digitalscepter

# Implementing Zero-Trust Security Architecture

ZACHRY SUM - DIRECTOR OF TECHNICAL SERVICES, DIGITAL SCEPTER
JON ROBINSON - PRESIDENT, DIGITAL SCEPTER

November 19th, 2024

# Service Catalog

- Firewall migrations
- Firewall operations - mass upgrades, backups, change/remove/add
- Firewall Healthchecks
- Panorama design
- Zero Trust Network Access
- Network Segmentation
- MFA
- SSL Decryption
- Inbound SSL Inspection
- Remote Access (“Always on”)
- Securing Cloud infrastructure
- Dual ISP redundancy
- Network engineering
- Endpoint Security/EDR/MDR

# Contracts

- CMAS
- NASPO
- SPURR
- OMNIA Partners

# Vendors

- Palo Alto Networks
- Crowdstrike
- SentinelOne
- Okta
- Arista
- Juniper
- HPe/Aruba
- AWS
- Microsoft/Azure
- Proofpoint
- Zscaler
- Gigamon
- Rapid7
- Knowbe4
- Netskope

# Agenda

1. Zero Trust Concept Overview
2. Negative Vs Positive Security Model
3. Zero Trust Prerequisites
4. Security Policy Building Blocks
5. Zero Trust Journey
6. Network Segmentation

# Zero Trust Concept Overview

# What is Zero Trust?

- A cybersecurity framework that's built upon the principle that no user should be implicitly trusted, i.e. default deny everywhere
- Users should be expected to meet strict criteria in order to be granted access to resources. A sample of items that should be verified:
  - Verify the identity of the user
  - Validate the user through the use of MFA
  - Ensure source device is trusted and healthy
  - Apply application level traffic controls, e.g. App-ID

How can this be achieved?

# Negative Vs Positive Security Model

# Negative Security Model

- The negative security model works on the principle that specific traffic will be denied, and any traffic not explicitly denied will be permitted
- This model is substantially more permissive than what is needed by an organization but can be used to get quick wins to stop threats and risky traffic

Device Group: digital_scepter_negative

| | NAME | LOCATION | TAGS | TYPE | Source | | | | Destination | | | APPLICATI... | SERVICE | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | | | | | |
| 1 | deny all inbound | digital_scepter_negative | none | universal | untrust | any | any | any | trust | any | any | any | any | Deny | | |
| 2 | permit all | digital_scepter_negative | none | universal | any | any | any | any | any | any | any | any | any | Allow | | |



digitalscepter

# Positive Security Model

- The positive security model works exactly opposite the negative security model–specific traffic is permitted, and everything else is explicitly denied
- This model is much more restrictive even in its simplest form

Device Group: digital_scepter_positive

| | | | | | Source | | | | Destination | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | NAME | LOCATION | TAGS | TYPE | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | ACTION | PROFILE | OPTIONS |
| 1 | permit dns | digital_scepter_positive | none | universal | trust | any | any | any | dns | dns_server_01 | any | any | dns_udp_53 | Allow | | |
| 2 | permit internet | digital_scepter_positive | none | universal | trust | any | any | any | untrust | any | any | any | tcp_80<br>tcp_443 | Allow | | |
| 3 | deny all | digital_scepter_positive | none | universal | any | any | any | any | any | any | any | any | any | Deny | none | |



digitalscepter

# Positive Security Model

- But what is getting through on tcp/80 and tcp/443?
- Service based policies aren't enough, App-ID should be used as much as possible

| Application Name | Bytes | Sessions |
|---|---|---|
| web-browsing | 27.05 G | 471.95 k |
| outlook-web-online | 6.13 G | 53.99 k |
| trendmicro | 111.58 M | 48.59 k |
| google-analytics | 2.18 G | 37.33 k |
| flash | 5.71 G | 12.56 k |
| facebook-base | 624.98 M | 11.26 k |
| http-proxy | 285.27 M | 10.02 k |
| twitter-base | 101.45 M | 9.63 k |
| google-plus-base | 342.04 M | 7.99 k |
| ocsp | 22.24 M | 6.99 k |
| ms-office365-base | 100.05 M | 5.40 k |
| itunes-base | 247.51 M | 4.93 k |
| ms-update | 1.43 G | 3.54 k |
| google-docs-base | 3.11 G | 3.53 k |
| youtube-base | 6.67 G | 3.18 k |
| icloud-base | 16.23 G | 2.66 k |
| google-drive-web | 567.95 M | 2.63 k |
| ammyy-admin | 2.96 M | 2.53 k |
| new-relic | 13.21 M | 2.45 k |
| panos-web-interface | 6.78 M | 2.01 k |
| gmail-base | 326.02 M | 2.01 k |
| http-audio | 1.21 G | 1.89 k |
| dropbox | 43.01 M | 1.85 k |
| msrpc | 2.66 M | 1.72 k |
| instagram | 162.32 M | 1.64 k |
| ooyala | 24.52 M | 1.21 k |
| sharepoint-base | 228.16 M | 1.20 k |
| apple-push-notifications | 13.42 M | 1.12 k |
| google-update | 1.90 G | 982 |
| disqus | 6.71 M | 956 |

# Zero Trust Prerequisites

# Are You Ready For Zero Trust?

- Before you can effectively implement zero trust, the below items should be evaluated:
  - **User-ID** - a core component of zero trust is controlling access by user, not just IP address, so ensuring IP to user mappings are up to date and distributed across your firewalls is critical
  - **Network Segmentation** - the more you isolate assets across unique subnets, the more you can control what is permitted between those assets
  - **Device Posturing** - allowing a user to assets is only advised once a device has been determined to be healthy through posture checks

# Security Policy Building Blocks

# Security Policy Match Conditions

- **Zone** - Source and Destination
- **IP Address** - Source and Destination
- **User** - Source and Destination
- **Device** - Source and Destination
- **Application**
- **Service**
- **URL Category**

| NAME | TAGS | Source | | | | Destination | | | APPLICATI... | SERVICE | URL CATEGORY |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | | | |

# IP Addresses and Zones

- Always consider how you can avoid using “any”
- IP addresses should be as specific as possible
- Leverage Dynamic Address Groups or External Dynamic Lists rather than subnets when possible
- Good:

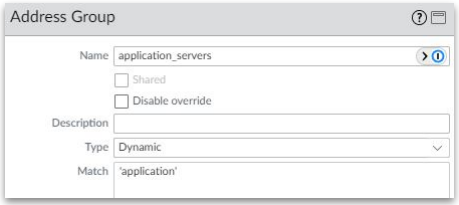
| | | Source | | | | Destination | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAME | TAGS | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | URL CATEGORY | ACTION |
| allow servers to internet | none | servers | any | any | any | untrust | any | any | any | any | any | Allow |

- Better:

| | | Source | | | | Destination | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAME | TAGS | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | URL CATEGORY | ACTION |
| allow servers to internet | none | servers | application_servers<br>database_servers<br>web_servers | any | any | untrust | any | any | any | any | any | Allow |

- Best:

| | | Source | | | | Destination | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAME | TAGS | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | URL CATEGORY | ACTION |
| allow servers to internet | none | servers | application_servers<br>database_servers<br>web_servers | any | any | untrust | any | any | any | any | any | Allow |






digitalscepter

# User

- Users should be required on all security policies that are sourced from devices where users log in
- Which policy would you rather use?

