S) is also supported

- In a single-stage execution model, the Python script not only generates the configuration payloads but also directly posts them to PA/PAN using a built-in HTTP/S client.
- This eliminates the need for manual execution of the Bash scripts, streamlining the process into a single automated flow. While this model is faster and easier to integrate into CI/CD pipelines, it offers less visibility into intermediate steps, making it more suitable for internal tools or production-like environments where configuration is stable and well-tested.
- Individual configuration sets can still be removed manually with the clean scripts generated. (Or simply use Revert to running configuration on the firewall)

Sample output



Supported PA configuration

- Device > Local users
- Network > Interfaces > Ethernet (with vsys, zone and vr assignment)
- Network > Interfaces > Loopback (with vsys, zone and vr assignment)
- Network > Interfaces > Tunnel (with vsys, zone and vr assignment)
- Network > Zones
- Network > DNS Proxy
- Network > IKE Gateways
- Network > IPSec Tunnels (with static routes through tunnels)
- Objects > Addresses
- Objects > Address Groups
- Objects > Services
- Objects > Service Groups
- Objects > Custom URL Category with url.txt
- Policies > Security
- Policies > NAT
- Policies > PBF

- Network > VR > Static Routes
- Network > VR > BGP peer groups x peers
- User-ID mapping

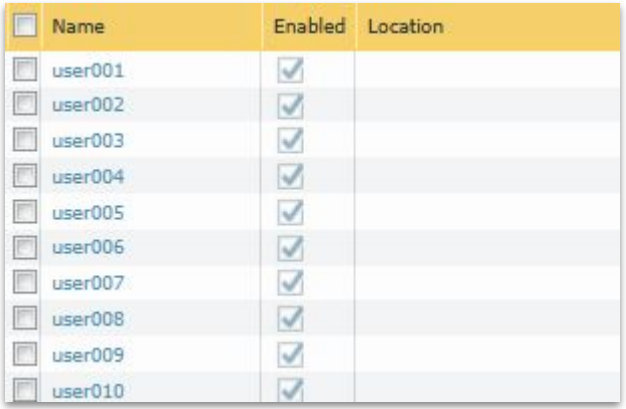
(Panorama only)

- Panorama > Device Groups
- Panorama > Templates

Device > Local users

Local users - configuration

```
# Local users
#
'N_USERS': 10,

'USER_NAME': "user{0:03d}",  # user001, user002, etc.
'USER_NAME_i': 1,
'USER_PASS': "pass123_",  # password to be stored as phash
```



Name	Enabled	Location
user001	✓	
user002	✓	
user003	✓	
user004	✓	
user005	✓	
user006	✓	
user007	✓	
user008	✓	
user009	✓	
user010	✓	

Network > Interfaces > Ethernet
Network > Interfaces > Loopback
Network > Interfaces > Tunnel

(with vsys, zone and VR assignment)

Interface configuration

- Generation of the following types of interfaces
  - Ethernet subinterfaces (L3 only)
  - Loopback interfaces
  - Tunnel interfaces
- Assignment of VSYS, ZONE and VR.
- IPv4 or IPv6 addressing, but not both (yet)
- Dependency-aware: interfaces are generated before configuration depending on them, such as IKE gateways and IPSec tunnels.

Ethernet configuration

```
'N_NET_IF_ETHERNET': 10,  # number of subinterfaces per each physical ethernet

'CONFIG_VERSION': "9.1.0",
'LHOST': "localhost.localdomain",
'VSYS': "vsys1",

'DEFAULT_ZONE1': "L3-Trust",
'DEFAULT_ZONE2': "L3-Untrust",
# 'DEFAULT_ZONE1': "L3-trust",
# 'DEFAULT_ZONE2': "L3-untrust",
'DEFAULT_ZONE3': "VPN",

'DEFAULT_VR': "default",

# ------------------------------------------------------------------------------
#
# ethernet subinterfaces, loopback and tunnel interfaces
#

'IF_ETHERNET_LIST': ["ethernet1/11", "ethernet1/12"],  # list of parent interfaces
'IF_ETHERNET_TYPE': "L3",  # only L3 is assumed
'IF_ETHERNET_TAG_i': 11,   # initial VLAN tag
# 'IF_ETHERNET_IP': None,  # "",
'IF_ETHERNET_IP': "11.11.0.1/30",
# 'IF_ETHERNET_IP': "2211:11::1/126",
'IF_ETHERNET_VR': cf['DEFAULT_VR'],
'IF_ETHERNET_ZONE': cf['DEFAULT_ZONE2'],
```

Ethernet - Configuration pushed

INTERFACE	INTERFACE TYPE	MANAGEMENT PROFILE	LINK STATE	IP ADDRESS	VIRTUAL ROUTER	TAG	VLAN / VIRTUAL-WIRE	VIRTUAL SYSTEM	SECURITY ZONE	SD-WAN INTERFACE PROFILE	UPSTREAM NAT	FEATURES	COMMENT
ethernet1/11	Layer3			none	none	Untagged	none	none	none		Disabled		
ethernet1/11.1	Layer3			11.11.0.1/30	default	11	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.2	Layer3			11.11.0.5/30	default	12	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.3	Layer3			11.11.0.9/30	default	13	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.4	Layer3			11.11.0.13/30	default	14	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.5	Layer3			11.11.0.17/30	default	15	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.6	Layer3			11.11.0.21/30	default	16	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.7	Layer3			11.11.0.25/30	default	17	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.8	Layer3			11.11.0.29/30	default	18	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.9	Layer3			11.11.0.33/30	default	19	none	vsys1	L3-Untrust		Disabled		
ethernet1/11.10	Layer3			11.11.0.37/30	default	20	none	vsys1	L3-Untrust		Disabled		
ethernet1/12	Layer3			none	none	Untagged	none	none	none		Disabled		
ethernet1/12.1	Layer3			11.11.0.41/30	default	21	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.2	Layer3			11.11.0.45/30	default	22	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.3	Layer3			11.11.0.49/30	default	23	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.4	Layer3			11.11.0.53/30	default	24	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.5	Layer3			11.11.0.57/30	default	25	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.6	Layer3			11.11.0.61/30	default	26	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.7	Layer3			11.11.0.65/30	default	27	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.8	Layer3			11.11.0.69/30	default	28	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.9	Layer3			11.11.0.73/30	default	29	none	vsys1	L3-Untrust		Disabled		
ethernet1/12.10	Layer3			11.11.0.77/30	default	30	none	vsys1	L3-Untrust		Disabled		
ethernet1/43	Layer3	Ping		1.1.0.243/16	default	Untagged	none	vsys1	L3-Trust		Disabled		
ethernet1/44	Layer3	Ping		12.12.0.243/24 13.13.0.243/24	default	Untagged	none	vsys1	L3-Untrust		Disabled		

Loopback configuration

```
'N_NET_IF_ETHERNET': 10,  # number of subinterfaces per each physical ethernet
'N_NET_IF_LOOPBACK': 10,
'N_NET_IF_TUNNEL': 10,

'CONFIG_VERSION': "9.1.0",
'LHOST': "localhost.localdomain",
'VSYS': "vsys1",

'DEFAULT_ZONE1': "L3-Trust",
'DEFAULT_ZONE2': "L3-Untrust",
# 'DEFAULT_ZONE1': "L3-trust",
# 'DEFAULT_ZONE2': "L3-untrust",
'DEFAULT_ZONE3': "VPN",

'DEFAULT_VR': "default",

# ------------------------------------------------------------------------------
#
# ethernet subinterfaces, loopback and tunnel interfaces
#

'IF_LOOPBACK_NAME_i': 10,  # i, the initial index value
# 'IF_LOOPBACK_IP': None,  # "",
# 'IF_LOOPBACK_IP': "100.0.0.1/32",
'IF_LOOPBACK_IP': "2100::1/120",
'IF_LOOPBACK_VR': cf['DEFAULT_VR'],
'IF_LOOPBACK_ZONE': cf['DEFAULT_ZONE1'],
```

IPv6 supported

Loopback - Configuration pushed

INTERFACE	MANAGEMENT PROFILE	IP ADDRESS	VIRTUAL ROUTER	VIRTUAL SYSTEM	SECURITY ZONE	FEATURES	COMMENT
loopback		none	none	none	none		
loopback.10		2100::1	default	vsys1	L3-Trust		
loopback.11		2100::2	default	vsys1	L3-Trust		
loopback.12		2100::3	default	vsys1	L3-Trust		
loopback.13		2100::4	default	vsys1	L3-Trust		
loopback.14		2100::5	default	vsys1	L3-Trust		
loopback.15		2100::6	default	vsys1	L3-Trust		
loopback.16		2100::7	default	vsys1	L3-Trust		
loopback.17		2100::8	default	vsys1	L3-Trust		
loopback.18		2100::9	default	vsys1	L3-Trust		
loopback.19		2100::a	default	vsys1	L3-Trust		

Tunnel configuration

```
'N_NET_IF_ETHERNET': 10,  # number of subinterfaces per each physical ethernet
'N_NET_IF_LOOPBACK': 10,
'N_NET_IF_TUNNEL': 10,

'CONFIG_VERSION': "9.1.0",
'LHOST': "localhost.localdomain",
'VSYS': "vsys1",

# 'DEFAULT_ZONE1': "L3-Trust",
# 'DEFAULT_ZONE2': "L3-Untrust",
'DEFAULT_ZONE1': "L3-trust",
'DEFAULT_ZONE2': "L3-untrust",
'DEFAULT_ZONE3': "VPN",

'DEFAULT_VR': "default",

# ------------------------------------------------------------------------------
#
# ethernet subinterfaces, loopback and tunnel interfaces
#

'IF_TUNNEL_NAME_i': 11,  # i, the initial index value
# 'IF_TUNNEL_IP': None,  # "",
# 'IF_TUNNEL_IP': "227.11.11.1/30",
'IF_TUNNEL_IP': "2227::1/126",
'IF_TUNNEL_VR': cf['DEFAULT_VR'],
'IF_TUNNEL_ZONE': cf['DEFAULT_ZONE3'],
```

Tunnel interface - Configuration pushed

INTERFACE	MANAGEMENT PROFILE	IP ADDRESS	VIRTUAL ROUTER	VIRTUAL SYSTEM	SECURITY ZONE	FEATURES	COMMENT
tunnel		none	none	none	none		
tunnel.1		none	default	vsys1	VPN		
tunnel.2		none	default	vsys1	VPN		
tunnel.11		2227::1/126	default	vsys1	VPN		
tunnel.12		2227::5/126	default	vsys1	VPN		
tunnel.13		2227::9/126	default	vsys1	VPN		
tunnel.14		2227::d/126	default	vsys1	VPN		
tunnel.15		2227::11/126	default	vsys1	VPN		
tunnel.16		2227::15/126	default	vsys1	VPN		
tunnel.17		2227::19/126	default	vsys1	VPN		
tunnel.18		2227::1d/126	default	vsys1	VPN		
tunnel.19		2227::21/126	default	vsys1	VPN		
tunnel.20		2227::25/126	default	vsys1	VPN		

Network > Zones

Security zone - configuration

```
'N_NET_ZONES': 10,

'ZONE_NAME': "Zone-{0:03d}",
'ZONE_NAME_i': 1,
'ZONE_TYPE': ["tap", "virtual-wire", "layer2", "layer3", "tunnel"],  # pick one here with its index value
'ZONE_UID': True,   # True or False, True to enable User-ID
```

	NAME	TYPE	INTERFACES / VIRTUAL SYSTEMS	ZONE PROTECTION PROFILE	PACKET BUFFER PROTECTION	LOG SETTING	User-ID			Device-ID		
							ENABLED	INCLUDED NETWORKS	EXCLUDED NETWORKS	ENABLED	INCLUDED NETWORKS	EXCLUDED NETWORKS
☐	Zone-001	layer3			☑		☑	any	none	☐	any	none
☐	Zone-002	layer3			☑		☑	any	none	☐	any	none
☐	Zone-003	layer3			☑		☑	any	none	☐	any	none
☐	Zone-004	layer3			☑		☑	any	none	☐	any	none
☐	Zone-005	layer3			☑		☑	any	none	☐	any	none
☐	Zone-006	layer3			☑		☑	any	none	☐	any	none
☐	Zone-007	layer3			☑		☑	any	none	☐	any	none
☐	Zone-008	layer3			☑		☑	any	none	☐	any	none
☐	Zone-009	layer3			☑		☑	any	none	☐	any	none
☐	Zone-010	layer3			☑		☑	any	none	☐	any	none

Zones can be with more config options
to perform in policy enforcement.

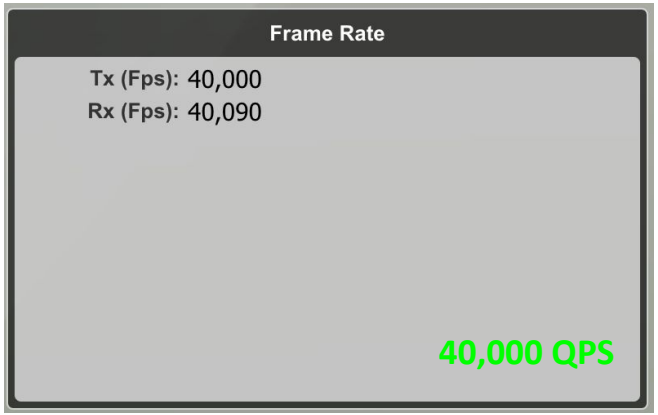
Network > DNS Proxy

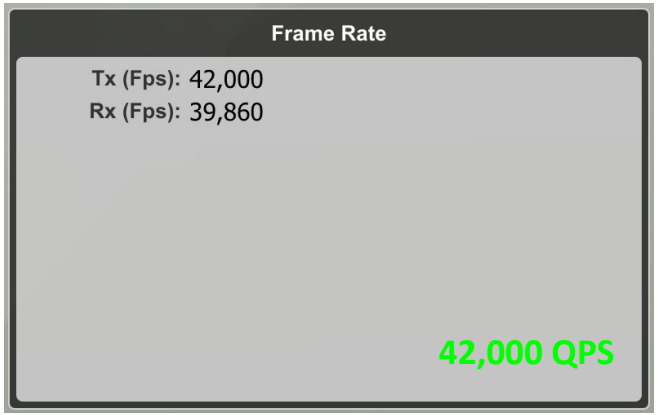
DNS Proxy

- Why ? Capacity testing
- Performance ? MPOV on DNS Proxy

  https://docs.google.com/presentation/d/1Pz-4uUFaLBCuh1VgyvNuEaoq50vmLUmtERAuEaMyj78/edit?usp=sharing

Higher QPS will cause frame rate unbalanced.



