Creating a Juniper Lab with EVE-NG - Part 3

The road so far…

Session 1:
Fetched the EVE-NG OS (iso File) (Community and Pro)
Installed EVE-NG on a Bare Metal Server via bootable USB (5min)
Changed initial password (Web-UI) and Install-Wizard (CLI)

Session 2:
Fetched images
Uploaded images (nodes)
Created basic labs
Commit-feature
Traffic-Filters

Current State:
EVE-NG Server (Pro) with Juniper images:
vSRX, vMX, vQFX, vJunOS-Switch

Our Goal

- Look at Clustering / Load Sharing
- Check mixed Environments (Bare, VM, Cloud)
- Talk about Pro Features for Troubleshooting / Testing (Link-Loss, Latency)
- Create a custom Template
- Look at HUGE Labs and how to deal with them

Clustering / Load Sharing

- Clustering allows for scaling / load sharing
- Mix and match what‘s best for your Devices (Bare, VM, Cloud)
- Number of CPU-Cores, HDD-Speed – choose what type of load you have
  vQFX hungry for CPU, JunOS Space Hungry for EVERYTHING, JSA Hungry for Disks
- Add more satellites if needed

What do I need?

- Valid Pro License (Clusters are Pro only)

- Network Ports opened between the master and the satellites

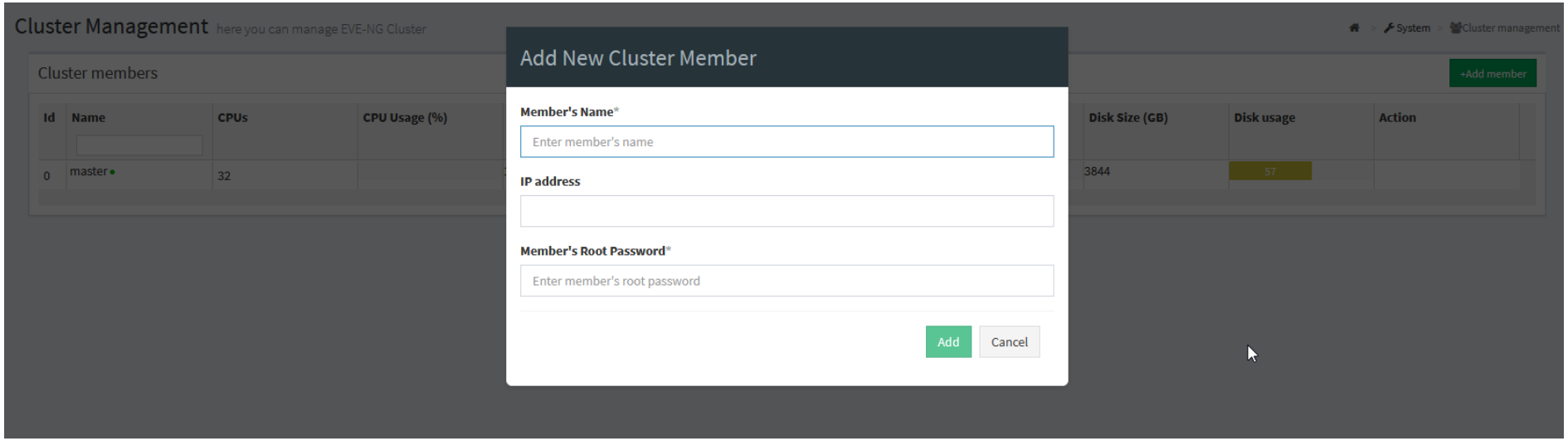
15.2.1 Firewall rules between Master and Satellite nodes

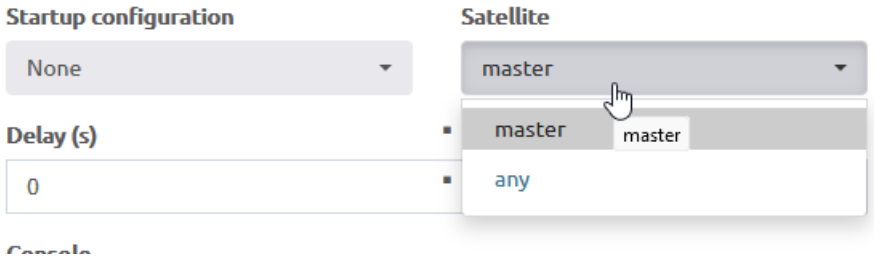
Node	Protocol	Port	Direction	Source	Destination
MASTER	TCP	22	ingress and egress	MASTER node IP	SATELLITE nodes IPs
MASTER	UDP	all	ingress and egress	MASTER node IP	SATELLITE nodes IPs
SATELLITE	TCP	22	ingress and egress	SATELLITE node IP	MASTER Node IP
SATELLITE	UDP	all	ingress and egress	SATELLITE node IP	MASTER Node IP

How to install it?

- Use the iso from eve-ng.net
- Satellite == Agent
- EVE-Cookbook
- Follow every step!

Adding a Satellite





Mixed Environments

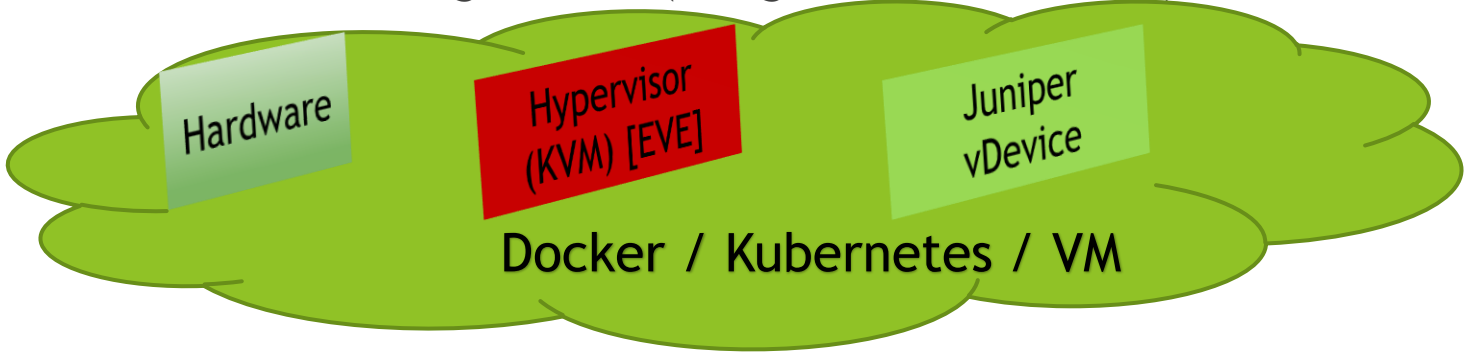
- Ubuntu (EVE) installed on BMS (**MOST PERFORMANT**)

- ESXi installed on BMS and EVE as VM inside ESXi

- Windows Client running VMwareWorkstation, running EVE (EVEception)

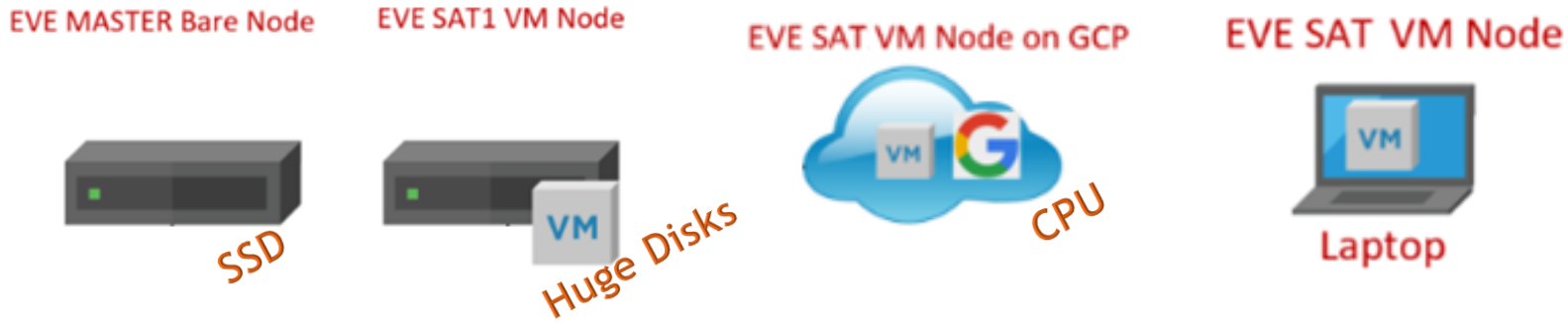
Mixed Environments

- EVE running on GCP (Google Cloud Platform)



Flavors?