| | | Source | | | | Destination | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAME | TAGS | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | URL CATEGORY | ACTION |
| allow access to accounting - user | none | trust | any | ds\accounting_users | any | servers | accounting_server | any | any | application-default | any | Allow |
| allow access to accounting - ip | none | trust | accounting_users | any | any | servers | accounting_server | any | any | application-default | any | Allow |

- Ideally, we combine the two:

| | | Source | | | | Destination | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAME | TAGS | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | APPLICATI... | SERVICE | URL CATEGORY | ACTION |
| allow access to accounting | none | trust | accounting_users | ds\accounting_users | any | servers | accounting_server | any | any | application-default | any | Allow |



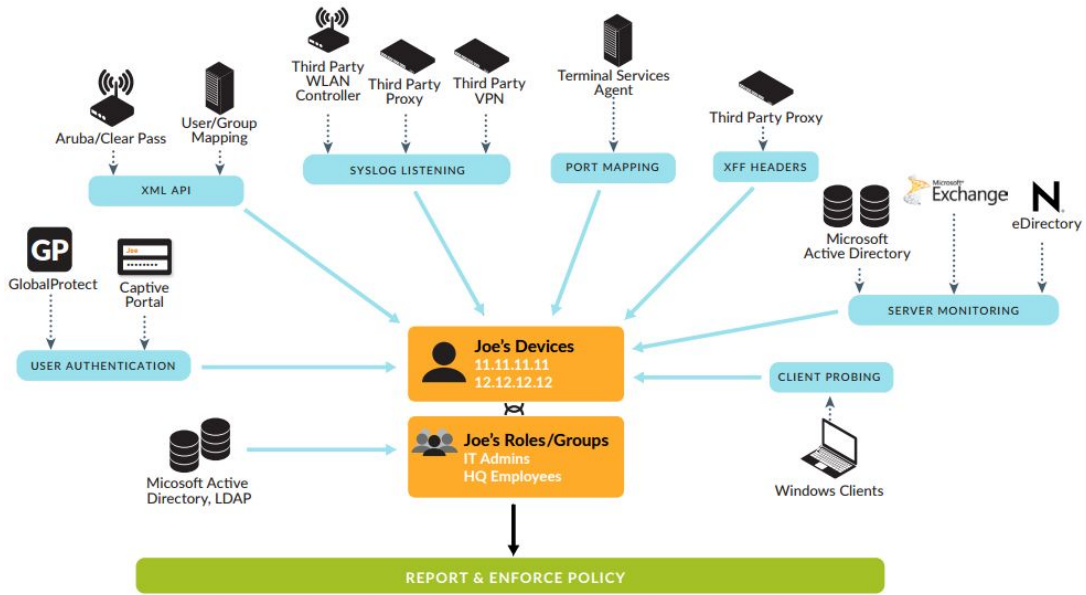
digitalscepter

# User

- Even policies sourced from servers can leverage User-ID, but this shouldn't apply to traffic that may be required without a logged in user, e.g. services, system updates
  - Practical example would be that servers can't access online-storage without User-ID
- Basic infrastructure rules should not leverage users in policy. Things like DNS and Active Directory traffic for example.
- With a heavy dependency on users for policy matching, User-ID architecture is important

# User-ID Mappings

- Use as many sources as possible
  - AD Domain Controllers
  - GlobalProtect VPN
  - Wireless Controllers
  - Captive Portal
  - Syslog
  - XML API
  - Other 3rd party integrations
- Design your sources to be highly available
- Ensure that all firewalls have all mappings

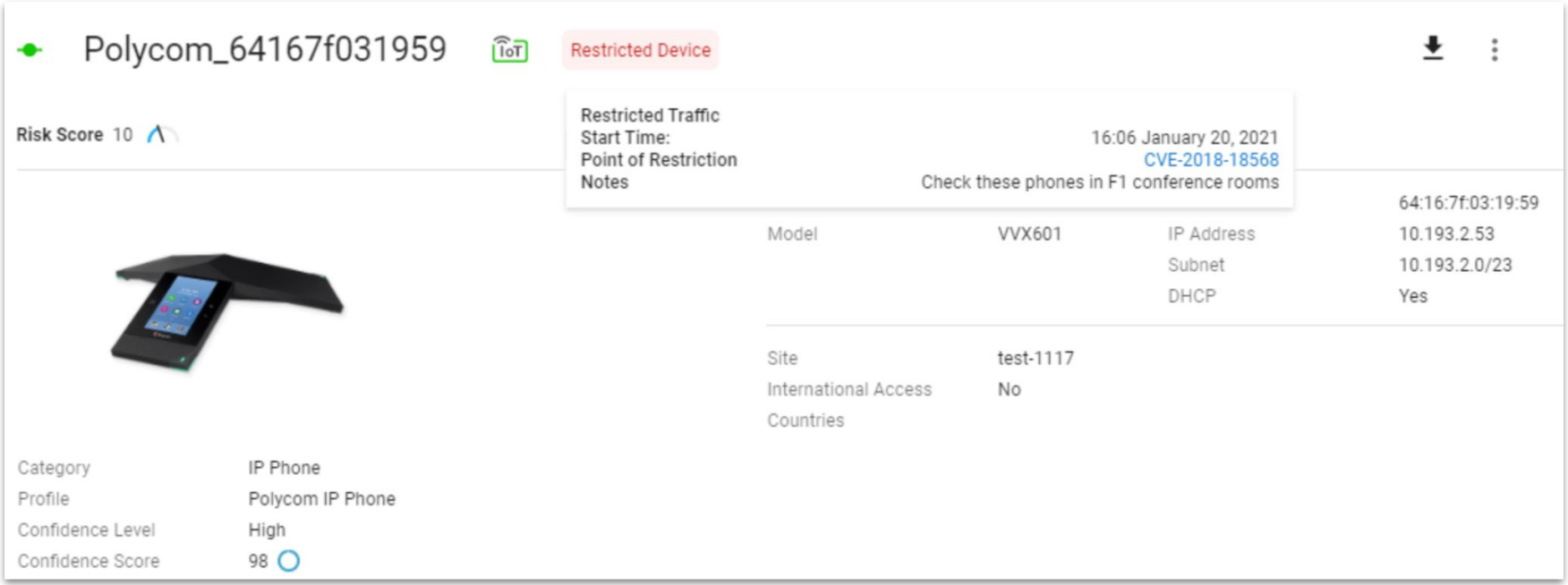





digitalscepter

# Devices

- IOT devices can be profiled and policies automatically created with IOT Security
  - Additionally vulnerable devices can have access restricted when detected

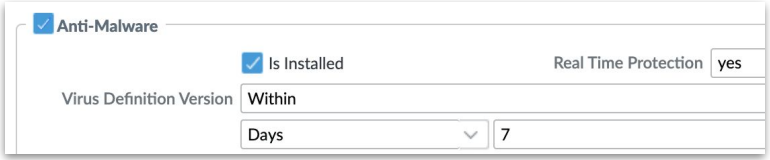




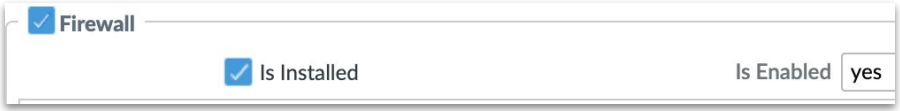

digitalscepter

# Devices

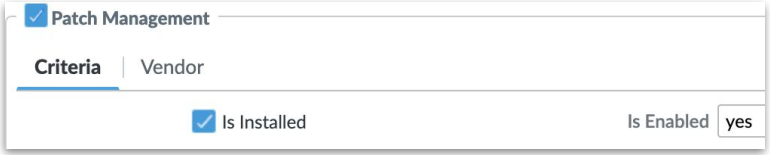
- Leverage HIP checks for all GlobalProtect enabled devices when internal and external
- Check for enabled antivirus