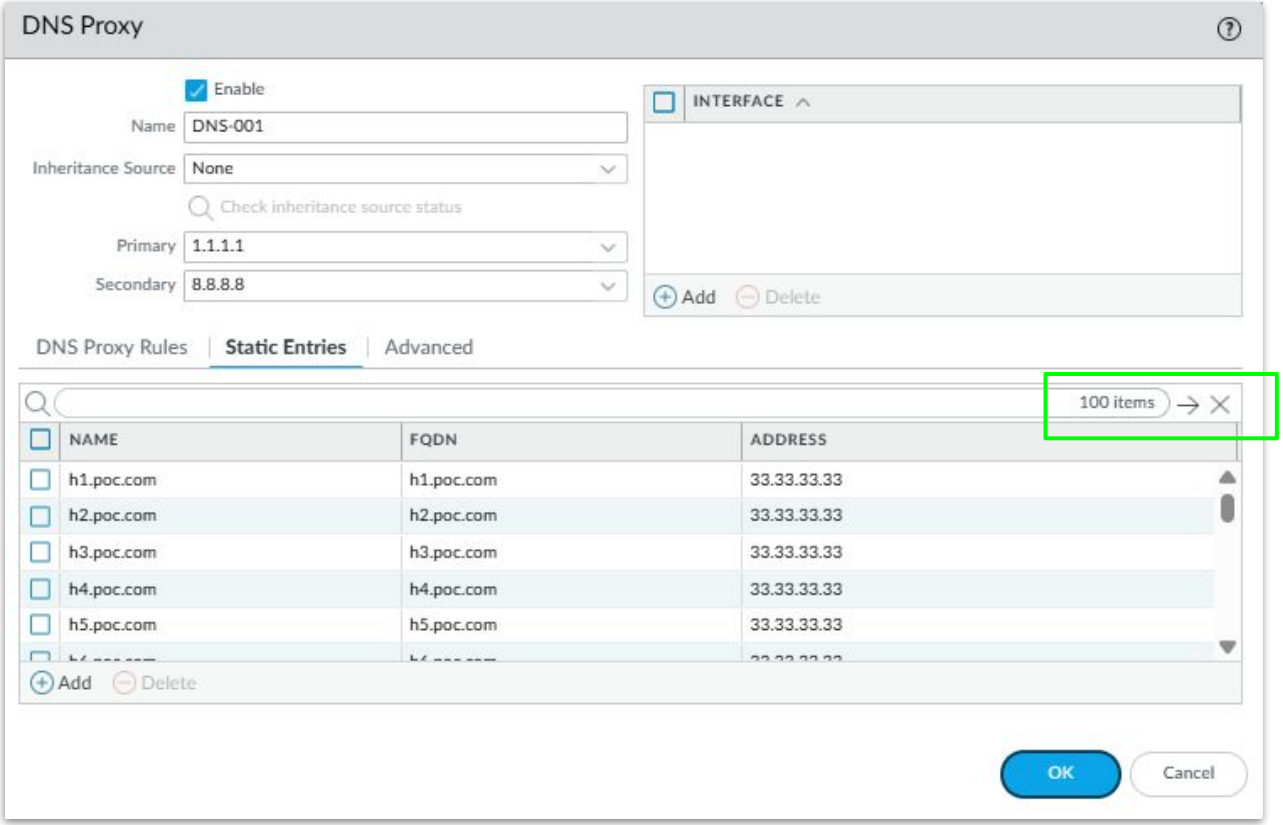


DNS Proxy - Script configuration

```
'N_NET_DNS_PROXY': 10,

'DNS_PROXY_NAME': "DNS-{0:03d}",
'DNS_PROXY_NAME_i': 1,                    # i, the initial index value for the names
'DNS_PROXY_PRIMARY': "1.1.1.1",           # primary DNS server IP
'DNS_PROXY_SECONDARY': "8.8.8.8",         # secondary DNS server IP
# 'DNS_PROXY_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],  # cannot be shared
'DNS_PROXY_INTERFACE_LIST': [],           # or [] for all with no interface assignment
'DNS_PROXY_STATIC_ENTRIES': 100,          # number of static entries
'DNS_PROXY_STATIC_FQDN': "h{0}.poc.com",  # FQDN of static entries
'DNS_PROXY_STATIC_ADDR': "33.33.33.33",   # IP of static entries
```

NAME	LOCATION	ENABLED	INTERFACES	PRIMARY DNS	SECONDARY DNS	DNS SERVER PROFILE	CACHE ENABLED	STATIC DNS COUNT
DNS-001		✓		1.1.1.1	8.8.8.8		✓	100
DNS-002		✓		1.1.1.1	8.8.8.8		✓	100
DNS-003		✓		1.1.1.1	8.8.8.8		✓	100
DNS-004		✓		1.1.1.1	8.8.8.8		✓	100
DNS-005		✓		1.1.1.1	8.8.8.8		✓	100
DNS-006		✓		1.1.1.1	8.8.8.8		✓	100
DNS-007		✓		1.1.1.1	8.8.8.8		✓	100
DNS-008		✓		1.1.1.1	8.8.8.8		✓	100
DNS-009		✓		1.1.1.1	8.8.8.8		✓	100
DNS-010		✓		1.1.1.1	8.8.8.8		✓	100

DNS Proxy - Configuration pushed



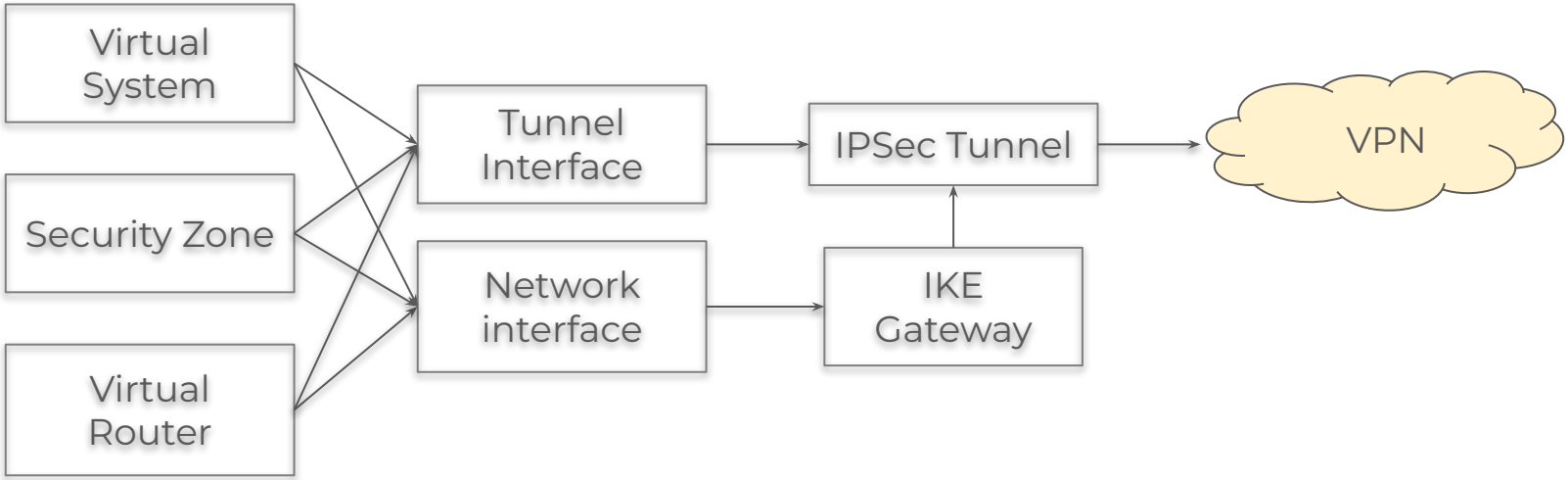
IKE Gateway / IPSec VPN

IPSec VPN Tunnels

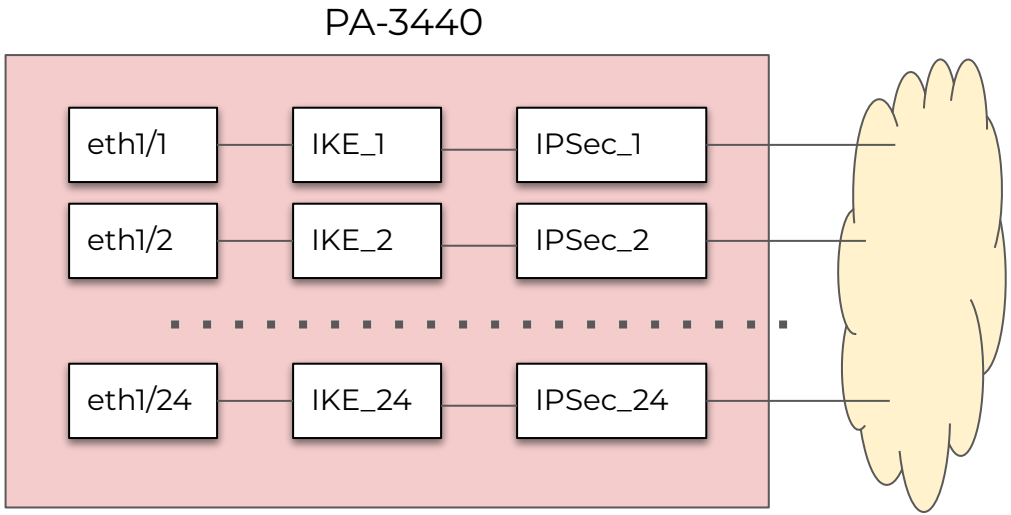
- Configuration is generated for device pair.
- Here PA1 and PA2 are specified in the config file conf/pa.py, and VPN configuration will be “mirrored” and pushed to both devices.

IPSec VPN - Configuration mixes

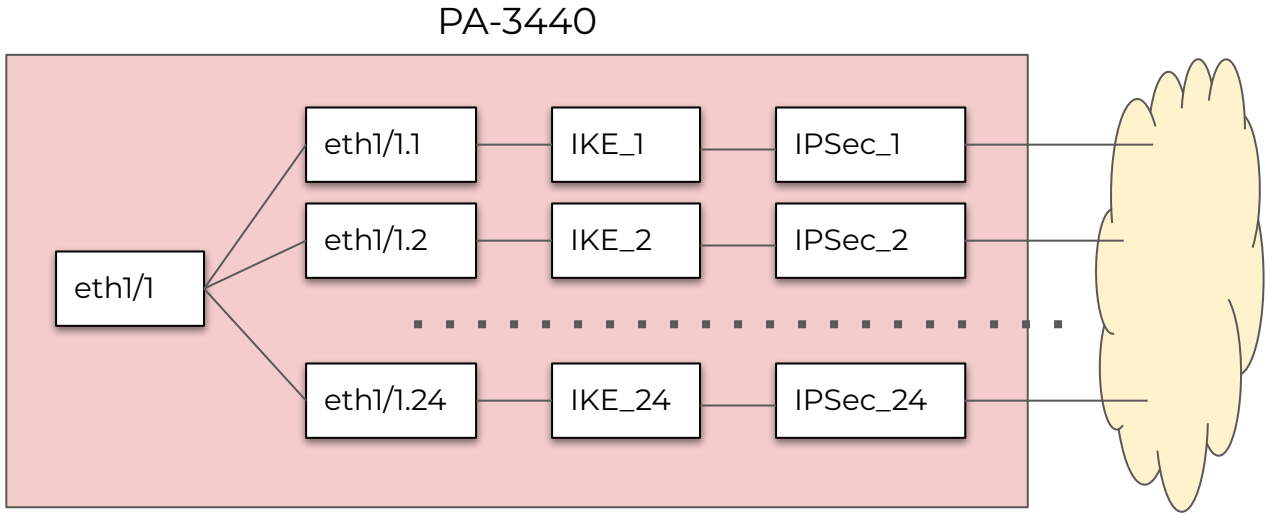
- IPSec VPN requires a number of configuration components to work together.
- There are a number of choices for each of these components
- The following diagrams represent their chain of dependency.