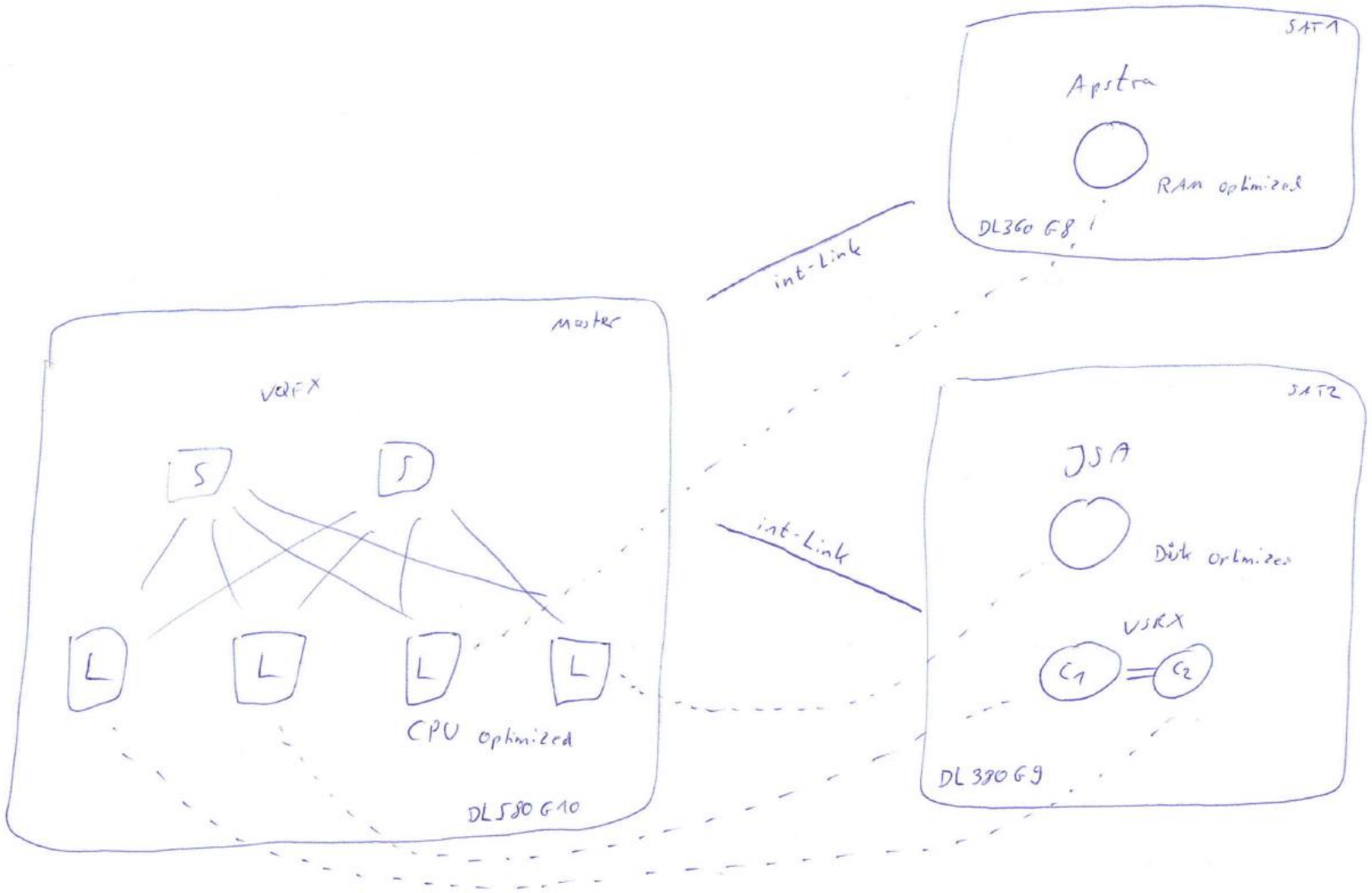
- EVE Hybrid (run multiple flavors of eve in a cluster)



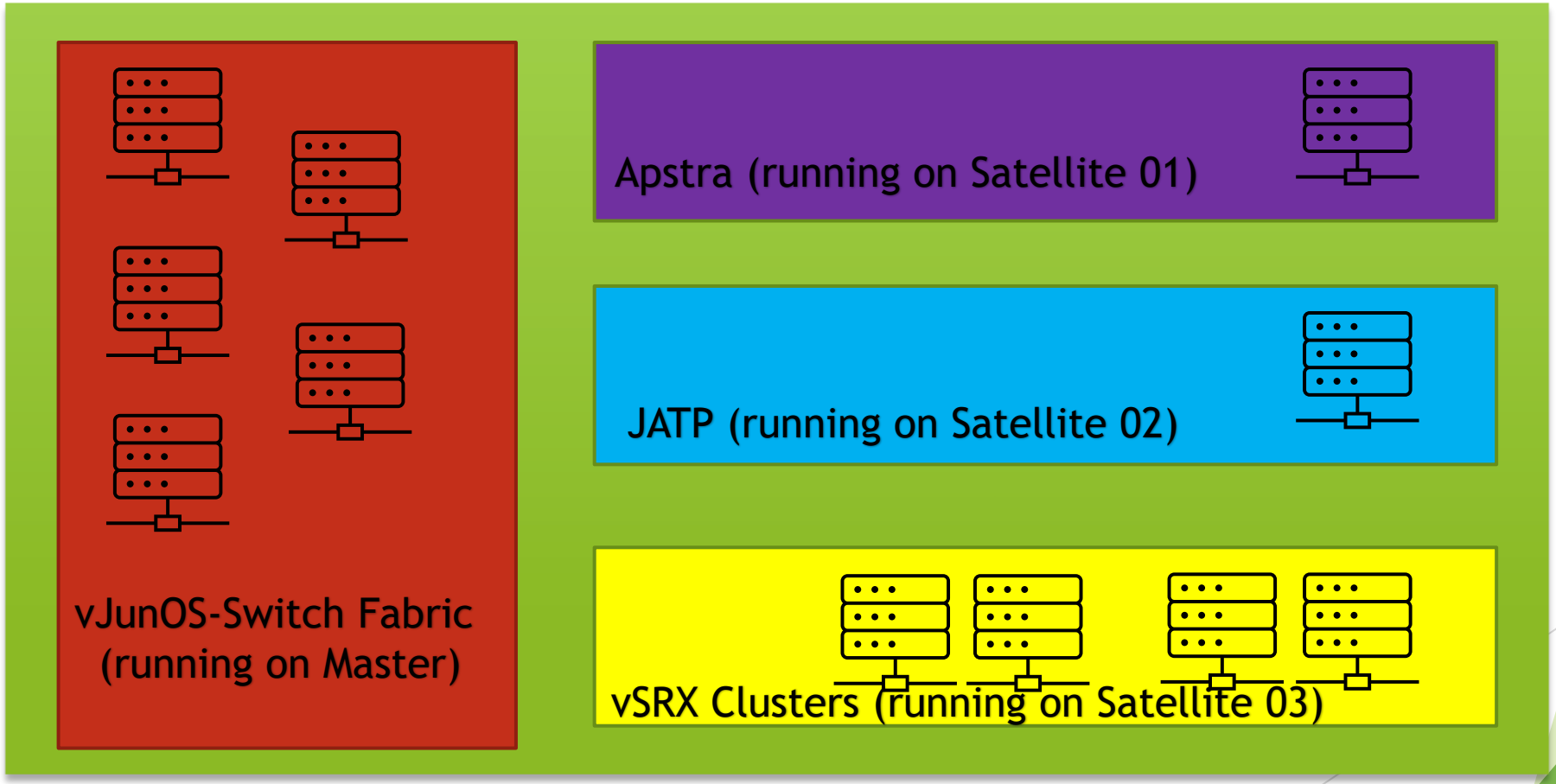
- Allows for easy scaling the Lab (need more resources? Throw more servers in)