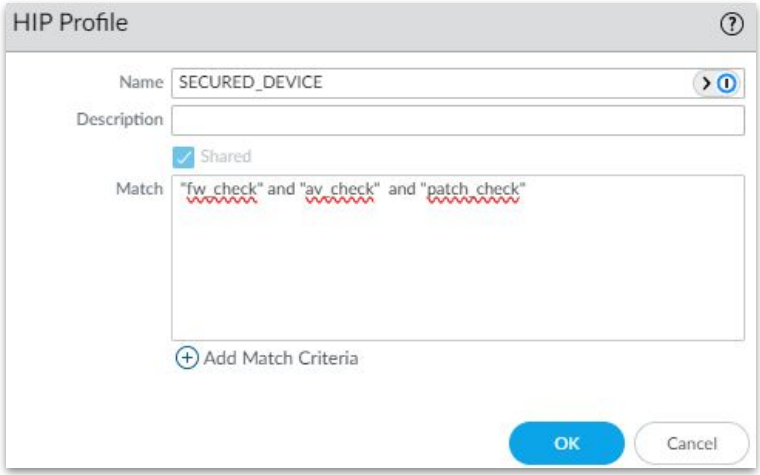



- Check for enabled firewall




- Check for patch management






digitalscepter

# Application

- For inbound rules, a static application list is best

| NAME | TAGS | Source | | | | Destination | | | APPLICATI... | SERVICE | URL CATEGORY | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | | | | | | |
| allow untrust to ext ftp server | none | untrust | any | any | any | dmz | extftp01_public | any | ftp | application-default | any | Allow | | |

- For targeted outbound rules, static applications is also ideal

| NAME | TAGS | Source | | | | Destination | | | APPLICATI... | SERVICE | URL CATEGORY | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | | | | | | |
| allow developers to github | none | trust | any | ds\developers | any | untrust | any | any | git-base<br>github | tcp_80<br>tcp_443 | github | Allow | | |

- For general internet access policies, application filters is the ideal method

| NAME | TAGS | Source | | | | Destination | | | APPLICATION | SERVICE | URL CATEGORY | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZONE | ADDRESS | USER | DEVICE | ZONE | ADDRESS | DEVICE | | | | | | |
| permit internet | none | trust | any | any | any | untrust | any | any | af_business tools<br>af_collaboration<br>af_general internet | tcp_80<br>tcp_443 | any | Allow | | |



digitalscepter

# Service

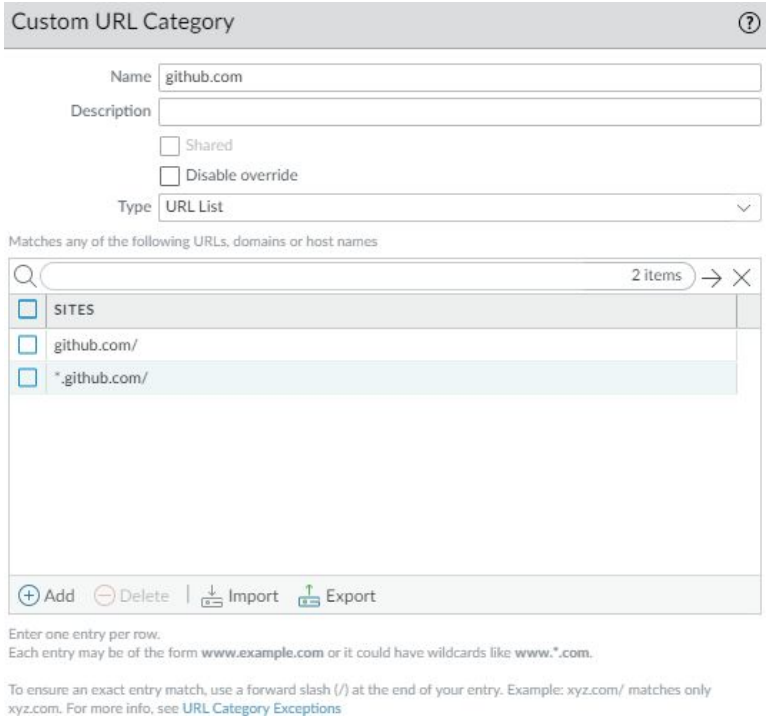
- Services should be leveraged differently depending on the context of the policy
- As a rule of thumb, the following guideline applies to using services on policies:
  - Deny rules should use service “any”
  - Allow rules should use specific services
  - Allow rules for application with dynamic port usage should use “application-default”
- Every application PAN publishes has known standard ports that are applied on a rule when using “application-default” on a policy




digitalscepter

# URL Category

- URL Categories can be used as a “destination” match condition
- These can be used for targeted internet access policies:
    - A department needs access to a specific website that is denied for everyone else. An allow rule can be created with a custom URL category to match on the specific website that needs to be permitted
- Only http/https traffic will match policies with URL categories





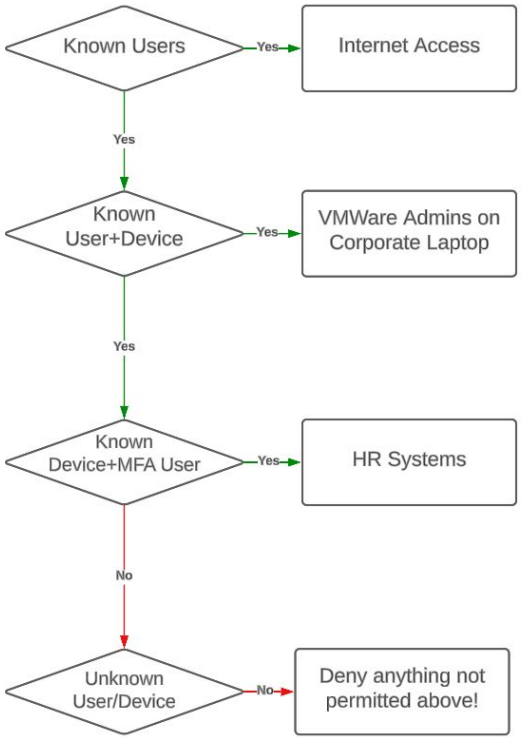

digitalscepter

# Zero Trust Journey



digitalscepter

# Sample Rulebase




Session has been identified with valid credentials

| NAME | TAGS | TYPE | Source ZONE | Source ADDRESS | Source USER | Source DEVICE | Destination ZONE | Destination ADDRESS | Destination DEVICE | APPLICATI... | SERVICE | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Known Users Internet Access | none | universal | inside | any | known-user | any | outside | any | any | any | application-default | Allow | | |