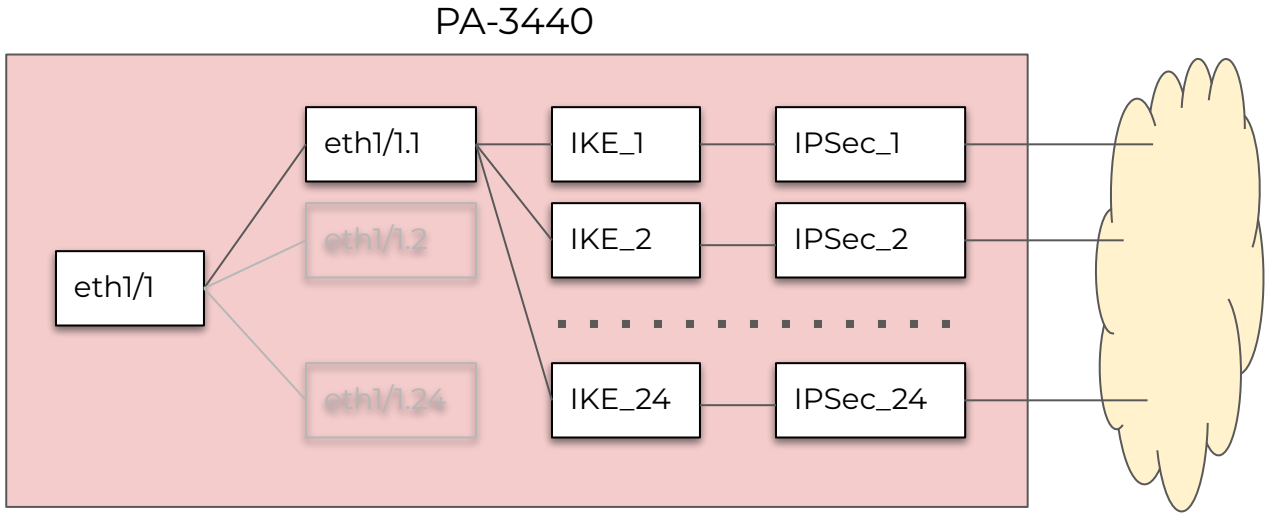
Typical settings (one-end, symmetric)



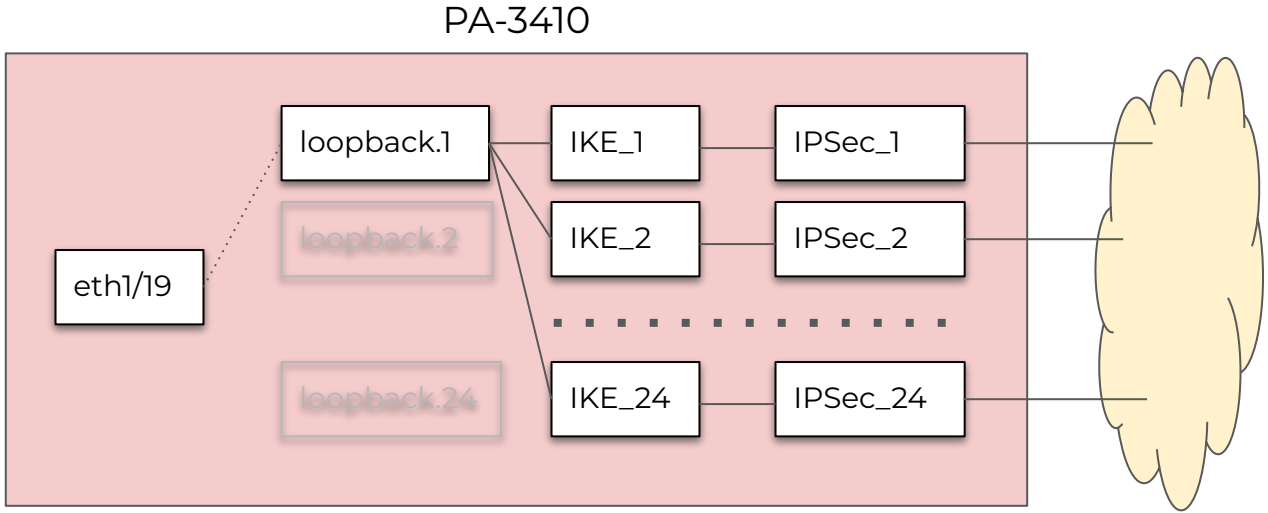
Subinterfaces each having an associated IKE gateway



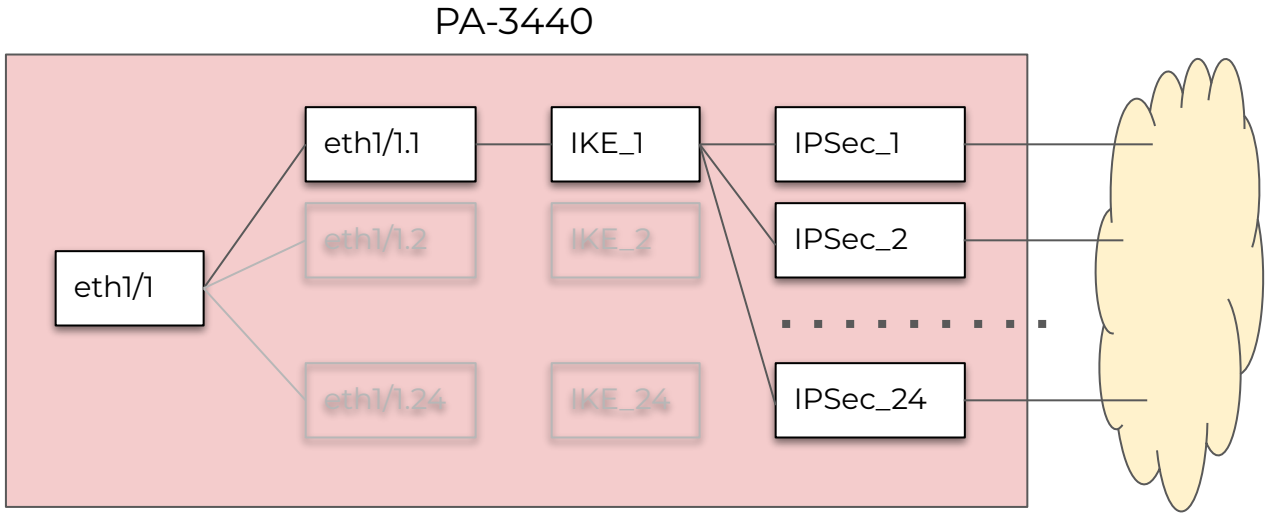
IKE gateways sharing the same (physical or sub-physical) interface



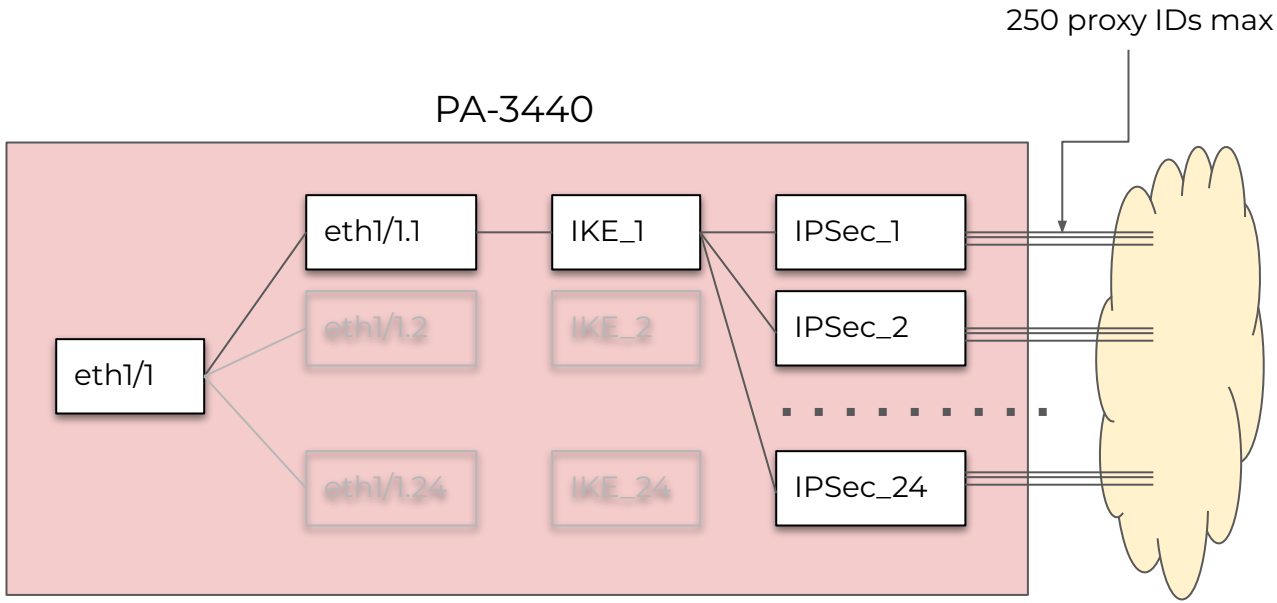
IKE gateways sharing the same loopback interface
POC16523, QoS on the physical interface to ignore the 3K tunnels



IPSec tunnel sharing the same IKE gateway



Proxy ID for scaling out number of tunnels



IKE Gateway - script configuration

```
# IPSec VPN
#
'N_NET_IKE': 32,
'N_NET_IPSEC': 32,

# ------------------------------------------------------------------------------
#
# VPN configuration: IKE gateways and IPSec tunnels
#

'IKE_NAME': "IKE_Gateway-{0}",
'IKE_NAME_i': 11,  # i, the initial index value
'IKE_VERSION': "ikev2-preferred",  # ikev1, ikev2 or ikev2-preferred
# 'IKE_INTERFACE_LIST': [("loopback.{0}", 1, 16],    # loopback interfaces
# 'IKE_INTERFACE_LIST': [("ethernet1/2", 1, 1)],     # ethernet/2
# 'IKE_INTERFACE_LIST': [("ethernet1/{0}", 1, 32)],  # ethernet1/1, .., ethernet1/32
'IKE_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],  # list of heterogeneous interfaces
'IKE_IP_LOCAL': "11.11.0.1",     # prefix specified separately
'IKE_IP_PEER': "11.11.0.2",      # prefix specified separately
'IKE_IP_PREFIX': "/30",          # required for mirrored config
'IKE_PRESHARED_KEY': "test123",  # -AQ==cojt0Pw//L6ToM8G41aOKFIWh7w=CVJ5/F84i6cL7ejjM15fRA==
'IKE_CRYPTO_PROFILE': "default",
```

IKE Gateway - script configuration

If the address family is IPv6, make sure the following are satisfied.

- All address fields such Destination and Next Hop, are IPv6.

```
'IKE_IP_LOCAL': "2011:11::1",      # prefix specified separately
'IKE_IP_PEER': "2011:11::2",       # prefix specified separately
'IKE_IP_PREFIX': "/126",           # required for mirrored config
```

- All interface fields support IPv6.

```
'IKE_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],
```

- The Interface IP's have to be consistent. You may check interface configuration.

```
'N_NET_IF_ETHERNET': 16,
'IF_ETHERNET_LIST': ["ethernet1/11", "ethernet1/12"],  # list of parent interfaces
'IF_ETHERNET_IP': "2211:11::1/126",

'IKE_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],
'IKE_IP_LOCAL': "2011:11::1",      # prefix specified separately
'IKE_IP_PREFIX': "/126",           # required for mirrored config
```

IKE Gateway - Configuration pushed

			Local Address		Peer ID		Local ID			IKE Advanced Options				
	NAME	PEER ADDRESS	INTERFACE	IP	ID	TYPE	ID	TYPE	VERSION	MODE	PASSIVE MODE	NAT TRAVERSAL	DPD	LIVENESS
	IKE_Gateway-11	11.11.0.2	ethernet1/11.1	11.11.0.1/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-12	11.11.0.6	ethernet1/11.2	11.11.0.5/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-13	11.11.0.10	ethernet1/11.3	11.11.0.9/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-14	11.11.0.14	ethernet1/11.4	11.11.0.13/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-15	11.11.0.18	ethernet1/11.5	11.11.0.17/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-16	11.11.0.22	ethernet1/11.6	11.11.0.21/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-17	11.11.0.26	ethernet1/11.7	11.11.0.25/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-18	11.11.0.30	ethernet1/11.8	11.11.0.29/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-19	11.11.0.34	ethernet1/11.9	11.11.0.33/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-20	11.11.0.38	ethernet1/11.10	11.11.0.37/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-21	11.11.0.42	ethernet1/11.11	11.11.0.41/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-22	11.11.0.46	ethernet1/11.12	11.11.0.45/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-23	11.11.0.50	ethernet1/11.13	11.11.0.49/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-24	11.11.0.54	ethernet1/11.14	11.11.0.53/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-25	11.11.0.58	ethernet1/11.15	11.11.0.57/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-26	11.11.0.62	ethernet1/11.16	11.11.0.61/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-27	11.11.0.66	ethernet1/12.1	11.11.0.65/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-28	11.11.0.70	ethernet1/12.2	11.11.0.69/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-29	11.11.0.74	ethernet1/12.3	11.11.0.73/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-30	11.11.0.78	ethernet1/12.4	11.11.0.77/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-31	11.11.0.82	ethernet1/12.5	11.11.0.81/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-32	11.11.0.86	ethernet1/12.6	11.11.0.85/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-33	11.11.0.90	ethernet1/12.7	11.11.0.89/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-34	11.11.0.94	ethernet1/12.8	11.11.0.93/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-35	11.11.0.98	ethernet1/12.9	11.11.0.97/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-36	11.11.0.102	ethernet1/12.10	11.11.0.101/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-37	11.11.0.106	ethernet1/12.11	11.11.0.105/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-38	11.11.0.110	ethernet1/12.12	11.11.0.109/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-39	11.11.0.114	ethernet1/12.13	11.11.0.113/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-40	11.11.0.118	ethernet1/12.14	11.11.0.117/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-41	11.11.0.122	ethernet1/12.15	11.11.0.121/30					ikev2-preferred	auto			enabled/default/default	default
	IKE_Gateway-42	11.11.0.126	ethernet1/12.16	11.11.0.125/30					ikev2-preferred	auto			enabled/default/default	default