- Allows part of your huge nodes to run on different servers (Space, Apstra, JATP)

Load Sharing – Optimizing the Servers

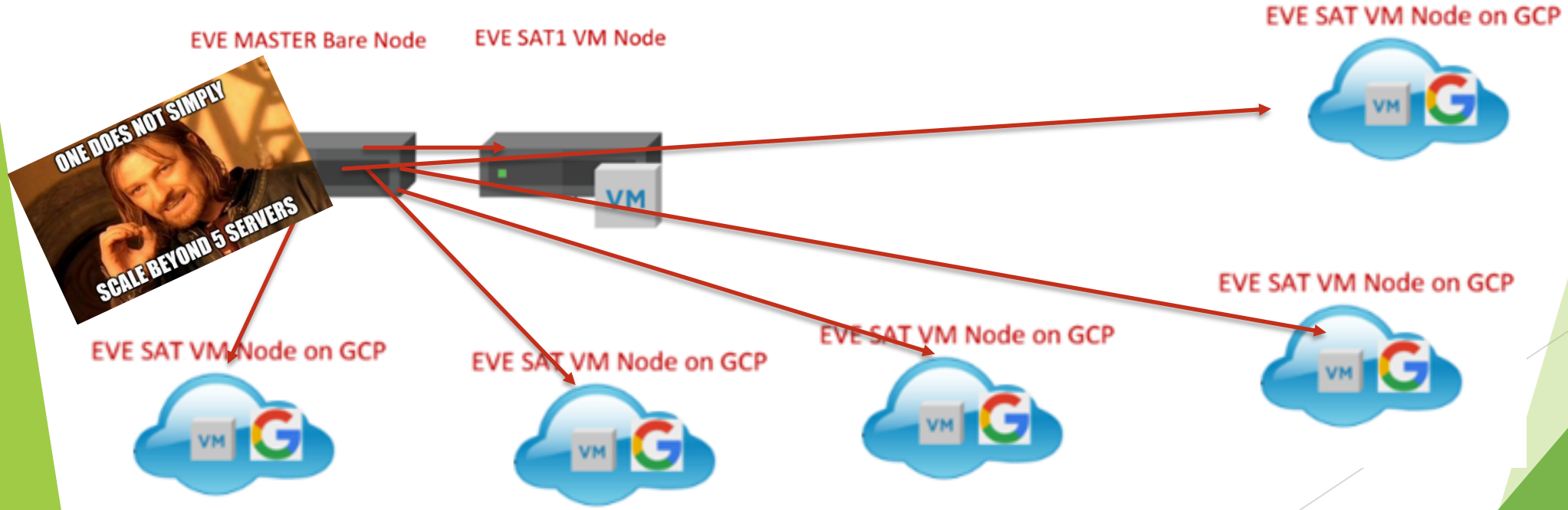


Load Sharing – Optimizing the Servers



How many nodes / satellites?

- Should not go above 5 satellites – master should have „spare resources“
  → Master talks to all satellites for traffic, statistics and such



Our Goal

- Talk about Clustering / Load Sharing ✓
- Look at mixed Environments (Bare, VM, Cloud) ✓
- Talk about Pro Features (Link-Loss, Latency, integrated Wireshark)
- Create a custom Template (vSRX 3.0, Apstra)
- Look at HUGE Labs and how to deal with them

Pro Features – great for everyday usage

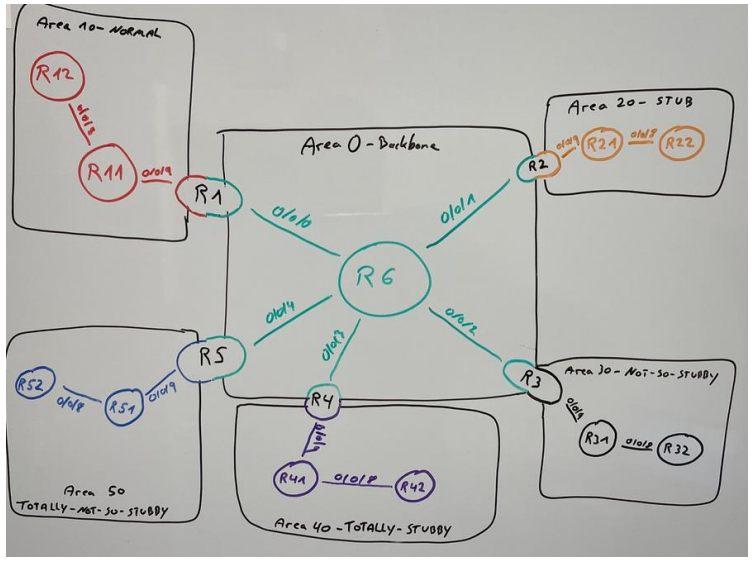
- Hot-Link-Add
- Upload PDF Files
- Packetloss / Jitter / Delay

Link Quality: vMX-VFP29 - vMX-VFP28

Apply | Save | Close

Interface	Delay (ms)	Jitter (ms)	Loss (%)	Rate(kbps)	Interface	Delay (ms)	Jitter (ms)	Loss (%)	Rate(kbps)
ge-0/0/8	0	0	0	0	ge-0/0/8	0	0	0	0

- Logical Maps



Our Goal

- Talk about Clustering / Load Sharing ✓
- Look at mixed Environments (Bare, VM, Cloud) ✓
- Talk about Pro Features (Link-Loss, Latency) ✓
- Create a custom Template (vSRX 4.0)
- Look at HUGE Labs and how to deal with them

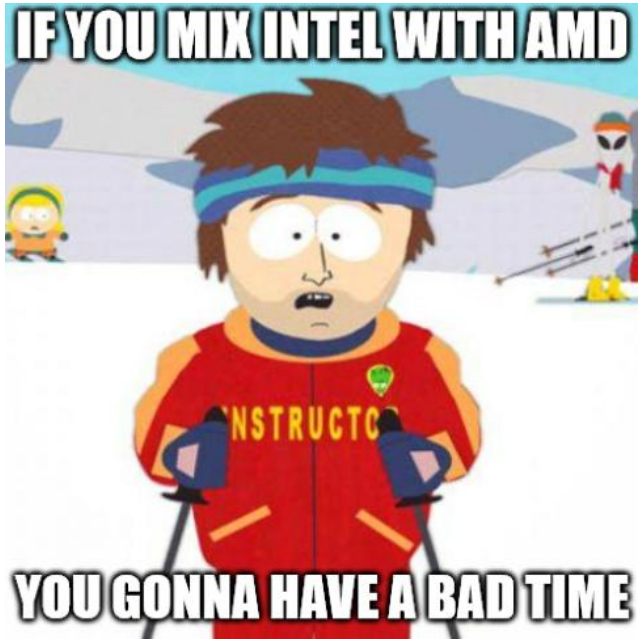
Custom Template

Custom Template

- Modify with sanity and care! Don‘t go „too low“ on Resources
  Weird side effects – not supported – hassle, oh the hassle…
  Always follow the Docs!
- You can copy templates to create a „new“ Device
- For JunOS Space PolicyEnforcer, JSA, Apstra, JATP: Start with Linux Template
- Creating a template alone is not enough!
  also modify the yml for the custom templates!