Valid Credentials and known device

| NAME | TAGS | TYPE | Source ZONE | Source ADDRESS | Source USER | Source DEVICE | Destination ZONE | Destination ADDRESS | Destination DEVICE | APPLICATI... | SERVICE | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Allow VMware Admins to vCenter | none | universal | inside | any | company\vmware_admins | corporate_secured | vmware | any | any | any | any | Allow | | |

Valid Credentials and known device

| | TAGS | TYPE | Source ZONE | Source ADDRESS | Source USER | Source DEVICE | Destination ZONE | Destination ADDRESS | Destination DEVICE | APPLICATI... | SERVICE | ACTION | PROFILE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| taff to HR Systems | none | universal | inside | any | company\hr_staff | corporate_secured | hr | hr_sy... | any | any | any | Allow | |

| NAME | TAGS | Source ZONE | Source ADDRESS | Source USER | Source DEVICE | Destination ZONE | Destination ADDRESS | Destination DEVICE | SERVICE | AUTHENTICATION ENFORCEMENT |
|---|---|---|---|---|---|---|---|---|---|---|
| Force MFA for HR Systems | none | inside | any | company\hr_staff | any | hr | hr_systems | any | any | force_mfra |

MFA Enforced for Access

| NAME | TAGS | TYPE | Source ZONE | Source ADDRESS | Source USER | Source DEVICE | Destination ZONE | Destination ADDRESS | Destination DEVICE | APPLICATI... | SERVICE | ACTION | PROFILE | OPTIONS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Deny All Unknown | none | universal | inside | any | any | any | any | any | any | any | any | Deny | none | |



digitalscepter

# Improving Existing Security Posture

The idea of getting to a zero trust model can be overwhelming. Try to break it into manageable chunks of work. For example:

- Add MFA to GlobalProtect
- Enable inbound inspection and convert inbound rules to use App-id
- Add User-ID to policies that enable access to critical systems
- Create internet access rules based on application filters
- Analyze the rulebase and try to find 3 things that you can change to improve security

# Zero Trust Prioritization

When considering how to move towards a zero-trust rulebase, you must consider the difficulty of the work along with the amount of improvement to your security posture. Based on these criteria, we have general recommendations:

1. MFA for remote access
    - Email or SMS alerts for successful logins from outside of the US

      ( status eq 'success' ) and ( srcregion neq 'US' ) and (( eventid eq 'portal-auth' ) or ( eventid eq 'gateway-auth' ))
2. SSL Decryption
3. Security Profiles on all allow rules
4. App-ID
5. User-ID
6. Device-ID

# Network Segmentation

# Overview

- Network segmentation is the process of classifying assets into unique subnets on your network with the intent of firewalling between these subnets.
- Firewalling these subnets is generally achieved by making the firewall the default gateway for the subnets assets are on, but another common option is using VRFs to force inter-VRF traffic through a firewall.

# Benefits

- Content inspection between subnets
- Prevent lateral spread of threats
- App-ID and User-ID between subnets
- Visibility into traffic flows between subnets
- Ability to easily isolate assets that may be compromised
- Foundation of a Zero Trust Architecture

# Methods of Implementation

Depending on your network topology, we would suggest taking one of the following design options:

1. Firewall on a stick model, with SVIs migrated to firewalls
2. VRF-Lite using different transit VLANs
3. L2 VNIs over VXLAN*
4. L3VPN Technologies (L3VPN / EVPN)*

* - Requires >1500 MTU or TCP MSS Clamping

# Note on MTU

The default Internet MTU is 1500 bytes.

- Clients will use this MTU to negotiate their TCP Maximum Segment Size.
  - 1460 bytes is typical: MTU(1500) - IP Header(20) - TCP Header(20)

If you use an overlay technique, there's additional per packet overhead.
To accommodate this, either jumbo frames or TCP Clamping may be used.
If MTU isn't increased - or client's aren't aware - fragmentation will occur (Bad).

Most switches support Jumbo frames up to 9000 bytes, some further (9200+).
Most ISPs also support Jumbo frames on their Ethernet service connections.

# MTU/TCP-MSS Examples

- Switch MTU defines the maximum frame size a switch will cary before it is dropped. (Default is 1500 bytes).
- This can typically be increased without impact, although the switch may require a reload.
- Care should also be taken if the switch functions as a router.

```
interface Ethernet1/3
  no switchport
  mtu 9216
```

```
SW1(config)#system mtu jumbo 9198
```

- TCP MSS Clamping is typically automatic on tunnel interfaces. Though it may need to be manually defined.
- This configures the router to alter the TCP Maximum Segment Size negotiated during the TCP 3-way handshake between a client and host.

```
SW(config-if)#ip tcp adjust-mss 1380
```



digitalscepter

# Choosing a Solution

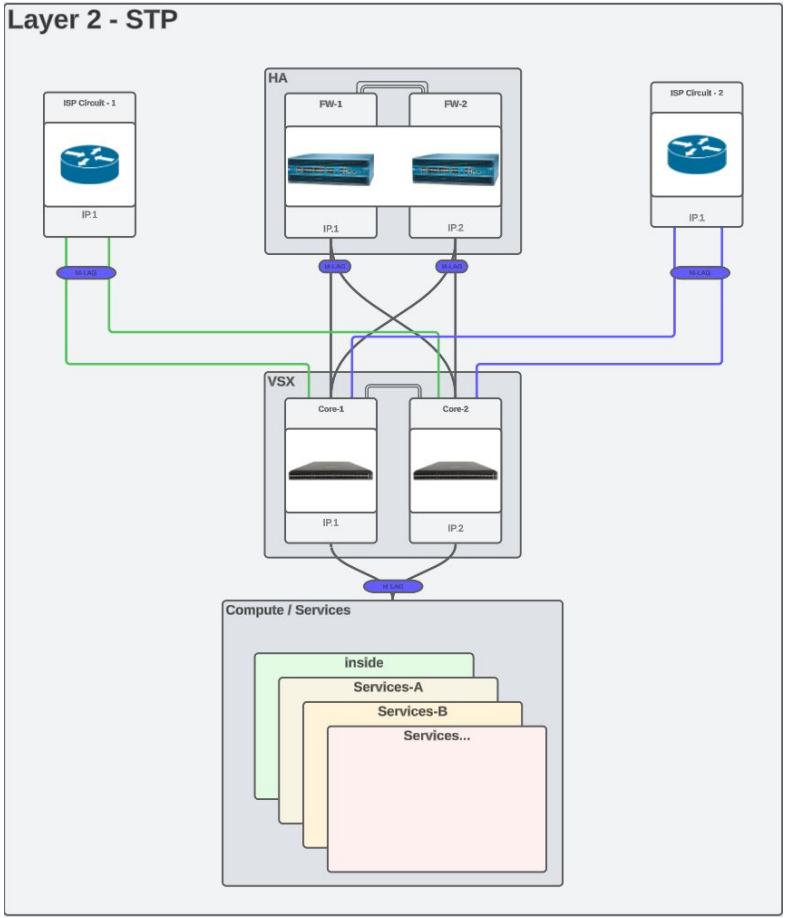
| Supported Condition | FW on a Stick | VRF Lite | L2 VNIs | L3VPN |
|---|---|---|---|---|
| Layer 2 between sites | Yes | Yes | Yes | Yes |
| Layer 3 between sites | No | Yes | Yes | Yes |
| Standard MTU | Yes | Yes | No | No |
| Jumbo frames | Yes | Yes | Yes | Yes |
| Low latency Intrasite | No | Yes | No | Yes |
| Scalability | Yes | No | No | Yes |

# Methods of Implementation

**Firewall on a stick**

+Simple design
+Quick migration
-Dependency on L2 links to remote sites for firewalling remote site networks
-VLANs can't overlap*
-MAC Limitations on Leased Circuits

*-802.1ad Q-in-Q may be a work-around.