IPSec Tunnel - script configuration

```
'DEFAULT_VR': "default",

'N_NET_IPSEC': 32,

# ------------------------------------------------------------------------------
#
# VPN configuration: IKE gateways and IPSec tunnels
#

'IPSEC_NAME': "IPSec_Tunnel-{0}",        # IPSec_Tunnel-$i
'IPSEC_NAME_i': 11,                      # initial index value for IPSec tunnel
'IPSEC_TUNNEL_INTERFACE': 'tunnel.{0}',  # Tunnel interface
'IPSEC_TUNNEL_INTERFACE_i': 11,
# 'IPSEC_IKE_GATEWAY': "IKE_Gateway",    # shared
'IPSEC_IKE_GATEWAY': "IKE_Gateway-{0}",  # IKE_Gateway-$i
'IPSEC_IKE_GATEWAY_i': 11,               # initial index value for IKE gateway
'IPSEC_CRYPTO_PROFILE': "default",
'IPSEC_REPLAY_PROTECTION': "no",
'IPSEC_PROXY_ID_ADD': False,  # True or False, True to include proxy IDs
'IPSEC_PROXY_ID_LIMIT': 250,  # max number of proxy IDs in an IPSec tunnel. Limit of a tunnel interface is 250
'IPSEC_PROXY_ID_NAME': "Proxy_ID-{0}.{1}",
'IPSEC_PROXY_ID_PROTOCOL': "any",
'IPSEC_IP_LOCAL': "1.1.0.1",
'IPSEC_IP_LOCAL_PREFIX': "/24",
'IPSEC_IP_REMOTE': "2.2.0.1",
'IPSEC_IP_REMOTE_PREFIX': "/24",
'IPSEC_ROUTE_ADD': True,  # True or False, whether routes should also be installed with IPSec tunnels
'IPSEC_ROUTE_NAME': "Tunnel_Route-{0}",
'IPSEC_ROUTE_i': 11,
'IPSEC_VR': cf['DEFAULT_VR'],
```

IPSec Tunnel - Configuration pushed

				IKE Gateway/Satellite				Tunnel Interface			
NAME	STATUS	TYPE	INTERFACE	LOCAL IP	PEER ADDRESS	STATUS	INTERFACE	VIRTUAL ROUTER	VIRTUAL SYSTEM	SECURITY ZONE
IPSec_Tunnel-11	Tunnel Info	Auto Key	ethernet1/11.1	11.11.0.1/30	11.11.0.2	IKE Info	tunnel.11	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-12	Tunnel Info	Auto Key	ethernet1/11.2	11.11.0.5/30	11.11.0.6	IKE Info	tunnel.12	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-13	Tunnel Info	Auto Key	ethernet1/11.3	11.11.0.9/30	11.11.0.10	IKE Info	tunnel.13	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-14	Tunnel Info	Auto Key	ethernet1/11.4	11.11.0.13/30	11.11.0.14	IKE Info	tunnel.14	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-15	Tunnel Info	Auto Key	ethernet1/11.5	11.11.0.17/30	11.11.0.18	IKE Info	tunnel.15	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-16	Tunnel Info	Auto Key	ethernet1/11.6	11.11.0.21/30	11.11.0.22	IKE Info	tunnel.16	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-17	Tunnel Info	Auto Key	ethernet1/11.7	11.11.0.25/30	11.11.0.26	IKE Info	tunnel.17	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-18	Tunnel Info	Auto Key	ethernet1/11.8	11.11.0.29/30	11.11.0.30	IKE Info	tunnel.18	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-19	Tunnel Info	Auto Key	ethernet1/11.9	11.11.0.33/30	11.11.0.34	IKE Info	tunnel.19	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-20	Tunnel Info	Auto Key	ethernet1/11.10	11.11.0.37/30	11.11.0.38	IKE Info	tunnel.20	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-21	Tunnel Info	Auto Key	ethernet1/11.11	11.11.0.41/30	11.11.0.42	IKE Info	tunnel.21	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-22	Tunnel Info	Auto Key	ethernet1/11.12	11.11.0.45/30	11.11.0.46	IKE Info	tunnel.22	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-23	Tunnel Info	Auto Key	ethernet1/11.13	11.11.0.49/30	11.11.0.50	IKE Info	tunnel.23	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-24	Tunnel Info	Auto Key	ethernet1/11.14	11.11.0.53/30	11.11.0.54	IKE Info	tunnel.24	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-25	Tunnel Info	Auto Key	ethernet1/11.15	11.11.0.57/30	11.11.0.58	IKE Info	tunnel.25	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-26	Tunnel Info	Auto Key	ethernet1/11.16	11.11.0.61/30	11.11.0.62	IKE Info	tunnel.26	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-27	Tunnel Info	Auto Key	ethernet1/12.1	11.11.0.65/30	11.11.0.66	IKE Info	tunnel.27	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-28	Tunnel Info	Auto Key	ethernet1/12.2	11.11.0.69/30	11.11.0.70	IKE Info	tunnel.28	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-29	Tunnel Info	Auto Key	ethernet1/12.3	11.11.0.73/30	11.11.0.74	IKE Info	tunnel.29	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-30	Tunnel Info	Auto Key	ethernet1/12.4	11.11.0.77/30	11.11.0.78	IKE Info	tunnel.30	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-31	Tunnel Info	Auto Key	ethernet1/12.5	11.11.0.81/30	11.11.0.82	IKE Info	tunnel.31	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-32	Tunnel Info	Auto Key	ethernet1/12.6	11.11.0.85/30	11.11.0.86	IKE Info	tunnel.32	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-33	Tunnel Info	Auto Key	ethernet1/12.7	11.11.0.89/30	11.11.0.90	IKE Info	tunnel.33	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-34	Tunnel Info	Auto Key	ethernet1/12.8	11.11.0.93/30	11.11.0.94	IKE Info	tunnel.34	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-35	Tunnel Info	Auto Key	ethernet1/12.9	11.11.0.97/30	11.11.0.98	IKE Info	tunnel.35	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-36	Tunnel Info	Auto Key	ethernet1/12.10	11.11.0.101/30	11.11.0.102	IKE Info	tunnel.36	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-37	Tunnel Info	Auto Key	ethernet1/12.11	11.11.0.105/30	11.11.0.106	IKE Info	tunnel.37	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-38	Tunnel Info	Auto Key	ethernet1/12.12	11.11.0.109/30	11.11.0.110	IKE Info	tunnel.38	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-39	Tunnel Info	Auto Key	ethernet1/12.13	11.11.0.113/30	11.11.0.114	IKE Info	tunnel.39	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-40	Tunnel Info	Auto Key	ethernet1/12.14	11.11.0.117/30	11.11.0.118	IKE Info	tunnel.40	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-41	Tunnel Info	Auto Key	ethernet1/12.15	11.11.0.121/30	11.11.0.122	IKE Info	tunnel.41	default (Show Routes)	vsys1	VPN
IPSec_Tunnel-42	Tunnel Info	Auto Key	ethernet1/12.16	11.11.0.125/30	11.11.0.126	IKE Info	tunnel.42	default (Show Routes)	vsys1	VPN

Objects > Addresses
Objects > Address Groups
Objects > Services
Objects > Service Groups
Objects > Custom URL Category

Addresses - Script configuration

```
# Objects
#
'N_OBJ_ADDRESS': 20,
'N_OBJ_ADDRESS_GROUP': 5,

# ------------------------------------------------------------------------------
#
# Objects: addresses/groups, services/groups, URL categories
#

'ADDR_NAME': "Address-{0:03d}",
'ADDR_NAME_i': 1,
# --- IP Netmask
'ADDR_TYPE': "ip-netmask",
# 'ADDR_ADDRESS': "10.0.0.0/24",
'ADDR_ADDRESS': "2401:b200:2000::/64",
# --- IP Range
# 'ADDR_TYPE': "ip-range",
# 'ADDR_RANGE_SIZE': 10,
# 'ADDR_ADDRESS': "192.168.0.8/24",
# 'ADDR_ADDRESS': "2401:b200:2000::1/96",
# --- FQDN
# 'ADDR_TYPE': "fqdn",
# 'ADDR_ADDRESS': "w{0}.poc.local",
'ADDR_SHARED': False,
```

Addresses - Configuration pushed

```
# --- IP Netmask
'ADDR_TYPE': "ip-netmask",
'ADDR_ADDRESS': "2401:b200:2000::/64",
```

NAME	LOCATION	TYPE	ADDRESS
Address-001		IP Netmask	2401:b200:2000::/64
Address-002		IP Netmask	2401:b200:2000:1::/64
Address-003		IP Netmask	2401:b200:2000:2::/64
Address-004		IP Netmask	2401:b200:2000:3::/64
Address-005		IP Netmask	2401:b200:2000:4::/64
Address-006		IP Netmask	2401:b200:2000:5::/64
Address-007		IP Netmask	2401:b200:2000:6::/64
Address-008		IP Netmask	2401:b200:2000:7::/64
Address-009		IP Netmask	2401:b200:2000:8::/64
Address-010		IP Netmask	2401:b200:2000:9::/64
Address-011		IP Netmask	2401:b200:2000:a::/64
Address-012		IP Netmask	2401:b200:2000:b::/64
Address-013		IP Netmask	2401:b200:2000:c::/64
Address-014		IP Netmask	2401:b200:2000:d::/64
Address-015		IP Netmask	2401:b200:2000:e::/64
Address-016		IP Netmask	2401:b200:2000:f::/64
Address-017		IP Netmask	2401:b200:2000:10::/64
Address-018		IP Netmask	2401:b200:2000:11::/64
Address-019		IP Netmask	2401:b200:2000:12::/64
Address-020		IP Netmask	2401:b200:2000:13::/64

Addresses - Configuration pushed

```
# --- IP Range
'ADDR_TYPE': "ip-range",
'ADDR_RANGE_SIZE': 10,
'ADDR_ADDRESS': "192.168.0.8/24",
```

NAME	LOCATION	TYPE	ADDRESS
Address-001		IP Range	192.168.0.8-192.168.0.17
Address-002		IP Range	192.168.1.8-192.168.1.17
Address-003		IP Range	192.168.2.8-192.168.2.17
Address-004		IP Range	192.168.3.8-192.168.3.17
Address-005		IP Range	192.168.4.8-192.168.4.17
Address-006		IP Range	192.168.5.8-192.168.5.17
Address-007		IP Range	192.168.6.8-192.168.6.17
Address-008		IP Range	192.168.7.8-192.168.7.17
Address-009		IP Range	192.168.8.8-192.168.8.17
Address-010		IP Range	192.168.9.8-192.168.9.17
Address-011		IP Range	192.168.10.8-192.168.10.17
Address-012		IP Range	192.168.11.8-192.168.11.17
Address-013		IP Range	192.168.12.8-192.168.12.17
Address-014		IP Range	192.168.13.8-192.168.13.17
Address-015		IP Range	192.168.14.8-192.168.14.17
Address-016		IP Range	192.168.15.8-192.168.15.17
Address-017		IP Range	192.168.16.8-192.168.16.17
Address-018		IP Range	192.168.17.8-192.168.17.17
Address-019		IP Range	192.168.18.8-192.168.18.17
Address-020		IP Range	192.168.19.8-192.168.19.17

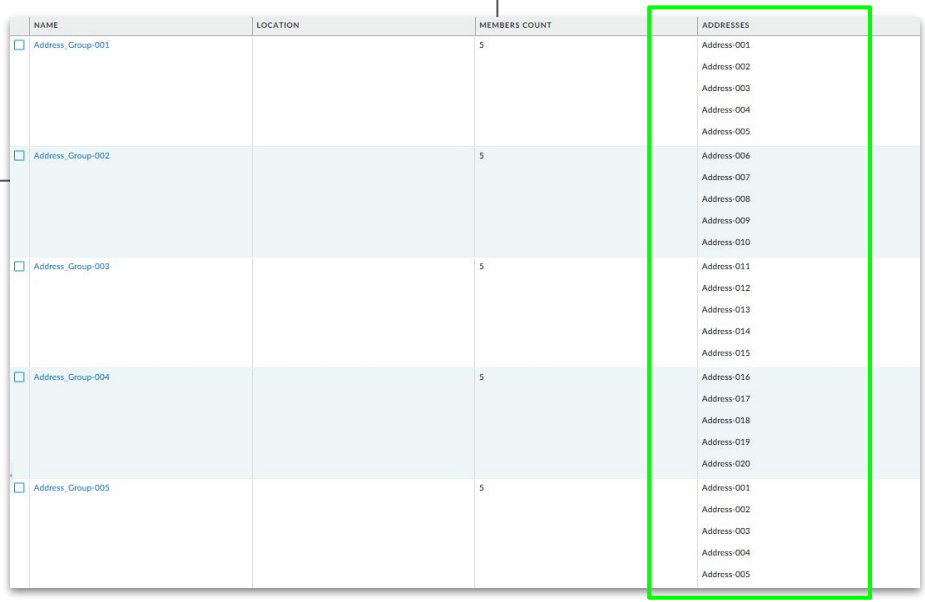
Addresses - Configuration pushed

```
# --- FQDN
'ADDR_TYPE': "fqdn",
'ADDR_ADDRESS': "w{0}.poc.local",
```

NAME	LOCATION	TYPE	ADDRESS
Address-001		FQDN	w1.poc.local
Address-002		FQDN	w2.poc.local
Address-003		FQDN	w3.poc.local
Address-004		FQDN	w4.poc.local
Address-005		FQDN	w5.poc.local
Address-006		FQDN	w6.poc.local
Address-007		FQDN	w7.poc.local
Address-008		FQDN	w8.poc.local
Address-009		FQDN	w9.poc.local
Address-010		FQDN	w10.poc.local
Address-011		FQDN	w11.poc.local
Address-012		FQDN	w12.poc.local
Address-013		FQDN	w13.poc.local
Address-014		FQDN	w14.poc.local
Address-015		FQDN	w15.poc.local
Address-016		FQDN	w16.poc.local
Address-017		FQDN	w17.poc.local
Address-018		FQDN	w18.poc.local
Address-019		FQDN	w19.poc.local
Address-020		FQDN	w20.poc.local