Custom Template

- /opt/unetlab/html/templates/intel/ (for Intel CPUs)
- /opt/unetlab/html/templates/amd/ (for AMD CPUs)

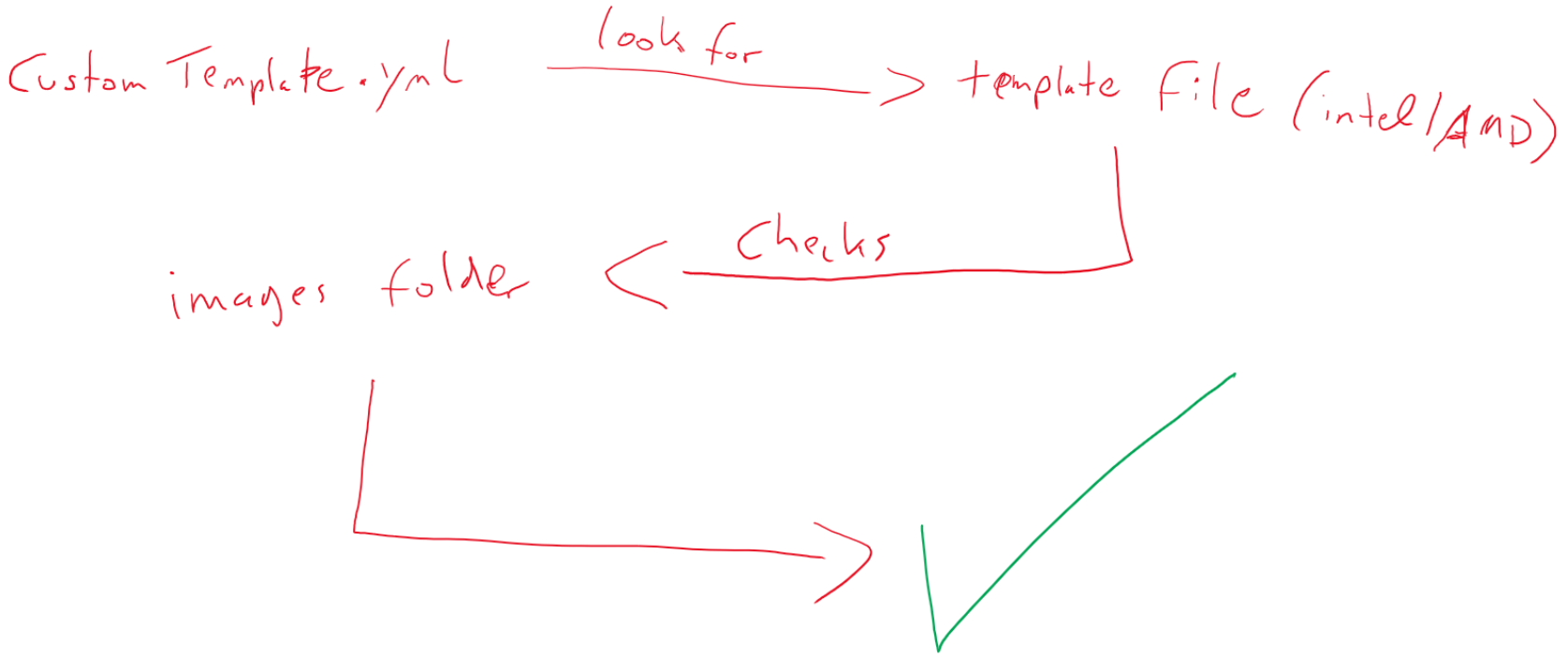


Custom Template - Paths

- /opt/unetlab/html/includes/custom_templates.yml→ your config file

```
root@eve-ng:/opt/unetlab/html/includes# vim custom_templates.yml
---
custom_templates:
- name: JATP
  listname: 'EVE-BETA :: Juniper JATP Appliance (SkyATP on Premise)'
- name: PACKETFENCE
  listname: 'EVE-BETA :: PacketFence OpenSource NAC'
- name: CSRX
  listname: 'EVE-BETA :: Juniper cSRX'
- name: vsrx30
  listname: 'EVE-BETA :: Juniper vSRX3.0'
- name: jspacelogcollector
  listname: 'EVE-BETA :: JunOS Space Log Collector'
- name: riverbedSCM
  listname: 'EVE-BETA :: Riverbed SteelConnect Manager (on Premise)'
- name: riverbedSC
  listname: 'EVE-BETA :: Riverbed SteelConnect VM'
- name: ucopia
  listname: 'EVE-BETA :: Ucopia'
...
~
```

Custom Template - Checks

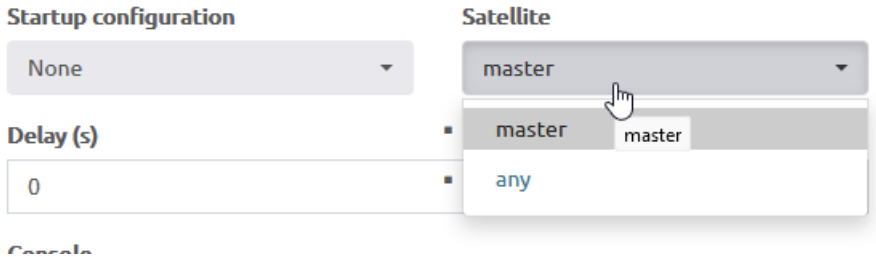


Our Goal

- Talk about Clustering / Load Sharing ✓
- Look at mixed Environments (Bare, VM, Cloud) ✓
- Talk about Pro Features (Link-Loss, Latency) ✓
- Create a custom Template (vSRX 4.0) ✓
- Look at HUGE Labs and how to deal with them