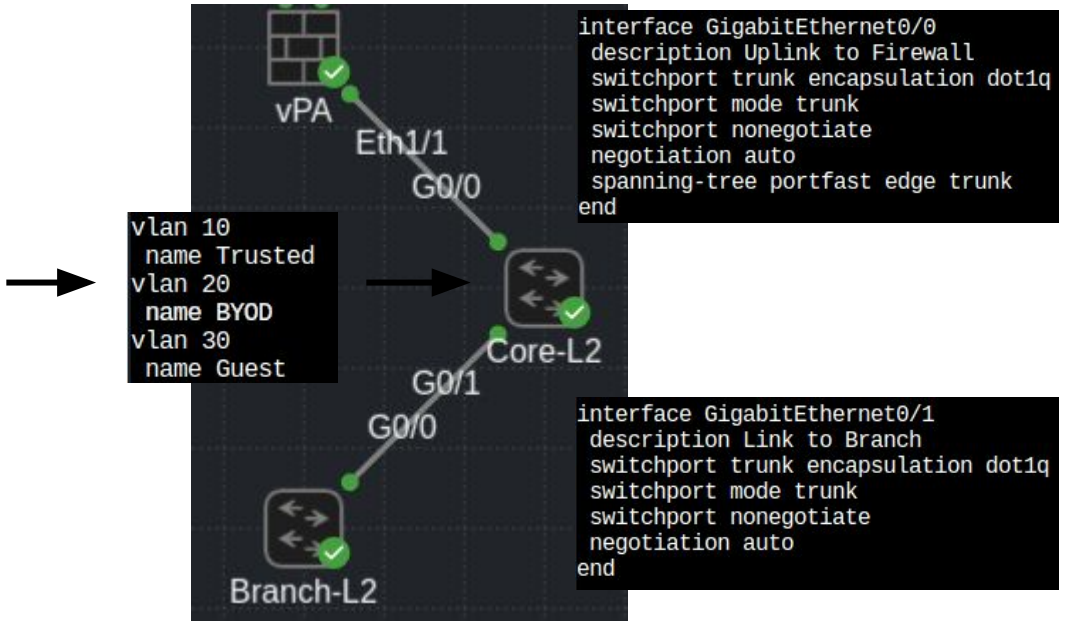



digitalscepter

# Methods of Implementation

## Firewall on a stick

| INTERFACE | INTERF... TYPE | LINK STATE | IP ADDRESS | SECURI... ZONE |
|---|---|---|---|---|
| ethernet1/1 | Layer3 | | none | none |
| ethernet1/1.10 | Layer3 | | 10.1.10.254/24 | Trusted |
| ethernet1/1.20 | Layer3 | | 10.1.20.254/24 | BYOD |
| ethernet1/1.30 | Layer3 | | 10.1.30.254/24 | Guest |




**Firewall on a Stick/VLAN Extension:**
You only need Layer 2 VLANs and Trunks configured.

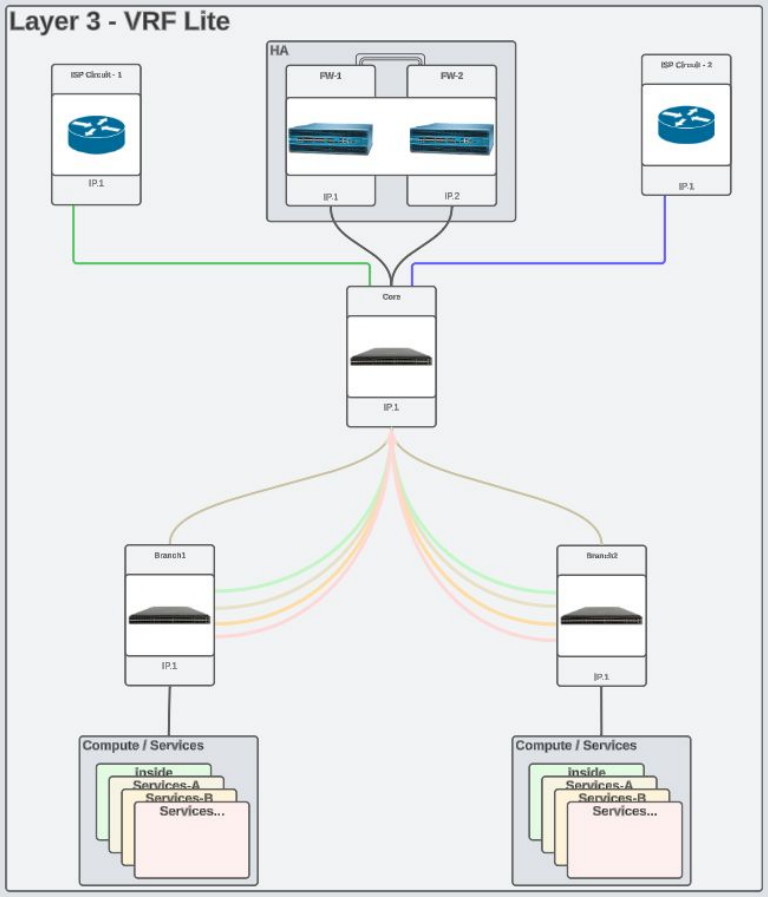


digitalscepter

# Methods of Implementation

**VRF-Lite**

+VLANs can overlap
+Smaller broadcast domains
+Widely supported
+VRF-Lite + Tunnel can act as a basic overlay.
+/- VRF-Lite using 802.1q has no overlay overhead. Tunnel based overlay has high overhead.
-Possible Dependency on 802.1q L2 links to remote site
-Not-scalable - Dedicated routing protocol per VRF/Zone.