Address Groups - Script configuration

```
# Objects
#
'N_OBJ_ADDRESS': 20,
'N_OBJ_ADDRESS_GROUP': 5,

# ------------------------------------------------------------------------------
#
# Objects: addresses/groups, services/groups, URL categories
#

'ADDR_GROUP_NAME': "Address_Group-{0:03d}",
'ADDR_GROUP_NAME_i': 1,
'ADDR_GROUP_MEMBER_COUNT': 5,
```

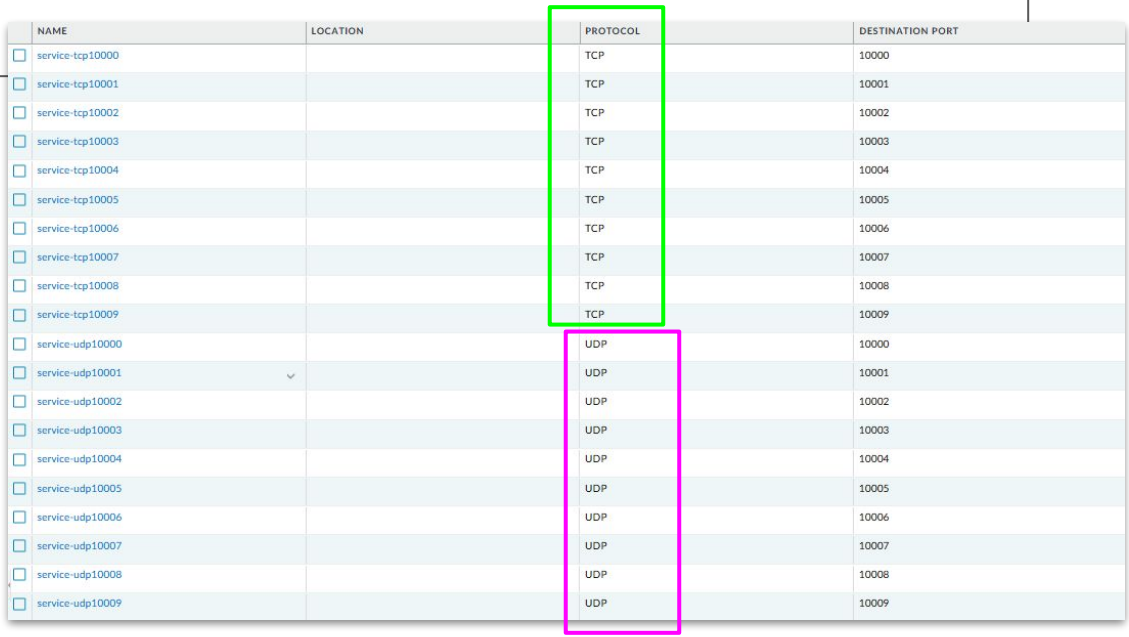


NAME	LOCATION	MEMBERS COUNT	ADDRESSES
Address_Group-001		5	Address-001
			Address-002
			Address-003
			Address-004
			Address-005
Address_Group-002		5	Address-006
			Address-007
			Address-008
			Address-009
			Address-010
Address_Group-003		5	Address-011
			Address-012
			Address-013
			Address-014
			Address-015
Address_Group-004		5	Address-016
			Address-017
			Address-018
			Address-019
			Address-020
Address_Group-005		5	Address-001
			Address-002
			Address-003
			Address-004
			Address-005

- Addresses previously configured will be assigned to address groups.
- Each of the addresses may be assigned more than once. They will be rotated when assignment has reached the end of the address list.

Services - Script configuration

```
'N_OBJ_SERVICE': 10,
'N_OBJ_SERVICE_GROUP': 5,

'SERVICE_NAME': "service-{0}{1}",
'SERVICE_PROTOCOL': "both",   # "tcp" or "udp" or "both"
'SERVICE_PORT_DST': 10000,    # initial dst port number
'SERVICE_PORT_SRC': 0,        # non-zero value will make source port grow with the destination port
```

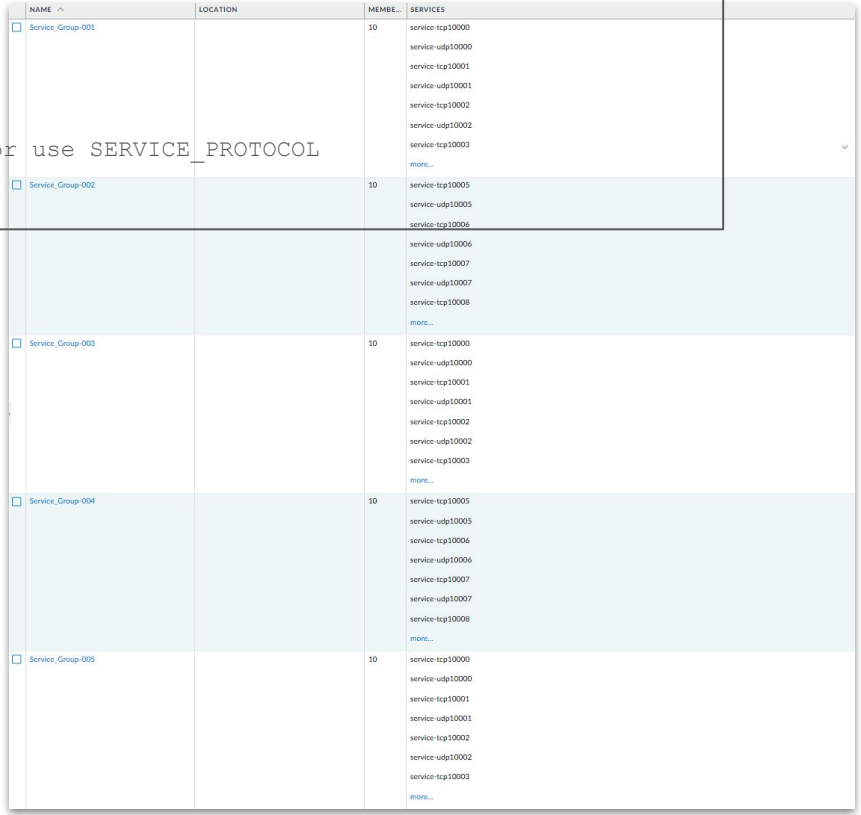


NAME	LOCATION	PROTOCOL	DESTINATION PORT
service-tcp10000		TCP	10000
service-tcp10001		TCP	10001
service-tcp10002		TCP	10002
service-tcp10003		TCP	10003
service-tcp10004		TCP	10004
service-tcp10005		TCP	10005
service-tcp10006		TCP	10006
service-tcp10007		TCP	10007
service-tcp10008		TCP	10008
service-tcp10009		TCP	10009
service-udp10000		UDP	10000
service-udp10001		UDP	10001
service-udp10002		UDP	10002
service-udp10003		UDP	10003
service-udp10004		UDP	10004
service-udp10005		UDP	10005
service-udp10006		UDP	10006
service-udp10007		UDP	10007
service-udp10008		UDP	10008
service-udp10009		UDP	10009

Services Groups - Script configuration

```
'N_OBJ_SERVICE': 10,
'N_OBJ_SERVICE_GROUP': 5,

'SERVICE_GROUP_NAME': "Service_Group-{0:03d}",
'SERVICE_GROUP_NAME_i': 1,
'SERVICE_GROUP_PROTOCOL': "",  # "tcp", "udp" or "both", or use SERVICE_PROTOCOL
'SERVICE_GROUP_MEMBER_COUNT': 5,
```

- Services previously configured will be assigned to Service Groups.
- Each of the services may be assigned more than once. They will be rotated when assignment has reached the end of the address list.



URL Categories - Script configuration

```
# Total custom URL entries = N_OBJ_URL_CATS x N_OBJ_URL_ENTRIES
#
'N_OBJ_URL_CATS': 10,
'N_OBJ_URL_ENTRIES': 5,

'URL_CAT_NAME': "URL_Category-{0:03d}",
'URL_CAT_NAME_i': 1,          # i, the initial index value of categories
'URL_ENTRY': "w{0}.d{1}.com",
'URL_ENTRY_j': 1,             # j, the initial index value of entries
'URL_TYPE': "URL List",
```

Custom URL Category - Configuration pushed

- There is no need to assign any existing objects to the categories.
- Instead each category is mapped to a list of URL's of specific patterns.
- One suggestion is that URL's may be retrieved from a text file, but we will need a use case to justify this effort.

NAME	LOCATION	TYPE	MATCH
URL_Category-001		URL List	w1.d1.com
			w2.d1.com
			w3.d1.com
			w4.d1.com
			w5.d1.com
URL_Category-002		URL List	w1.d2.com
			w2.d2.com
			w3.d2.com
			w4.d2.com
			w5.d2.com
URL_Category-003		URL List	w1.d3.com
			w2.d3.com
			w3.d3.com
			w4.d3.com
			w5.d3.com
URL_Category-004		URL List	w1.d4.com
			w2.d4.com
			w3.d4.com
			w4.d4.com
			w5.d4.com
URL_Category-005		URL List	w1.d5.com
			w2.d5.com
			w3.d5.com
			w4.d5.com
			w5.d5.com
URL_Category-006		URL List	w1.d6.com
			w2.d6.com
			w3.d6.com
			w4.d6.com
			w5.d6.com
URL_Category-007		URL List	w1.d7.com
			w2.d7.com
			w3.d7.com
			w4.d7.com
			w5.d7.com
URL_Category-008		URL List	w1.d8.com
			w2.d8.com
			w3.d8.com
			w4.d8.com
			w5.d8.com
URL_Category-009		URL List	w1.d9.com
			w2.d9.com
			w3.d9.com
			w4.d9.com
			w5.d9.com
URL_Category-010		URL List	w1.d10.com
			w2.d10.com
			w3.d10.com
			w4.d10.com
			w5.d10.com

Policies > Security
Policies > NAT
Policies > PBF

Security Policy - Script configuration

```
# policy rules
#
'N_RULES_SEC': 10,
'N_RULES_NAT': 10,
'N_RULES_PBF': 10,

# ------------------------------------------------------------------------------
#
# Policy rules: security, NAT, and PBF rules
#

'SEC_RULEBASE': "pre-rulebase",  # pre-rulebase or post-rulebase for Panorama
'SEC_NAME': "Rule-{0:03d}",  # Rule-1, Rule-2, etc.
'SEC_NAME_i': 1,  # Rule-1, Rule-2, etc.
'SEC_SRC_ZONE': cf['DEFAULT_ZONE1'],
'SEC_DST_ZONE': cf['DEFAULT_ZONE2'],
'SEC_SOURCE': "1.1.0.0/24",
'SEC_DESTINATION': "2.2.0.0/24",
'SEC_SERVICE': "any",
# 'SEC_SERVICE': "applicaton-default",
'SEC_ACTION': "deny",
```

Security Policy Rules - Configuration pushed

- Source and Destination are incremented in parallel per their network prefixes. For example, if source is 1.1.0.0/24, the source used in the next rule will become 1.1.1.0/24.
- IPv6 is supported.
- Some fields, such as User and Device, can only be static. As of this writing, they will not change over iterations.

				Source				Destination							
	NAME	TAGS	TYPE	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATION	SERVICE	ACTION	PROFILE	OPTIONS
19	Rule-001	none	universal	L3-Trust	1.1.0.0/24	any	any	L3-Untrust	2.2.0.0/24	any	any	any	Deny	none	
20	Rule-002	none	universal	L3-Trust	1.1.1.0/24	any	any	L3-Untrust	2.2.1.0/24	any	any	any	Deny	none	
21	Rule-003	none	universal	L3-Trust	1.1.2.0/24	any	any	L3-Untrust	2.2.2.0/24	any	any	any	Deny	none	
22	Rule-004	none	universal	L3-Trust	1.1.3.0/24	any	any	L3-Untrust	2.2.3.0/24	any	any	any	Deny	none	
23	Rule-005	none	universal	L3-Trust	1.1.4.0/24	any	any	L3-Untrust	2.2.4.0/24	any	any	any	Deny	none	
24	Rule-006	none	universal	L3-Trust	1.1.5.0/24	any	any	L3-Untrust	2.2.5.0/24	any	any	any	Deny	none	
25	Rule-007	none	universal	L3-Trust	1.1.6.0/24	any	any	L3-Untrust	2.2.6.0/24	any	any	any	Deny	none	
26	Rule-008	none	universal	L3-Trust	1.1.7.0/24	any	any	L3-Untrust	2.2.7.0/24	any	any	any	Deny	none	
27	Rule-009	none	universal	L3-Trust	1.1.8.0/24	any	any	L3-Untrust	2.2.8.0/24	any	any	any	Deny	none	
28	Rule-010	none	universal	L3-Trust	1.1.9.0/24	any	any	L3-Untrust	2.2.9.0/24	any	any	any	Deny	none	

NAT Policy - Script configuration

```
# policy rules
#
'N_RULES_SEC': 10,
'N_RULES_NAT': 10,
'N_RULES_PBF': 10,

# ------------------------------------------------------------------------------
#
# Policy rules: security, NAT, and PBF rules
#

'NAT_RULEBASE': "pre-rulebase",  # pre-rulebase or post-rulebase for Panorama
'NAT_NAME': "NAT_Rule-{0}",
'NAT_NAME_i': 1,
'NAT_SRC_ZONE': cf['DEFAULT_ZONE1'],
'NAT_DST_ZONE': cf['DEFAULT_ZONE2'],
'NAT_SOURCE': "1.1.0.0/24",
'NAT_DESTINATION': "2.2.0.0/24",  # 2.2.$j.0/24, $i is hidden
'NAT_SERVICE': "any",
```

NAT Policy Rules - Configuration pushed

- Source and Destination are incremented in parallel per their network prefixes. For example, if source is 1.1.0.0/24, the source used in the next rule will become 1.1.1.0/24.
- IPv6 is supported.
- NAT translation is not defined yet but can be hardcoded.