How to handle HUGE Labs

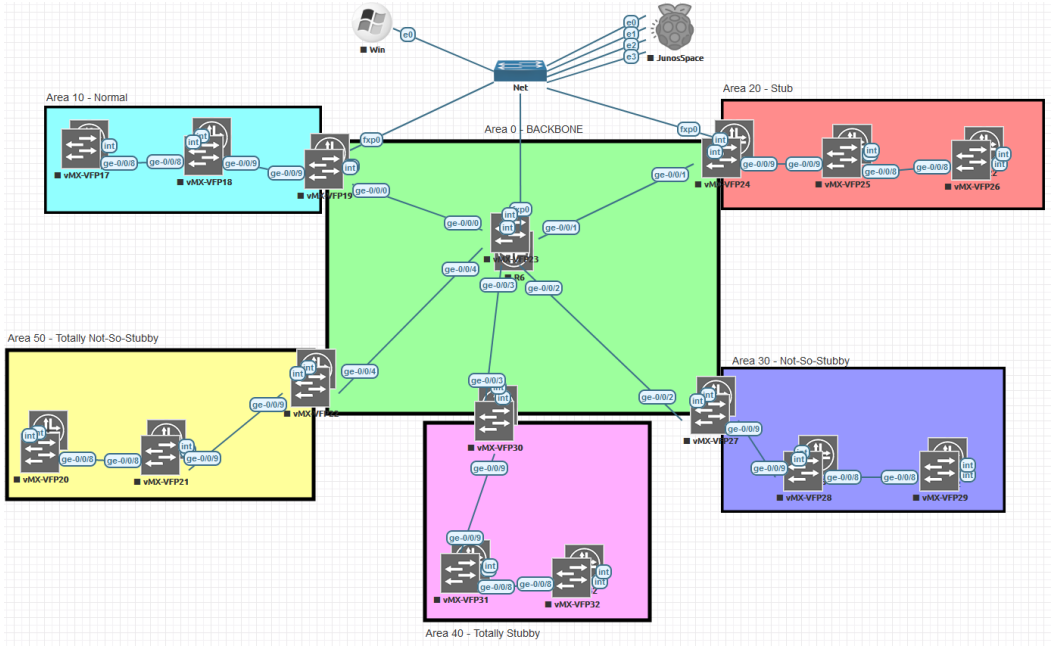
- Delay Start

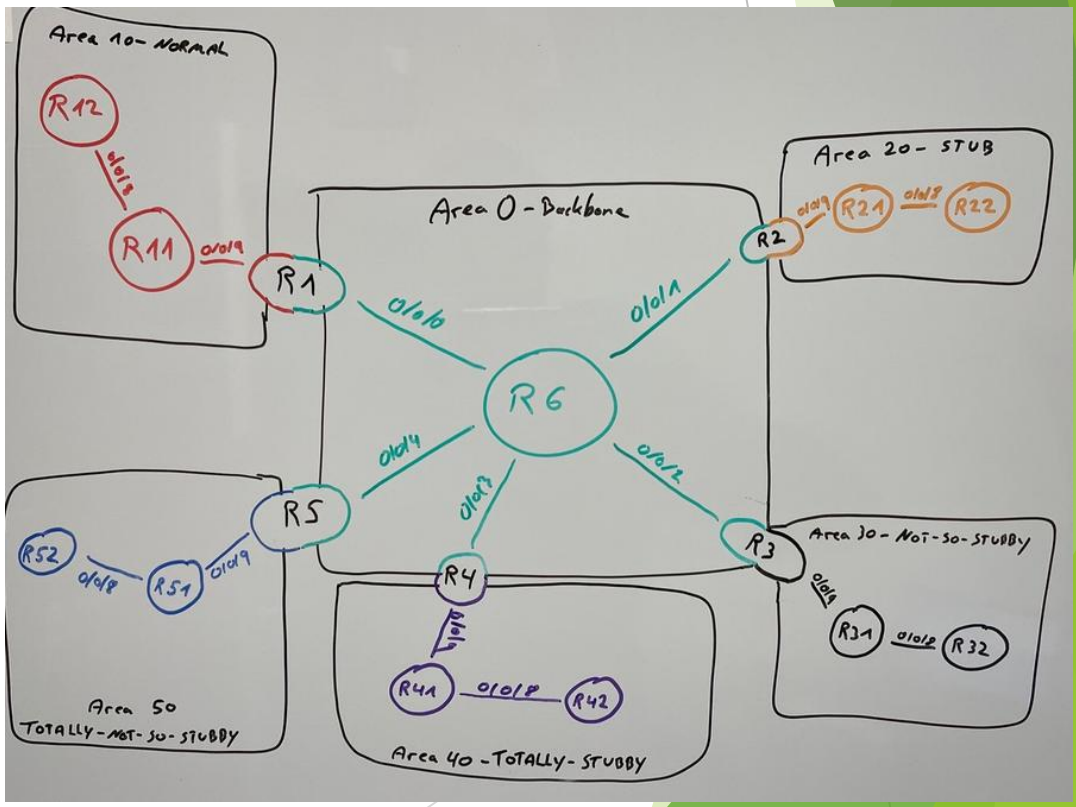


- Use Logical Maps

How to handle HUGE Labs

- **Use Logical Maps**





What to do in case something goes wrong?

Pitfall: not enough resources (CPU / RAM)

- Symptom:
- Multiple unexplainable errors / strange behavior in multiple ways

- Cause:
- Device has not enough CPU / RAM to perform basic tasks

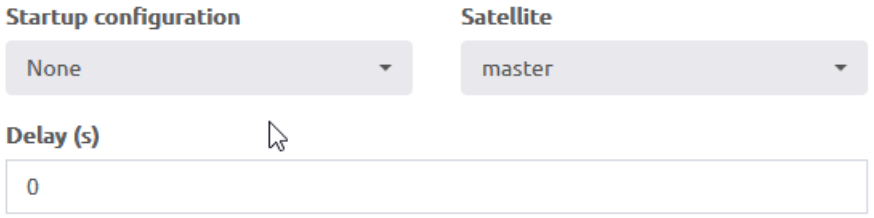
- Solution:
- NEVER go below the recommendation from the template

Pitfall: booting up everything at once

- Symptom:
- Lab takes literally forever to start

- Cause:
- Device takes WAY more resources during bootup and CPU is overwhelmed

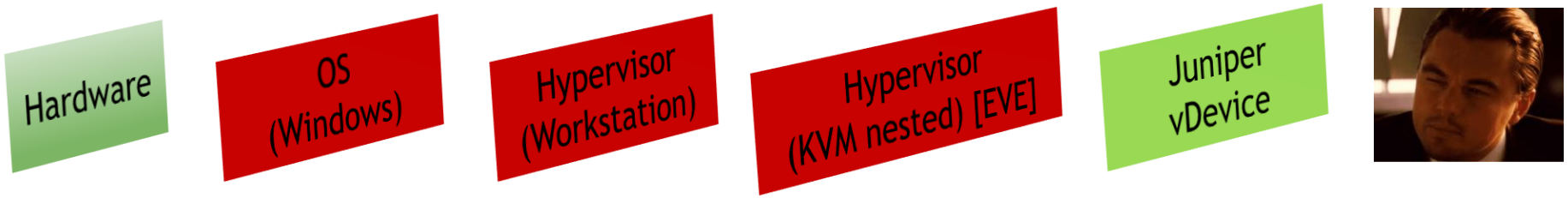
- Solution:
- Use the „delay“-option to start the devices one after another



Pitfall: eve-ception

- Symptom:
- Running EVE-NG on your Laptop in vmware Workstation and starting a lab is not working

- Cause:



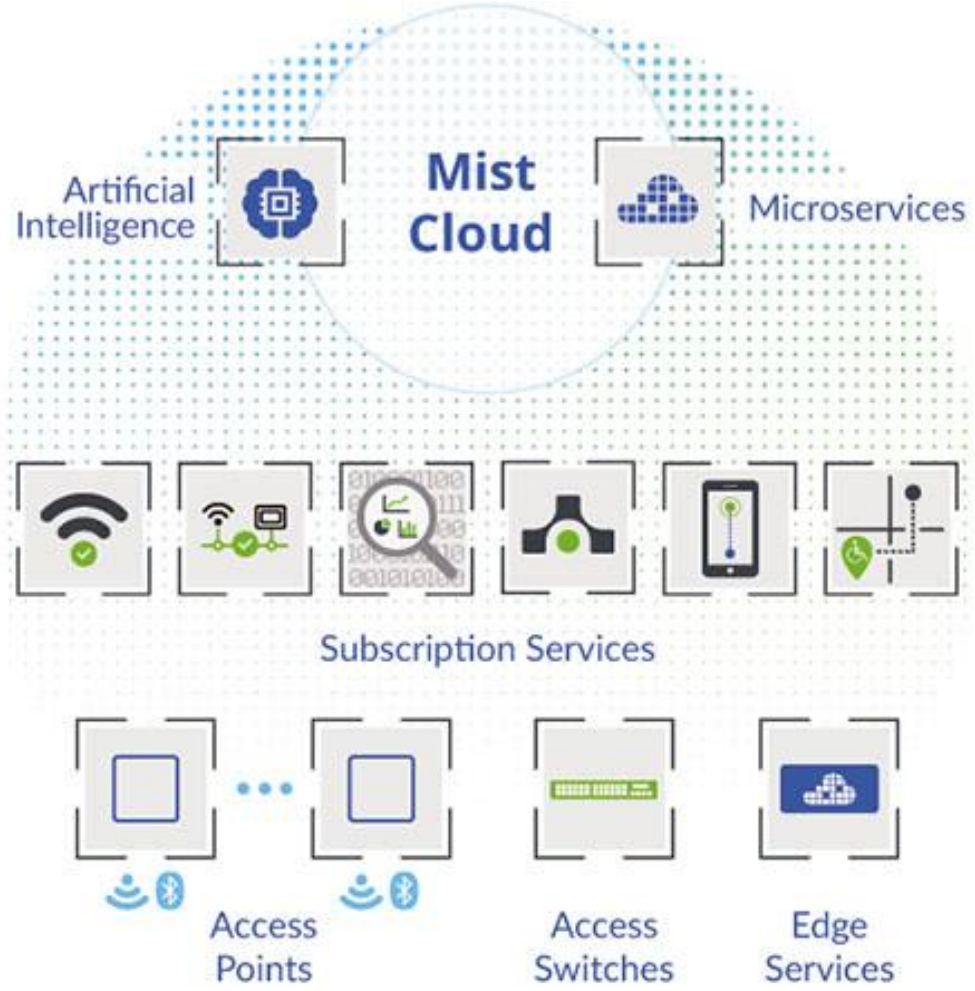
- Solution:
- Don‘t use EVE-NG on your Laptop
  Use a proper Server (Vmware, Cloud or Bare-Metal) and access it via your Webbrowser
  Usually cheap to fetch on eBay

MIST-Edge

- This service lets you make a seamless transition, moving from an existing centralized data plane with legacy controller architectures to the modern Juniper Mist microservices cloud, without affecting network design.
- For large campus networks, Edge provides seamless roaming through on-premises tunnel termination of traffic to and from access points.
- Juniper Mist Edge supports an elastically scalable cluster (with options for backup clusters) composed of an unlimited number of nodes within a cluster.
- https://www.juniper.net/us/en/products/access-points/edge.html

MIST-Edge

- Hardware-Appliance or VM
- Subscription needed
  (1 per AP tunneling Traffic)
- Easy to deploy



MIST-Edge

- Requirements

Hardware Specifications for a Mist Edge Virtual Machine

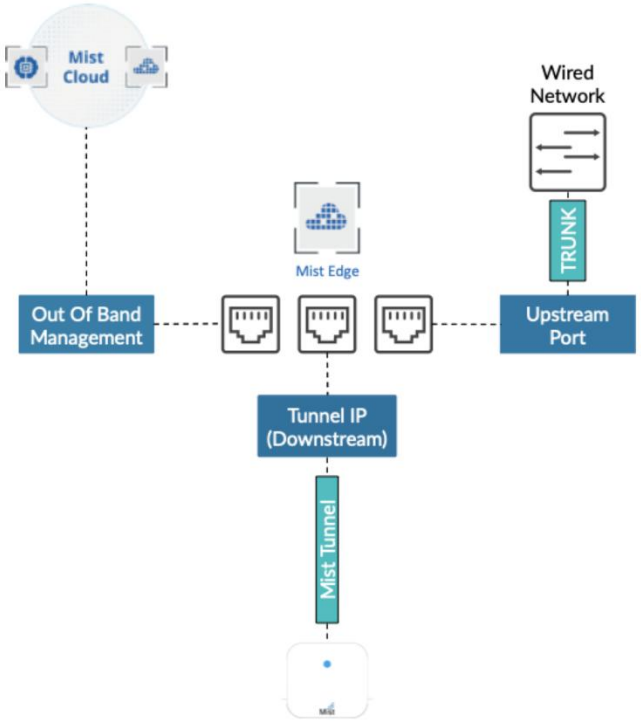
The following are the minimum hardware requirements to implement a Mist Edge VM.

Table 1: Hardware Specifications for a Mist Edge VM

Hardware Component	Quantity or Capacity
CPU	4 vCPUs
RAM	32 GB
Hard disk	100 GB (thick provisioned)
NIC	Three virtual NICs

MIST-Edge

- Create a new Edge-VM in MIST and configure it (including WLAN‘s to MIST-Tunnel mapping)!!!
- Download the iso for MIST-Edge (select VM and download the iso)
- Upload the iso into EVE-NG
- Create a new Linux-based-Template for MIST-Edge with iso mapped as cdrom-file https://www.eve-ng.net/index.php/documentation/howtos/howto-create-own-linux-host-image/
- Deploy one or more Edge-VM‘s in EVE-NG
- Assign Interfaces
- Start labbing ☺

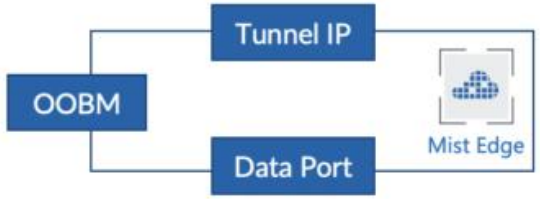


MIST-Edge

- 3 Interfaces: Upstream (Tunnel), Downstream (Switch), OOB
- **Note: Tunnel IP SVI on Mist Edge is a protected interface, so even if it is not connected to a firewall, it is only open for ports UDP: 1701 (L2TPv3), 500 and 4500 (IPsec) and TCP port 2083 for RADSEC.**

Tunnel IP is the interface where AP communicates with to setup the L2TPv3 Tunnel between AP and Mist Edge. This IP needs to be configured from Tunnel IP section on Mist UI. If there is a firewall between AP management subnet and Mist Edge Tunnel IP , traffic destined to Tunnel IP on port 1701 needs to be allowed.

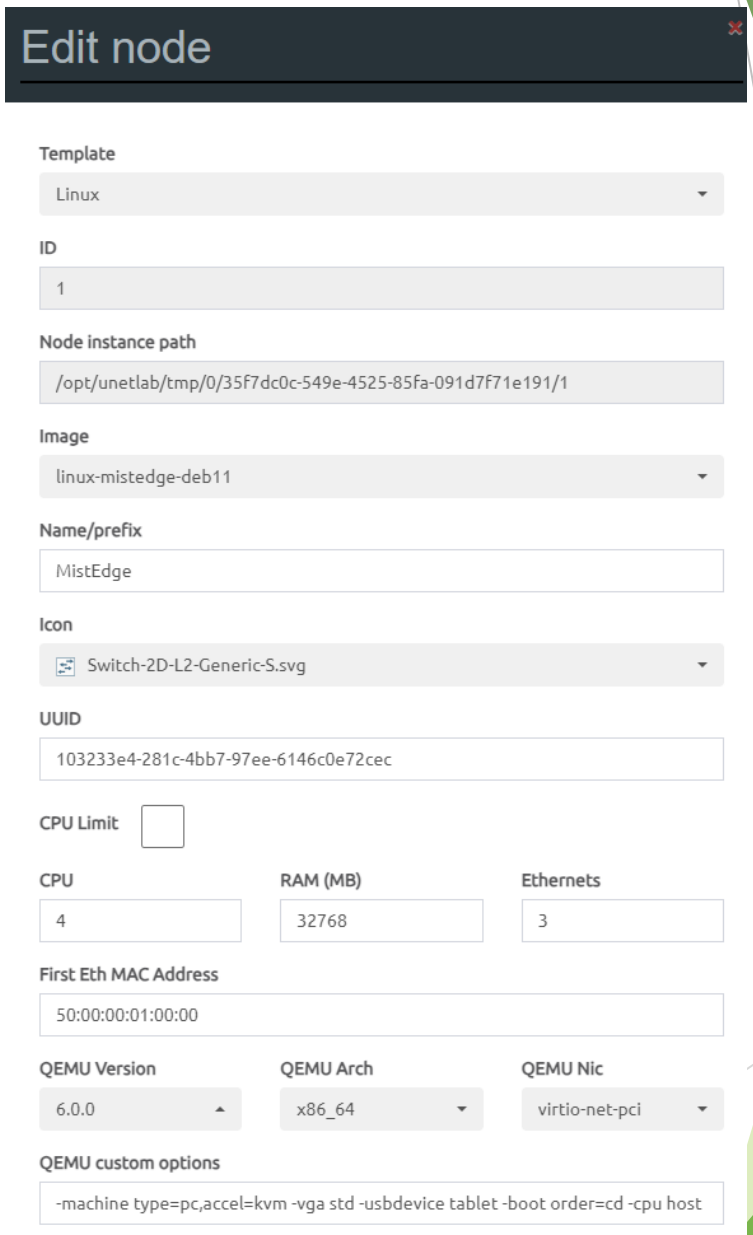
The Out-of-Band-Management (OOBM)
Interface communicates with the Mist cloud and is there to configure, send stats and check status of Mist Edge , Mist Edge Cluster and AP Tunnels.
Interface expects a DHCP IP address by default and can be configured with static IP address