digitalscepter

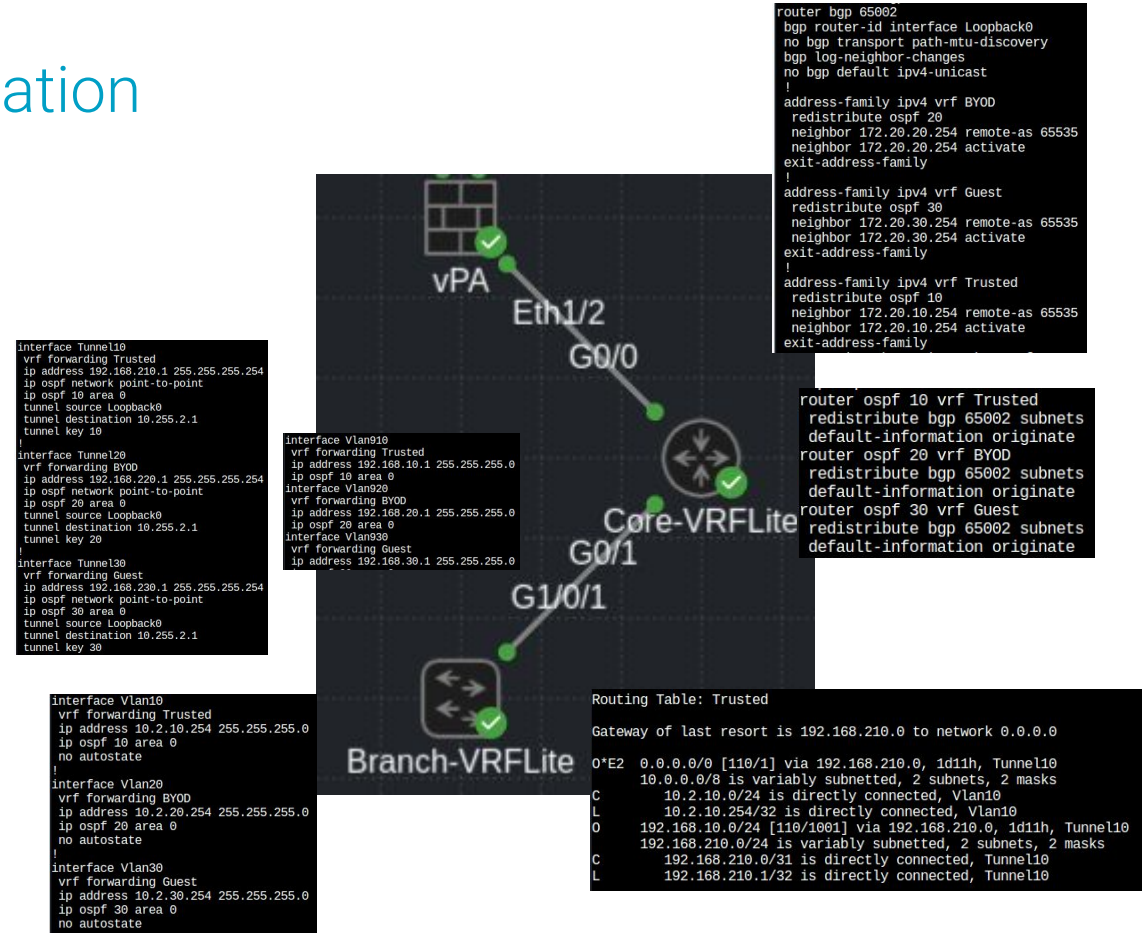
# Methods of Implementation

## VRF-Lite

| INTERF... | COMMENT | IP ADDRESS | SECURI... ZONE |
|---|---|---|---|
| vlan | | none | none |
| vlan.10 | Trust - L3 Peer | 172.20.10.254/24 | Trusted |
| vlan.20 | BYOD - L3 Peer | 172.20.20.254/24 | BYOD |
| vlan.30 | Guest - L3 Peer | 172.20.30.254/24 | Guest |

**VRF Lite:**

- Each VRF needs its own router process and path.
- Each router in the path needs to have VRF configuration.



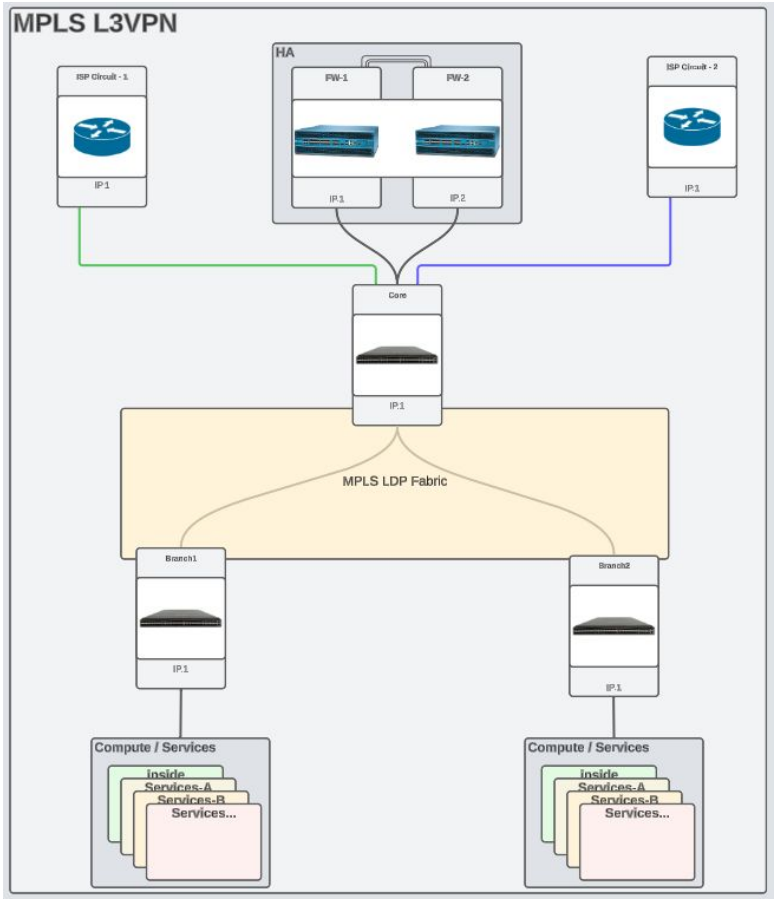



digitalscepter

# Methods of Implementation

**MPLS L3VPN**

+VLANs can overlap
+Smaller broadcast domains
+Highly Scalable (ISPs use it Globally)
+Low Overlay Overhead (8 bytes)
-All devices in labeled path need to support MPLS.
-Not a common skillset.
-TCP Clamping Not Easily Implemented (Use Jumbo MTU)



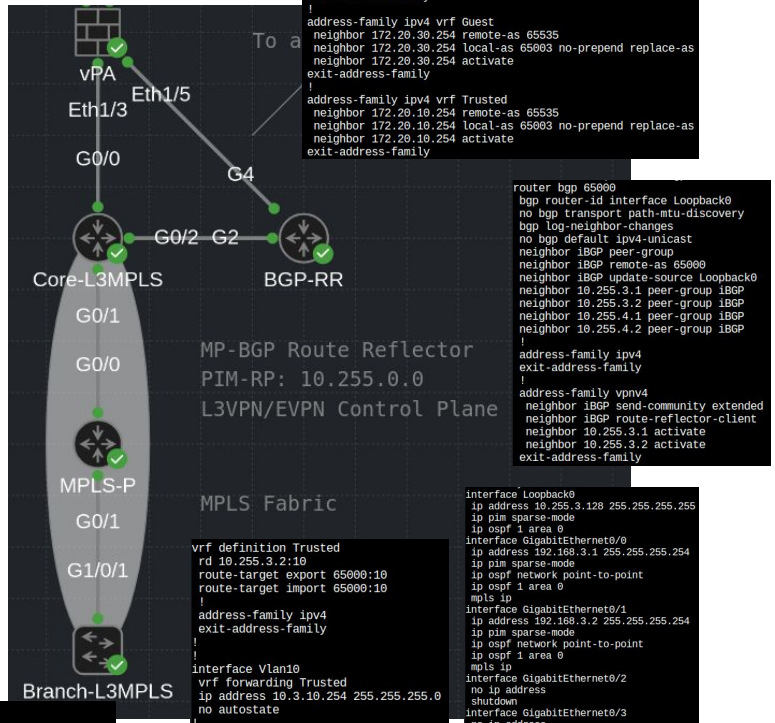



digitalscepter

# Methods of Implementation

## MPLS L3VPN

| INTERF... | COMMENT | IP ADDRESS | SECURI... ZONE |
|---|---|---|---|
| vlan | | none | none |
| vlan.10 | Trust - L3 Peer | 172.20.10.254/24 | Trusted |
| vlan.20 | BYOD - L3 Peer | 172.20.20.254/24 | BYOD |
| vlan.30 | Guest - L3 Peer | 172.20.30.254/24 | Guest |




**MPLS L3VPN:**
iBGP Extended Communities are used to Import/Export Routes per VRF. MPLS LDP will dynamically build a path to carry the data.