			Original Packet						Translated Packet	
	NAME	TAGS	SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION
10	NAT_Rule-1	none	L3-Trust	L3-Untrust	any	1.1.0.0/24	2.2.0.0/24	any	none	none
11	NAT_Rule-2	none	L3-Trust	L3-Untrust	any	1.1.1.0/24	2.2.1.0/24	any	none	none
12	NAT_Rule-3	none	L3-Trust	L3-Untrust	any	1.1.2.0/24	2.2.2.0/24	any	none	none
13	NAT_Rule-4	none	L3-Trust	L3-Untrust	any	1.1.3.0/24	2.2.3.0/24	any	none	none
14	NAT_Rule-5	none	L3-Trust	L3-Untrust	any	1.1.4.0/24	2.2.4.0/24	any	none	none
15	NAT_Rule-6	none	L3-Trust	L3-Untrust	any	1.1.5.0/24	2.2.5.0/24	any	none	none
16	NAT_Rule-7	none	L3-Trust	L3-Untrust	any	1.1.6.0/24	2.2.6.0/24	any	none	none
17	NAT_Rule-8	none	L3-Trust	L3-Untrust	any	1.1.7.0/24	2.2.7.0/24	any	none	none
18	NAT_Rule-9	none	L3-Trust	L3-Untrust	any	1.1.8.0/24	2.2.8.0/24	any	none	none
19	NAT_Rule-10	none	L3-Trust	L3-Untrust	any	1.1.9.0/24	2.2.9.0/24	any	none	none

PBF Policy - Script configuration

```
# policy rules
#
'N_RULES_SEC': 10,
'N_RULES_NAT': 10,
'N_RULES_PBF': 10,

# ------------------------------------------------------------------------------
#
# Policy rules: security, NAT, and PBF rules
#

'PBF_RULEBASE': "pre-rulebase",  # pre-rulebase or post-rulesbase for Panorama
'PBF_NAME': "PBF_Rule-{0:03d}",
'PBF_NAME_i': 1,
'PBF_SRC_ZONE': cf['DEFAULT_ZONE1'],
'PBF_SOURCE': "3.3.0.0/24",
'PBF_DESTINATION': "4.4.0.0/24",
'PBF_SERVICE': "any",
'PBF_ACTION': ["forward", "discard", "no-pbf"][0],  # pick one as the action of all rules
'PBF_EGRESS_INTERFACE': "ethernet1/11",
'PBF_NEXTHOP': "100.1.0.100",
```

PBF Policy Rules - Configuration pushed

- Source and Destination are incremented in parallel per their network prefixes. For example, if source is 1.1.0.0/24, the source used in the next rule will become 1.1.1.0/24.
- IPv6 is supported.
- Forwarding action is always Forward yet but can be extended.

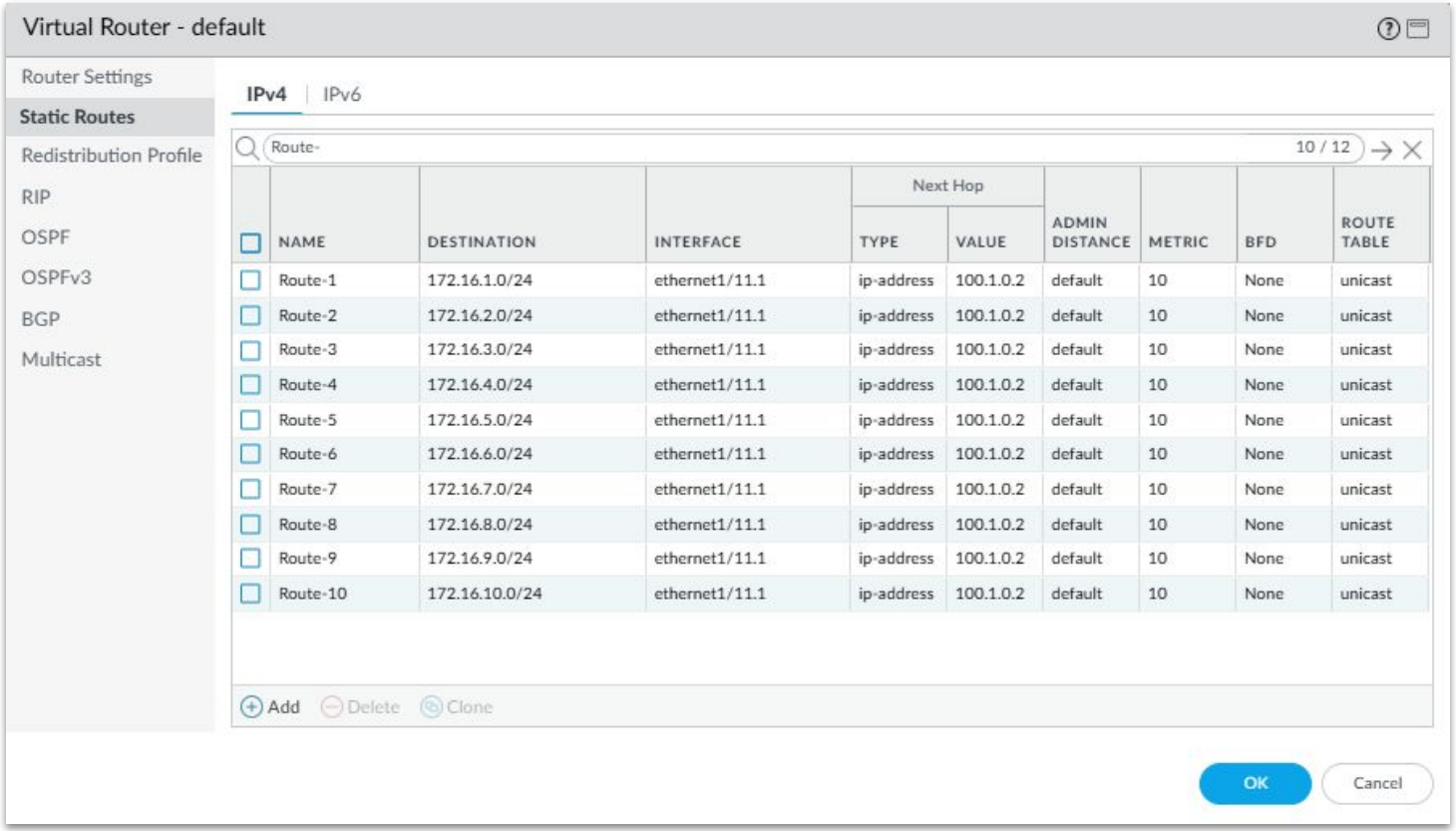
			Source			Destination				Forwarding			Monitoring			
	NAME	TAGS	ZONE/INTERFACE	ADDRESS	USER	ADDRESS	APPLICATION	SERVICE	ACTION	EGRESS I/F	NEXT HOP	ENFORCE SYMMETRIC RETURN	PROFILE	TARGET	DISABLE IF UNREACHABLE	SCHEDULE
1	PBF_Rule-001	none	L3-Trust	3.3.0.0/24	any	4.4.0.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
2	PBF_Rule-002	none	L3-Trust	3.3.1.0/24	any	4.4.1.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
3	PBF_Rule-003	none	L3-Trust	3.3.2.0/24	any	4.4.2.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
4	PBF_Rule-004	none	L3-Trust	3.3.3.0/24	any	4.4.3.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
5	PBF_Rule-005	none	L3-Trust	3.3.4.0/24	any	4.4.4.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
6	PBF_Rule-006	none	L3-Trust	3.3.5.0/24	any	4.4.5.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
7	PBF_Rule-007	none	L3-Trust	3.3.6.0/24	any	4.4.6.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
8	PBF_Rule-008	none	L3-Trust	3.3.7.0/24	any	4.4.7.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
9	PBF_Rule-009	none	L3-Trust	3.3.8.0/24	any	4.4.8.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none
10	PBF_Rule-010	none	L3-Trust	3.3.9.0/24	any	4.4.9.0/24	any	any	forward	ethernet1/11	100.1.0.100	false	none	none	false	none

Network > VR > Static Routes
Network > VR > BGP peer groups x peers

Virtual Router > Static Routes - Script configuration

```
# virtual router config
#
'N_VR_STATIC': 10,
'N_VR_BGP_PEER_GROUPS': 10,
'N_VR_BGP_PEERS_PER_GROUP': 10,

# ------------------------------------------------------------------------------
#
# VR configuration: static routes, BGP groups
#

'VR_STATIC_VR': cf['DEFAULT_VR'],
'VR_STATIC_NAME': "Route-{0}",
'VR_STATIC_NAME_i': 1,
'VR_STATIC_DESTINATION': "172.16.1.0/24",
'VR_STATIC_INTERFACE': "ethernet1/11.1",
'VR_STATIC_NEXTHOP': "100.1.0.2",
```

Virtual Router > Static Routes - Configuration pushed



NAME	DESTINATION	INTERFACE	Next Hop TYPE	Next Hop VALUE	ADMIN DISTANCE	METRIC	BFD	ROUTE TABLE
Route-1	172.16.1.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-2	172.16.2.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-3	172.16.3.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-4	172.16.4.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-5	172.16.5.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-6	172.16.6.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-7	172.16.7.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-8	172.16.8.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-9	172.16.9.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast
Route-10	172.16.10.0/24	ethernet1/11.1	ip-address	100.1.0.2	default	10	None	unicast

Virtual Router > Static Routes - Script configuration

```
# virtual router config
#
'N_VR_STATIC': 10,
'N_VR_BGP_PEER_GROUPS': 10,
'N_VR_BGP_PEERS_PER_GROUP': 10,

# ------------------------------------------------------------------------------
#
# VR configuration: static routes, BGP groups
#

'VR_STATIC_VR': cf['DEFAULT_VR'],
'VR_STATIC_NAME': "Route-{0}",
'VR_STATIC_NAME_i': 1,
'VR_STATIC_DESTINATION': "2172:16::0/120",
'VR_STATIC_INTERFACE': "ethernet1/11.1",
'VR_STATIC_NEXTHOP': "2100::2",
```

Virtual Router > Static Routes - Script configuration

If the address family is IPv6, make sure the following are satisfied.

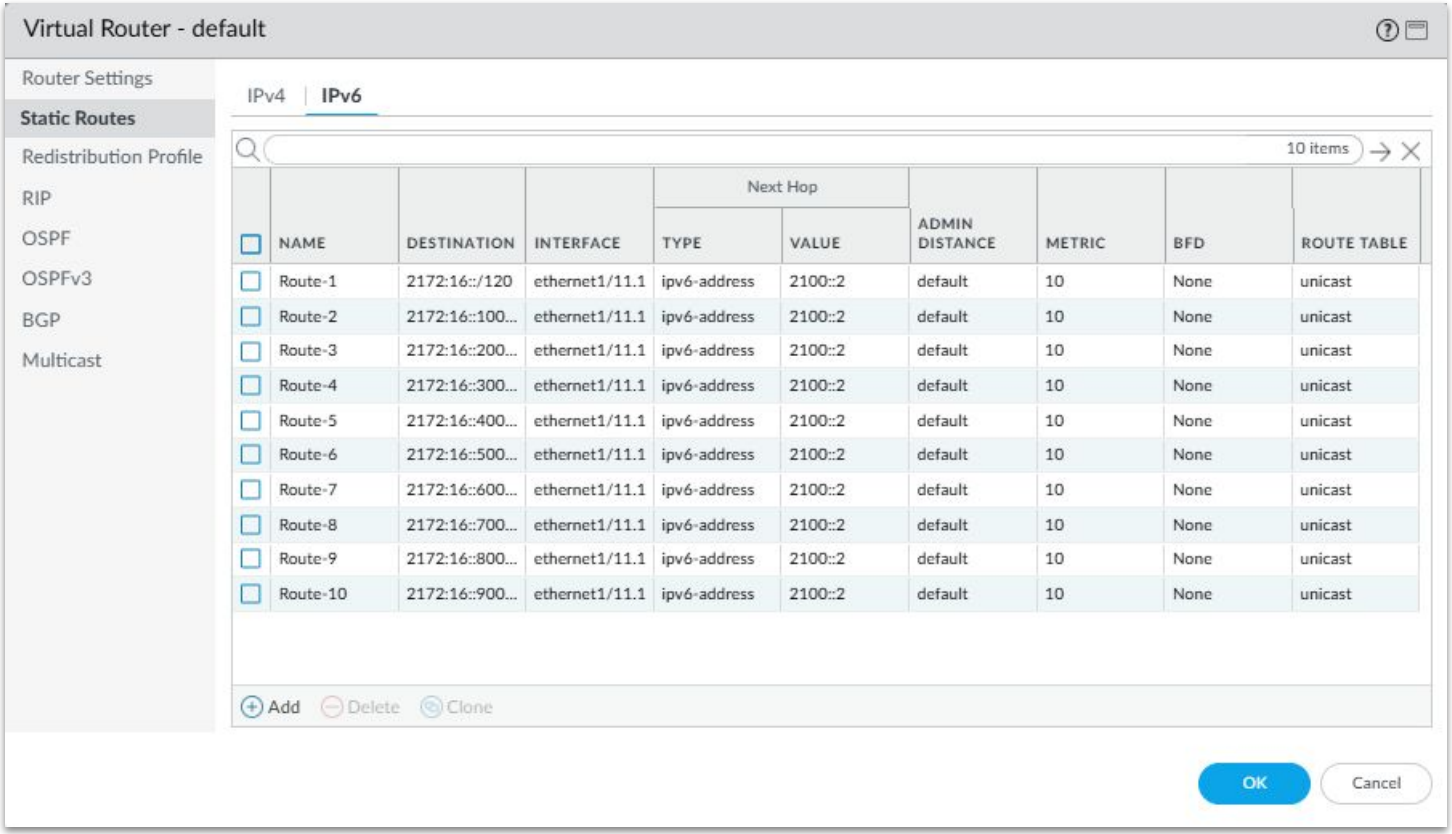
- All address fields such Destination and Next Hop, are IPv6.

```
'VR_STATIC_DESTINATION': "2172:16::0/120",
'VR_STATIC_NEXTHOP': "2100::2",
```