Data Port is connected to a trunk port that has all the VLANs configured where the WLAN need to be mapped to

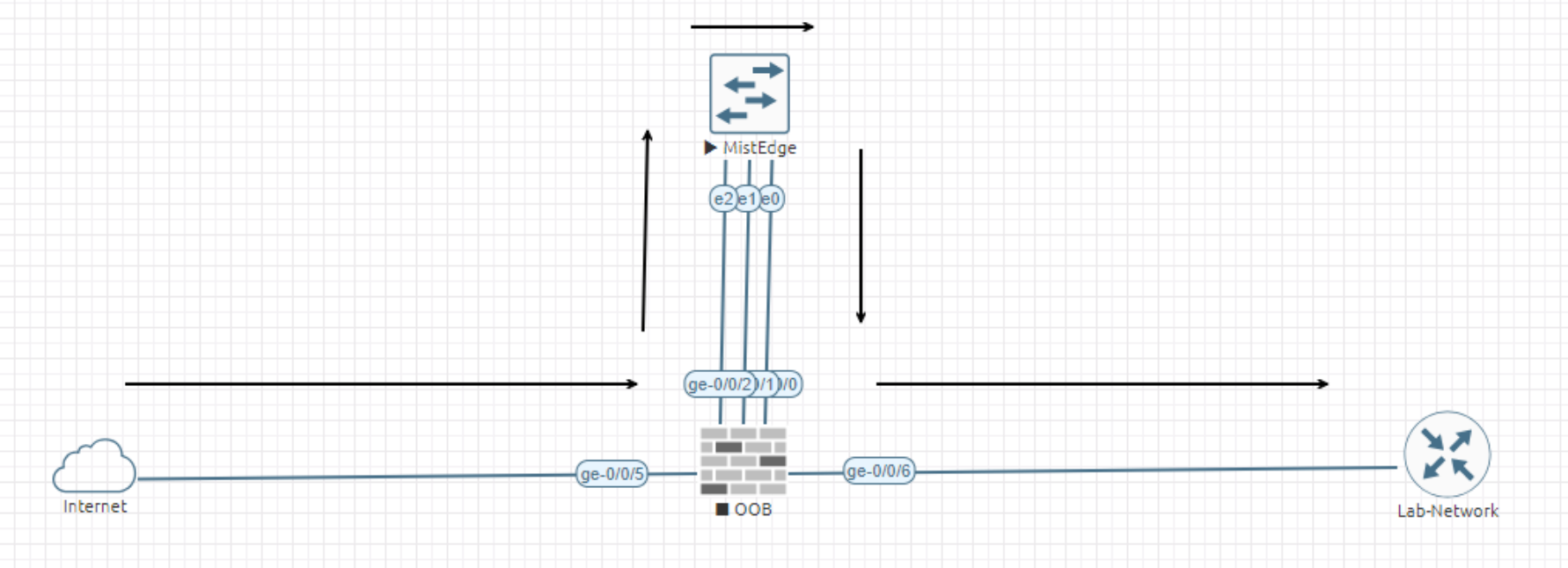
MIST-Edge

- Use the Linux-Template for Mist-Edge
- Official Template will follow soon
- /opt/qemu/bin/qemu-img create -f qcow2 hda.qcow2 100G
- **DO NOT USE VIRTIOA.QCOW2!** The Installer will FAIL!



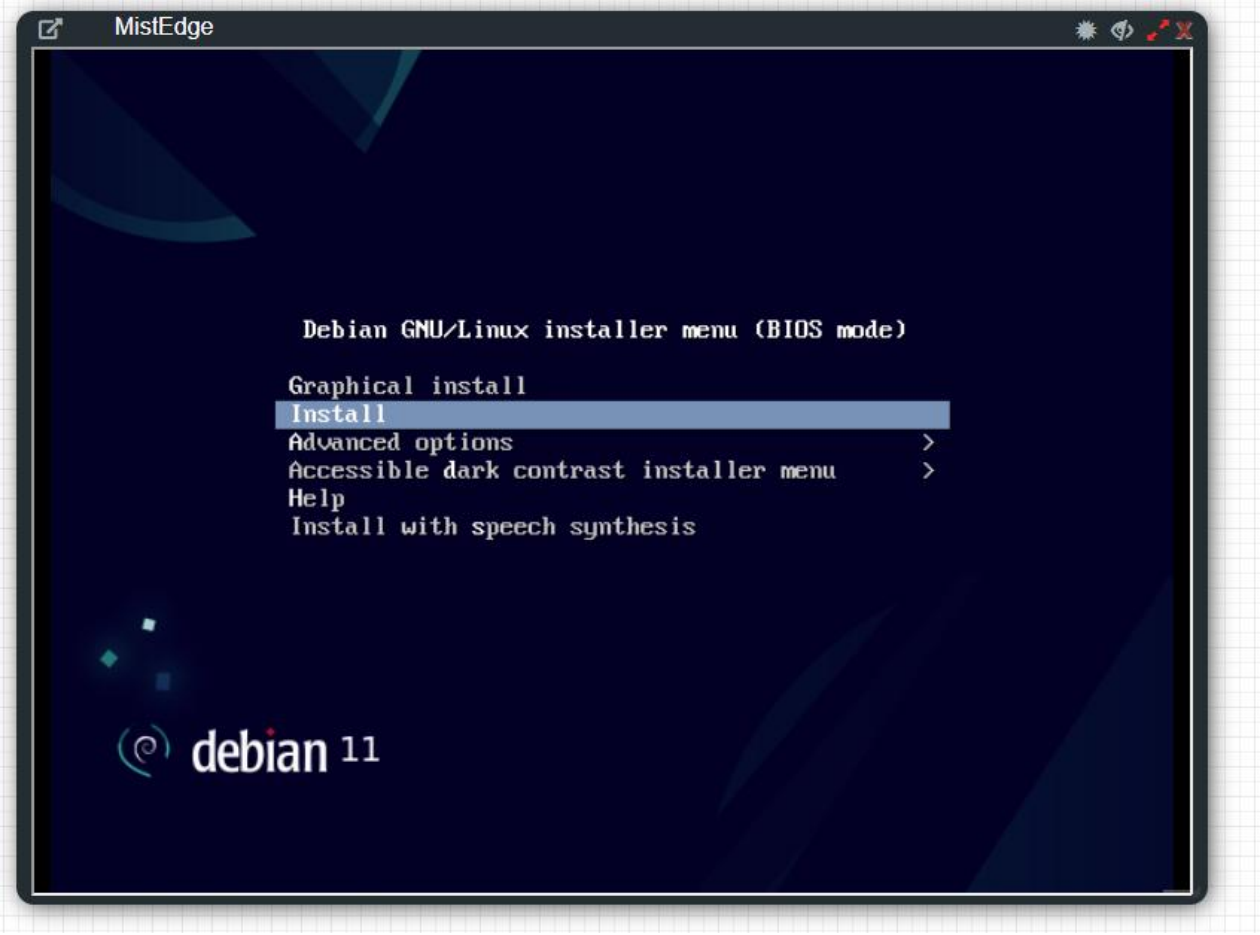
MIST-Edge

- Create a Topology utilizing all 3 Interfaces
- This can be 3 separate links to 3 separate devices
- Or it can be 3 Links towards a central firewall in your lab-environment