- OSPF is used in the underlay to provide reachability between loopbacks
- BGP-Route Reflector is used for easy scalability.

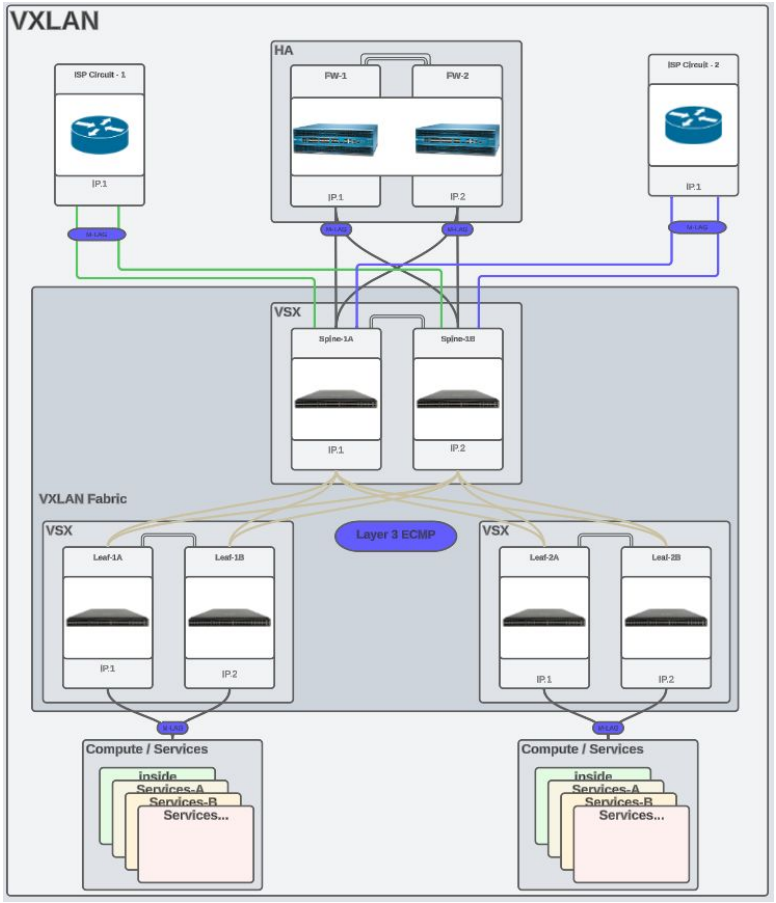


digitalscepter

# Methods of Implementation

**BGP EVPN**

+VLANs can overlap
+Smaller broadcast domains
+Highly Scalable (DC/Colos use it Globally)
+Data carried by UDP datagram - No special transport requirements.
+Can function as both L2 and L3 extension.
-High Overlay Overhead (## bytes)
-Not a common skillset.
-TCP Clamping Not Easily Implemented (Use Jumbo MTU)






digitalscepter

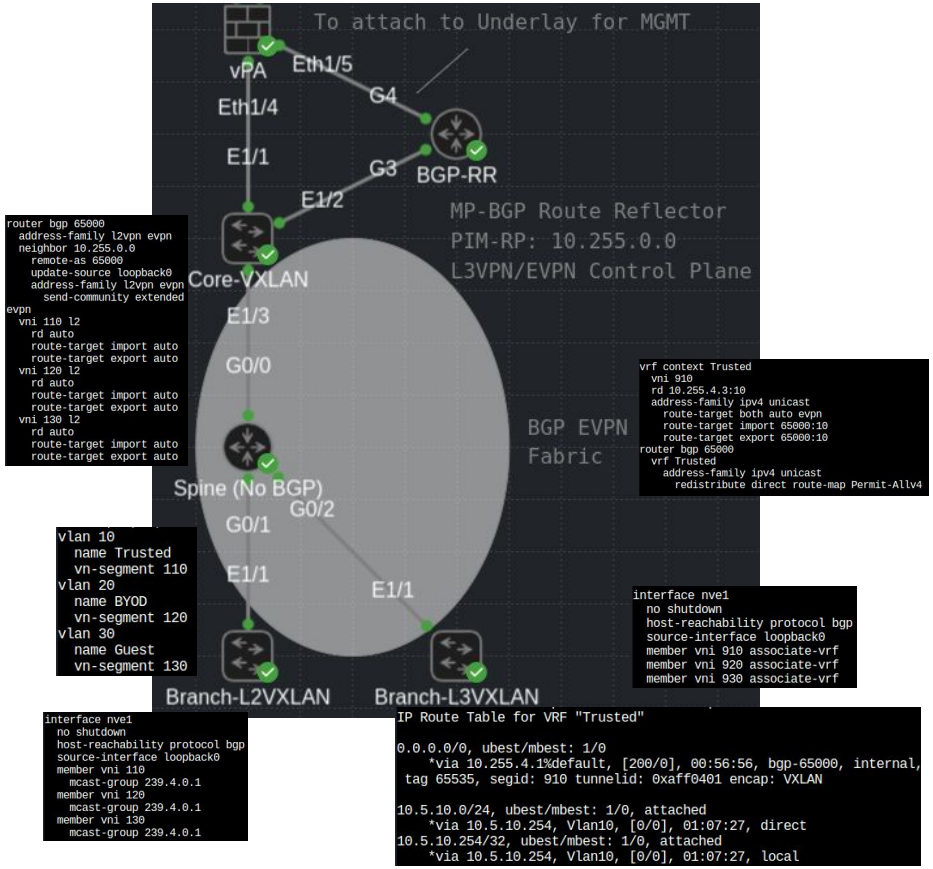
# Methods of Implementation

## BGP EVPN

| INTERF... | COMMENT | IP ADDRESS | SECURI... ZONE |
|---|---|---|---|
| vlan | | none | none |
| vlan.10 | Trust - L3 Peer | 172.20.10.254/24 | Trusted |
| vlan.20 | BYOD - L3 Peer | 172.20.20.254/24 | BYOD |
| vlan.30 | Guest - L3 Peer | 172.20.30.254/24 | Guest |

**BGP EVPN:**
iBGP Extended Communities are used to Import/Export Routes per VRF/VNI. VXLAN NVEs will dynamically forward traffic to peer switches.