- All interface fields support IPv6.

```
'VR_STATIC_INTERFACE': "ethernet1/11.1",
```

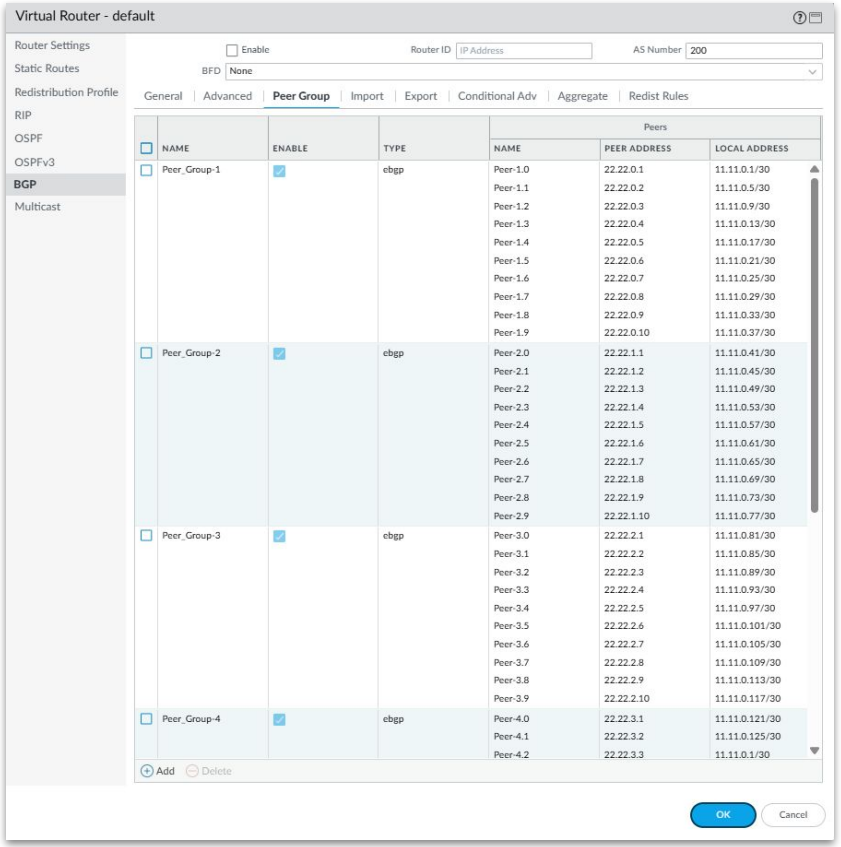
Virtual Router > Static Routes - Configuration pushed

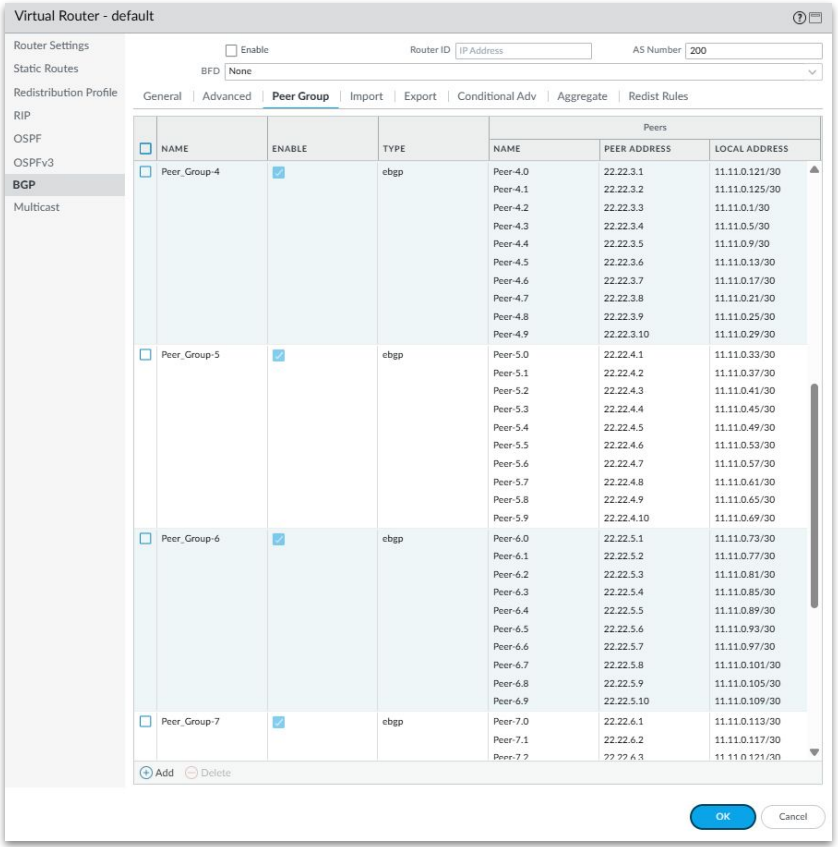


Virtual Router > BGP Peer Groups - Script configuration

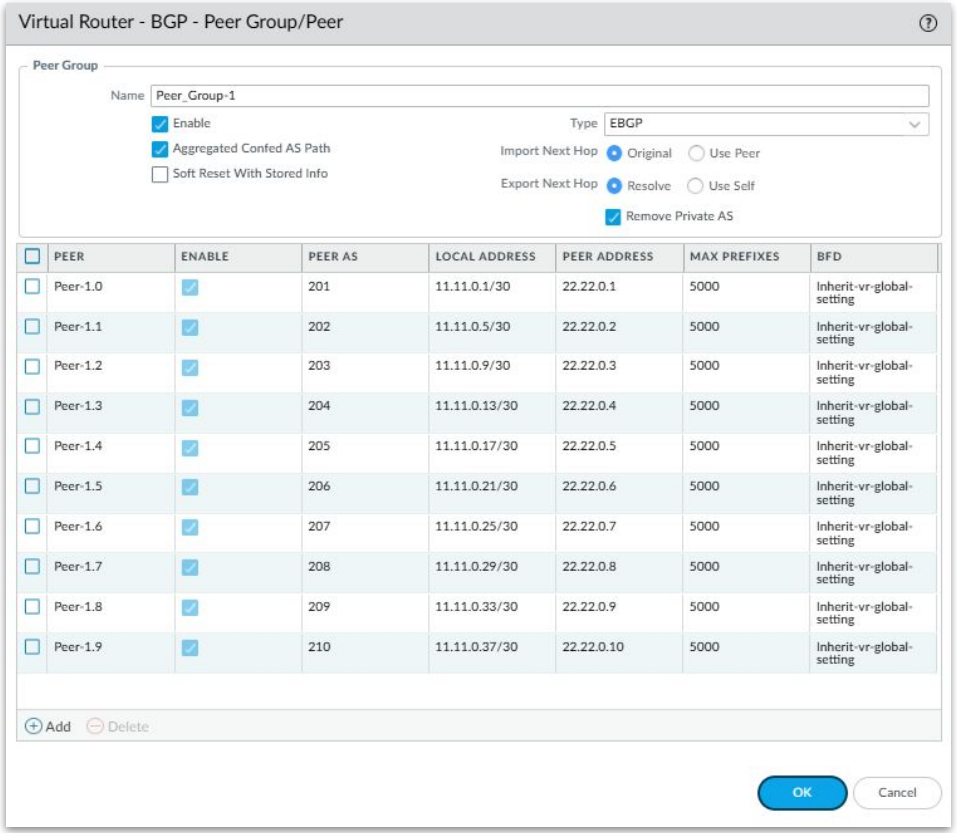
```
# virtual router config
#
'N_VR_STATIC': 10,
'N_VR_BGP_PEER_GROUPS': 10,
'N_VR_BGP_PEERS_PER_GROUP': 10,

# ------------------------------------------------------------------------------
#
# VR configuration: static routes, BGP groups
#

'VR_BGP_VR': cf['DEFAULT_VR'],
'VR_BGP_PEER_GROUP_NAME': "Peer_Group-{0}",
'VR_BGP_PEER_GROUP_NAME_i': 1,
'VR_BGP_PEER_GROUP_TYPE': "ebgp",  # ebgp is supported at the moment
'VR_BGP_PEER_NAME': "Peer-{0}.{1}",
'VR_BGP_PEER_AS': "201",
'VR_BGP_PEER_LOCAL_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],  # to be rotated
'VR_BGP_PEER_LOCAL_IP': "11.11.0.1/30",
'VR_BGP_PEER_PEER_IP': "22.22.0.1/24",
```

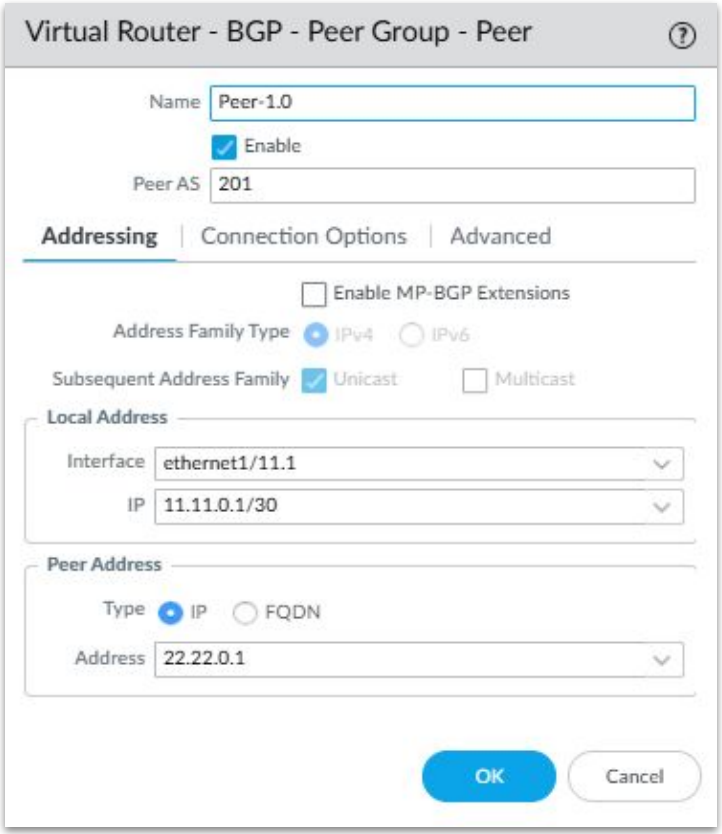
Virtual Router > BGP Peer Groups - Configuration pushed





Virtual Router > BGP Peer Groups - Configuration pushed





Virtual Router > BGP Peer Groups - Script configuration

```
# virtual router config
#
'N_VR_STATIC': 10,
'N_VR_BGP_PEER_GROUPS': 10,
'N_VR_BGP_PEERS_PER_GROUP': 10,

# ------------------------------------------------------------------------------
#
# VR configuration: static routes, BGP groups
#

'VR_BGP_VR': cf['DEFAULT_VR'],
'VR_BGP_PEER_GROUP_NAME': "Peer_Group-{0}",
'VR_BGP_PEER_GROUP_NAME_i': 1,
'VR_BGP_PEER_GROUP_TYPE': "ebgp",  # ebgp is supported at the moment
'VR_BGP_PEER_NAME': "Peer-{0}.{1}",
'VR_BGP_PEER_AS': "201",
'VR_BGP_PEER_LOCAL_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],  # to be rotated
'VR_BGP_PEER_LOCAL_IP': "2211:11::1/126",
'VR_BGP_PEER_PEER_IP': "2022:22::1/120",
```

Virtual Router > BGP Peer Groups - Script configuration

- Make sure the virtual router has the BGP AS defined with new peers. For example the VR has AS 200.

```
'VR_BGP_PEER_AS': "201",
```

If the address family is IPv6, make sure the following are satisfied.