- OSPF is used in the underlay to provide reachability between loopbacks.
- PIM is used to create multicast underlay for flood BUM traffic. (Broadcast, Unknown-unicast, and Multicast)
*Non-Multicast options are also available (Ingress-Replication)
- BGP-Route Reflector is used for easy scalability.






digitalscepter

# Note on MTU (Examples)

- 802.1q

```
▶ Frame 8: 118 bytes on wire (944 bits), 118 bytes captured (944 bits)
▶ Ethernet II, Src: RealtekU_00:12:27 (52:54:00:00:12:27), Dst: RealtekU_0f:d8:65 (52:54:00:0f:d8:65)
▶ 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 910
▶ Internet Protocol Version 4, Src: 10.2.10.254, Dst: 8.8.8.8
▶ Internet Control Message Protocol
```

- MPLS

```
▶ Frame 7: 122 bytes on wire (976 bits), 122 bytes captured (976 bits)
▶ Ethernet II, Src: RealtekU_01:5c:0b (52:54:00:01:5c:0b), Dst: RealtekU_15:6d:ef (52:54:00:15:6d:ef)
▶ MultiProtocol Label Switching Header, Label: 17, Exp: 0, S: 0, TTL: 255
▶ MultiProtocol Label Switching Header, Label: 56, Exp: 0, S: 1, TTL: 255
▶ Internet Protocol Version 4, Src: 10.3.10.254, Dst: 8.8.8.8
▶ Internet Control Message Protocol
```

- GRE

```
▶ Frame 10: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits)
▶ Ethernet II, Src: RealtekU_00:12:0d (52:54:00:00:12:0d), Dst: RealtekU_0f:d8:65 (52:54:00:0f:d8:65)
▶ Internet Protocol Version 4, Src: 10.255.2.2, Dst: 10.255.2.1
▶ Generic Routing Encapsulation (IP)
▶ Internet Protocol Version 4, Src: 10.2.10.254, Dst: 8.8.8.8
▶ Internet Control Message Protocol
```

- VXLAN

```
▶ Frame 6: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits)
▶ Ethernet II, Src: RealtekU_18:f4:60 (52:54:00:18:f4:60), Dst: 52:10:d7:d6:1b:08 (52:10:d7:d6:1b:08)
▶ Internet Protocol Version 4, Src: 10.255.4.3, Dst: 10.255.4.1
▶ User Datagram Protocol, Src Port: 52215, Dst Port: 4789
▶ Virtual eXtensible Local Area Network
▶ Ethernet II, Src: 52:1d:4b:d9:1b:08 (52:1d:4b:d9:1b:08), Dst: 52:10:d7:d6:1b:08 (52:10:d7:d6:1b:08)
▶ Internet Protocol Version 4, Src: 10.5.10.254, Dst: 8.8.8.8
▶ Internet Control Message Protocol
```

- IPSEC

```
▶ Frame 6: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits)
▶ Ethernet II, Src: RealtekU_08:a5:9a (52:54:00:08:a5:9a), Dst: RealtekU_1e:8a:ad (52:54:00:1e:8a:ad)
▶ Internet Protocol Version 4, Src: 10.255.2.3, Dst: 10.255.2.1
▶ Encapsulating Security Payload
```

Different frame sizes using different overlay techniques.

Base ICMP ping frame size is 114 bytes.

802.1q and MPLS are the smallest as they sit in front of the original IP header.

The other techniques encapsulate the original IP packet inside of a new IP packet.

# Methods of Implementation

- Option 1 - Migrate server vlan interfaces from  core switch and place them on firewall
  - Quicker to implement
  - May need to migrate ACLs from switch
  - May need to further segment existing subnets
- Option 2 - Create new server subnets on firewall and migrate applications to new subnets
  - Migrating applications to new subnets is a large effort that carries risk (services using IP address versus hostname will break)
  - Will require rulebase updates for IP changes, but will lead to cleaner rulebase
  - Applications can be moved one at a time allowing slow, methodical approach

# Recommendations

- If there are just a few server subnets
  - Option 1, followed by option 2
  - This will allow instant improvement of security posture by getting subnets on the firewall
  - Option 2 can then be implemented over time to continue improving posture
- If there are significant server subnets
  - Option 1
  - If assets are already properly categorized into subnets, migrating the subnets straight to the firewall should be all that is needed
  - Make sure ACLs are properly migrated prior to migrating

# Considerations

- Security and NAT policies will need to be updated to reflect changes to zones
- Load balancers can lead to asymmetric routes and will need to be considered before migrating subnets

# What is Falco?

- A tool to detect configuration issues
- A managed service to assist with fixing them

FALCO

FALCO

# Sample Falco Report

Falco Plus

Summary | Policies | Objects | Network | Device

Device: PA5250-1

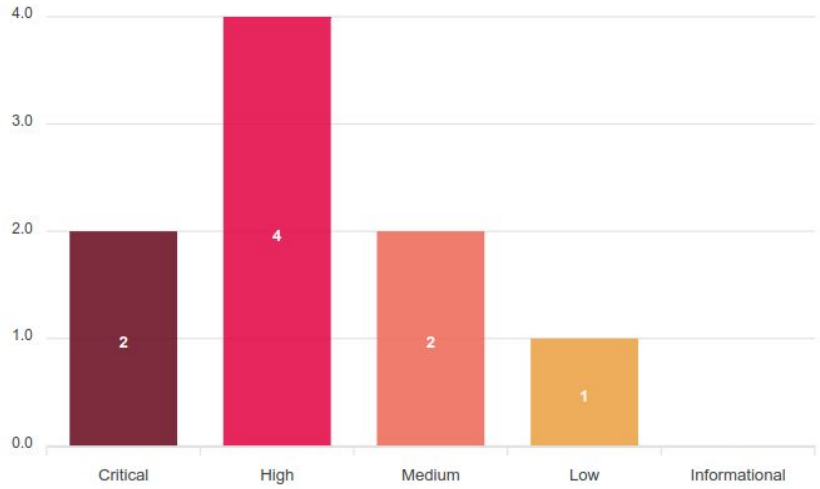
80% passed
1 Devices Audited

1/1 devices
Recommended Releases

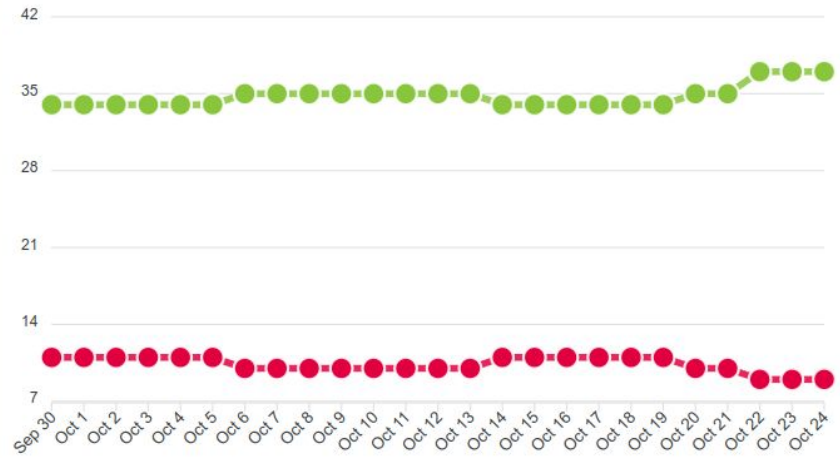
No Vulnerabilities
No Known Vulnerabilties Found

Support Licenses
All Devices Have Valid Support Licenses

## Failed Check Severity




## Report History




sales@digitalscepter.com
(888) 299-3718

digitalscepter.com