- All address fields such Destination and Next Hop, are IPv6.

```
'VR_BGP_PEER_LOCAL_IP': "2211:11::1/126",
'VR_BGP_PEER_PEER_IP': "2022:22::1/120",
```

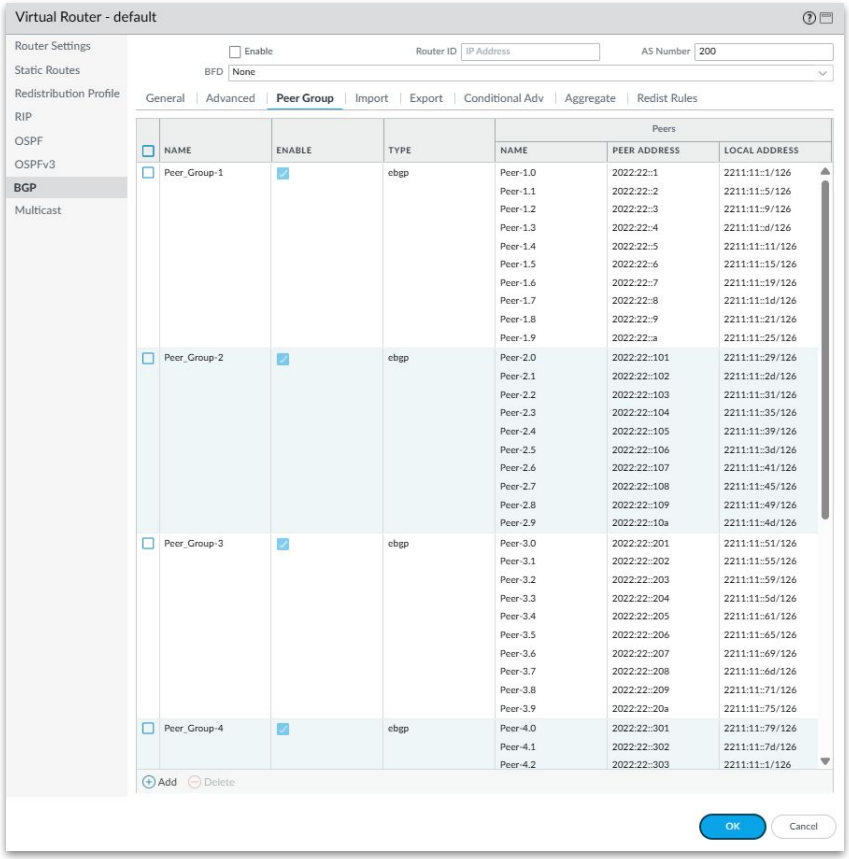
- All interface fields support IPv6.

```
'VR_BGP_PEER_LOCAL_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],
```

- The Interface IP's have to be consistent. You may check interface configuration.

```
'N_NET_IF_ETHERNET': 16,
'IF_ETHERNET_LIST': ["ethernet1/11", "ethernet1/12"],  # list of parent interfaces
'IF_ETHERNET_IP': "2211:11::1/126",

'VR_BGP_PEER_LOCAL_INTERFACE_LIST': [("ethernet1/11.{0}", 1, 16), ("ethernet1/12.{0}", 1, 16)],
'VR_BGP_PEER_LOCAL_IP': "2211:11::1/126",
```

Virtual Router > BGP Peer Groups - Configuration pushed



User-IP mapping

User-ID - configuration

```
# total IP-user mappings = N_UID_NETS x N_UID_ENTRIES
#
'N_UID_NETS': 6,
'N_UID_ENTRIES': 30,

# ------------------------------------------------------------------------------
#
# IP-user mappings
#

'UID_USER': "domain-{0}\\user-{1}",  # domain-$i\user$j
'UID_DOMAIN_i': 1,                   # i, the initial index value for domain
'UID_USER_j': 1,                     # j, the initial index value for user

# list of UID discrete ranges of IP's
#
'UID_IP': [
    "1.1.0.1/16",
    "2.2.0.1/16",
    "3.3.0.1/16",
    "4.4.0.1/16",
    "5.5.0.1/16",
    "2006:6::1/116",
    "7.7.0.1/16",
    "8.8.0.1/16",
    ],
'UID_TIMEOUT': "600",  # timeout in mins. 600 means 600 minutes
```

User-ID mapping

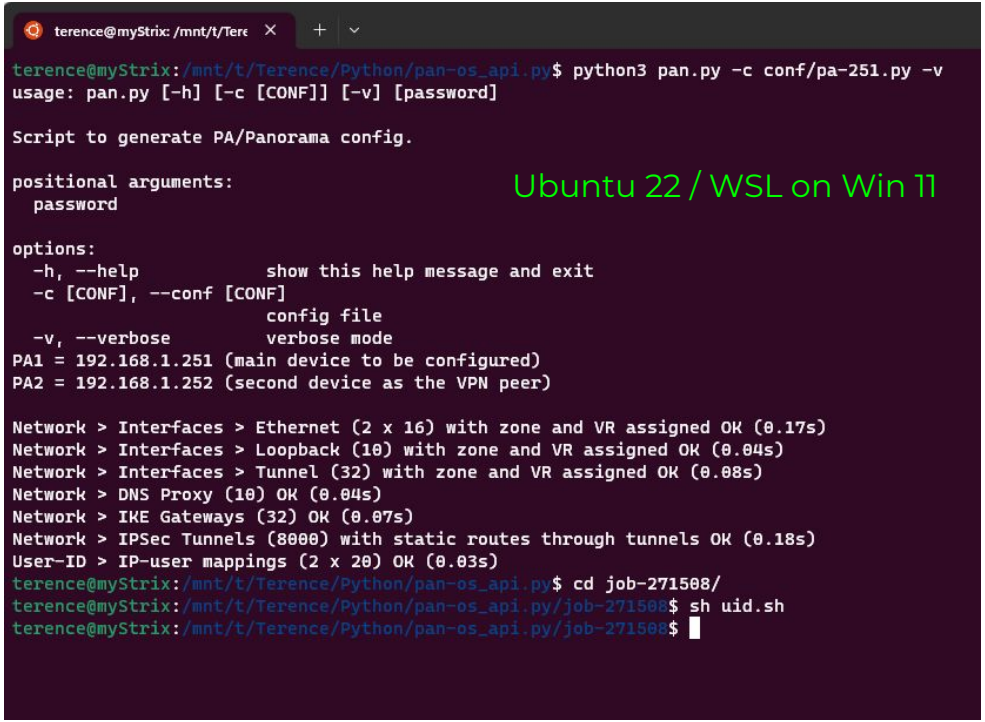
The generated uid.sh can be used to establish user-ID mapping through API.

```
$ sh uid.sh
```

PAN-OS CLI to observe the entries.

```
> show user ip-user-mapping all
```

```
> clear user-cache all
```



User-ID mapping

IP	Vsys	From	User	IdleTimeout(s)	MaxTimeout(s)
3.3.0.4	vsys1	XMLAPI	domain-c\user-4	35623	35623
2.2.0.19	vsys1	XMLAPI	domain-b\user-19	35623	35623
2006:6::11	vsys1	XMLAPI	domain-f\user-17	35623	35623
1.1.0.26	vsys1	XMLAPI	domain-a\user-26	35623	35623
3.3.0.13	vsys1	XMLAPI	domain-c\user-13	35623	35623
4.4.0.9	vsys1	XMLAPI	domain-d\user-9	35623	35623
2.2.0.9	vsys1	XMLAPI	domain-b\user-9	35623	35623
5.5.0.1	vsys1	XMLAPI	domain-e\user-1	35623	35623
5.5.0.6	vsys1	XMLAPI	domain-e\user-6	35623	35623
5.5.0.24	vsys1	XMLAPI	domain-e\user-24	35623	35623
2.2.0.12	vsys1	XMLAPI	domain-b\user-12	35623	35623
4.4.0.25	vsys1	XMLAPI	domain-d\user-25	35623	35623
3.3.0.18	vsys1	XMLAPI	domain-c\user-18	35623	35623
4.4.0.29	vsys1	XMLAPI	domain-d\user-29	35623	35623
1.1.0.22	vsys1	XMLAPI	domain-a\user-22	35623	35623
2.2.0.7	vsys1	XMLAPI	domain-b\user-7	35623	35623
3.3.0.10	vsys1	XMLAPI	domain-c\user-10	35623	35623
1.1.0.20	vsys1	XMLAPI	domain-a\user-20	35623	35623
1.1.0.7	vsys1	XMLAPI	domain-a\user-7	35623	35623
5.5.0.17	vsys1	XMLAPI	domain-e\user-17	35623	35623
3.3.0.11	vsys1	XMLAPI	domain-c\user-11	35623	35623
2.2.0.16	vsys1	XMLAPI	domain-b\user-16	35623	35623
2.2.0.15	vsys1	XMLAPI	domain-b\user-15	35623	35623
2006:6::1b	vsys1	XMLAPI	domain-f\user-27	35623	35623
2.2.0.2	vsys1	XMLAPI	domain-b\user-2	35623	35623
4.4.0.18	vsys1	XMLAPI	domain-d\user-18	35623	35623
3.3.0.24	vsys1	XMLAPI	domain-c\user-24	35623	35623
5.5.0.22	vsys1	XMLAPI	domain-e\user-22	35623	35623
4.4.0.10	vsys1	XMLAPI	domain-d\user-10	35623	35623
2.2.0.30	vsys1	XMLAPI	domain-b\user-30	35623	35623
2006:6::1c	vsys1	XMLAPI	domain-f\user-28	35623	35623
5.5.0.30	vsys1	XMLAPI	domain-e\user-30	35623	35623
3.3.0.22	vsys1	XMLAPI	domain-c\user-22	35623	35623
4.4.0.20	vsys1	XMLAPI	domain-d\user-20	35623	35623
4.4.0.26	vsys1	XMLAPI	domain-d\user-26	35623	35623
5.5.0.2	vsys1	XMLAPI	domain-e\user-2	35623	35623
3.3.0.30	vsys1	XMLAPI	domain-c\user-30	35623	35623
1.1.0.16	vsys1	XMLAPI	domain-a\user-16	35623	35623
2.2.0.4	vsys1	XMLAPI	domain-b\user-4	35623	35623
2.2.0.11	vsys1	XMLAPI	domain-b\user-11	35623	35623
3.3.0.12	vsys1	XMLAPI	domain-c\user-12	35623	35623
2006:6::7	vsys1	XMLAPI	domain-f\user-7	35623	35623
1.1.0.14	vsys1	XMLAPI	domain-a\user-14	35623	35623
1.1.0.3	vsys1	XMLAPI	domain-a\user-3	35623	35623
2006:6::6	vsys1	XMLAPI	domain-f\user-6	35623	35623
4.4.0.14	vsys1	XMLAPI	domain-d\user-14	35623	35623

lines 1-49

Panorama support

Panorama support

- Device Group and Template provisioning
- Configuration under created Device Groups and Templates
- With configuration target set to “PANORAMA”, xpaths will be adjusted accordingly.
- All previous configuration, including UID, has been tested with Panorama.
- Capacity testing of M-200/M-600 to support 1GB configuration (POC13663).

```
cf = {
    # 'TARGET': "PA",
    'TARGET': "PANORAMA",

    'PA1': {
        'HOST': "192.168.1.1",
        'USER': "admin",
        'PASS': "",           # this can be overridd
        'PASSENV': 'PAPASS',  # name of the environm
        'DESC': "main device to be configured"
    },
```

Panorama support - configuration

```
cf = {
    # 'TARGET': "PA",
    'TARGET': "PANORAMA",

    'PA1': {
        'HOST': "192.168.1.1",
        'USER': "admin",
        'PASS': "",           # this can be overridden with the environment variable see the next key PASSENV
        'PASSENV': 'PAPASS',  # name of the environment variable for the password
        'DESC': "main device to be configured"
    },

    'PANORAMA_DEVICE_GROUP': "Device_Group-001",
    'PANORAMA_TEMPLATE': "Template-001",
    'PANORAMA_TEMPLATE_STACK': "TempStack-001",
```

- Configuration under created Device Group “`Device_Group-001`“ and Template “`Template-001`”
- With configuration target set to “PANORAMA”, xpaths will be adjusted accordingly.
- All previous configuration, including UID, has been tested with Panorama.

Panorama Device Groups and Templates - Script configuration

```
# Panorama only
#
'N_PAN_DG': 10,    # device groups
'N_PAN_TPL': 10,   # templates

'DG_NAME': "DG-{0:03d}",
'DG_NAME_i': 11,

'TPL_NAME': "Template-{0:03d}",
'TPL_NAME_i': 11,
```

- Device Group and Template provisioning
- Here 10x Device Groups and 10x Templates will be created with specified name formats.

Panorama Device Groups and Templates - Configuration pushed

NAME	DESCRIPTION	AUTHORIZATION CODE	SW VERSION	USER ID INFORMATION	DEVICES/VIRTUAL SYSTEM	REFERENCE TEMPLATES
DG-011						
DG-012						
DG-013						
DG-014						
DG-015						
DG-016						
DG-017						
DG-018						
DG-019						
DG-020						

NAME	DESCRIPTION	TYPE	STACK	USER ID INFORMATION	DEVICES	VARIABLES	DEVICE KEY-VALUE TABLE
Template-011		template				Manage...	
Template-012		template				Manage...	
Template-013		template				Manage...	
Template-014		template				Manage...	
Template-015		template				Manage...	
Template-016		template				Manage...	
Template-017		template				Manage...	
Template-018		template				Manage...	
Template-019		template				Manage...	
Template-020		template				Manage...	

Demo

Supported PA configuration

- Device > Local users
- Network > Interfaces > Ethernet (with vsys, zone and vr assignment)
- Network > Interfaces > Loopback (with vsys, zone and vr assignment)
- Network > Interfaces > Tunnel (with vsys, zone and vr assignment)
- Network > Zones
- Network > DNS Proxy
- Network > IKE Gateways
- Network > IPSec Tunnels (with static routes through tunnels)
- Objects > Addresses
- Objects > Address Groups
- Objects > Services
- Objects > Service Groups
- Objects > Custom URL Category with url.txt
- Policies > Security
- Policies > NAT
- Policies > PBF

- Network > VR > Static Routes
- Network > VR > BGP peer groups x peers
- User-ID mapping

(Panorama only)

- Panorama > Device Groups
- Panorama > Templates

Documentation as additional references

POC Team is consistently developing technical content to accelerate testing and POC cycles. Here this is the collection of documents for various use cases. We welcome your feedback to help enhance our productivity with detailed and efficient documentation.

https://drive.google.com/drive/folders/1azplLToTYzfwynAF4gNlmC1W9ViaL9Xq?usp=drive_link

Note 9 - Subscriber-ID
Note 8 - PAN-OS API with Python
Note 7 - Explicit Web proxy w/ Network Packet Broker
Note 6 - PAN-OS SD-WAN
Note 5 - KVM w/ OVS-DPDK
Note 4 - Prisma SD-WAN POC setup v2.0
Note 3 - MFA
Note 2 - GlobalProtect w/ AD
Note 1 - User-ID

Developed tools on Github for POC and system admin https://github.com/telee0

paloalto NETWORKS

Thank you