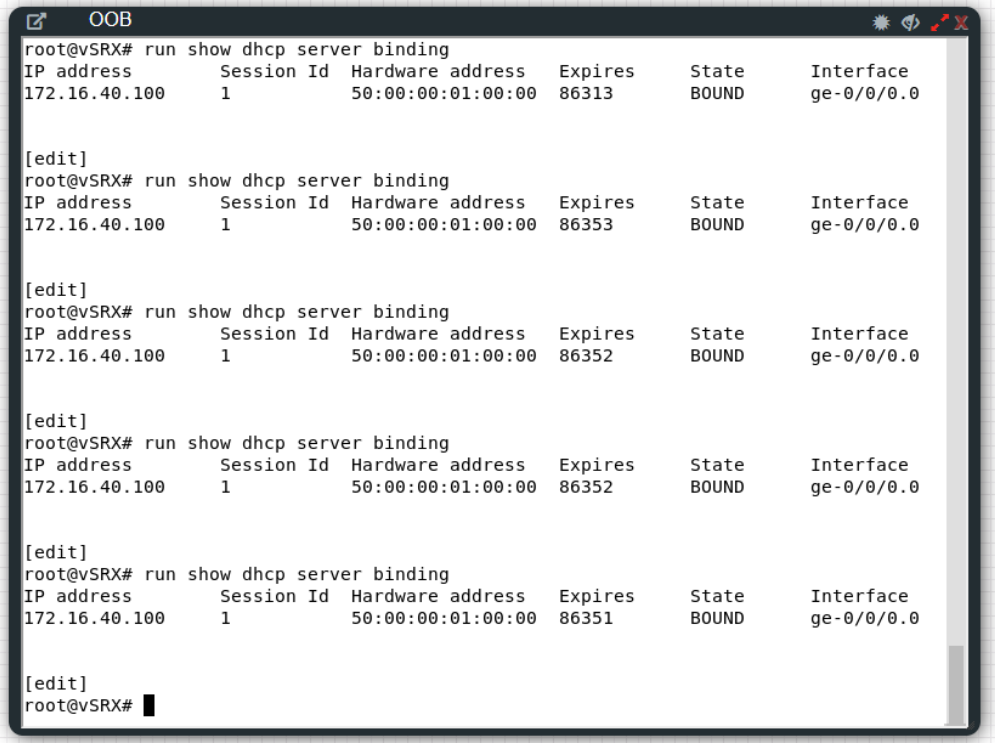
MIST-Edge

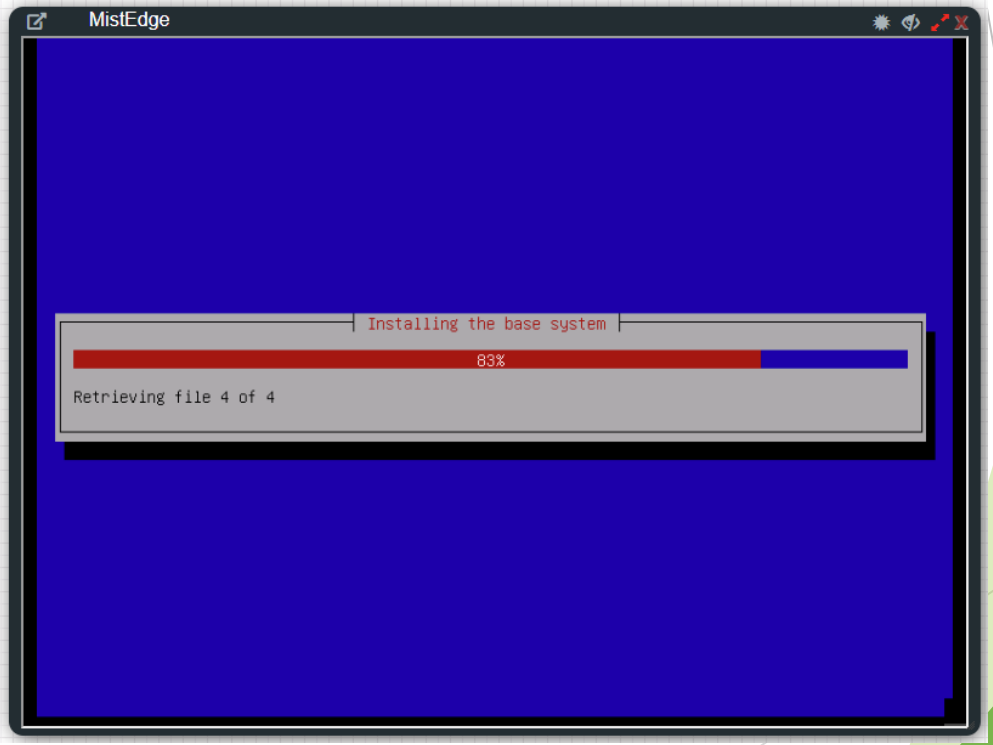
- Install MIST-Edge by selecting „Install“ (NOT Graphical install)!
- OOB-Interface grabs a DHCP-Address by default



MIST-Edge

- The MIST-Edge will take care of everything for you – lean back and enjoy ☺





MIST-Edge

- ssh into the Edge (from your Lab / Firewall)

```
172.16.40.101     2           50:00:00:01:00:00  86261       BOUND      ge-0/0/0.0


[edit]
root@vSRX# run ssh mist@172.16.40.101
The authenticity of host '172.16.40.101 (172.16.40.101)' can't be established.
ECDSA key fingerprint is SHA256:kd9bitkqfdtzIvajDsoUw2py15cluPrJA+RxkmT9n+E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.16.40.101' (ECDSA) to the list of known hosts.
mist@172.16.40.101's password:
Linux mxedge 5.10.0-26-amd64 #1 SMP Debian 5.10.197-1 (2023-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
sourced /etc/skel/.mxagent_aliases
mist@mxedge:~$
```

MIST-Edge

- JNPR MIST-Edge Docs: https://www.juniper.net/documentation/us/en/software/mist/mist-edge-virtual-solution/mist-edge/topics/topic-map/vm-deployment-example.html

Use SSH to connect to the Juniper Mist Edge with the username **mist**
`ssh mist@`*`OOBM-IP`*. Enter `Mist@1234` as the password.

Switch to root by issuing the command `su-`. Enter `mist` as the password.

To bootstrap the device and onboard it to the Mist Cloud, issue the following commands from CLI:
`mist@mxedge:~$ su - Password: mist root@mxedge:~# apt-get update`

After the update, register the device. Enter the command `mxagent-helper configure ----claim-code` *`REGISTRATION CODE`*.
At the end of the process, you see the following message:

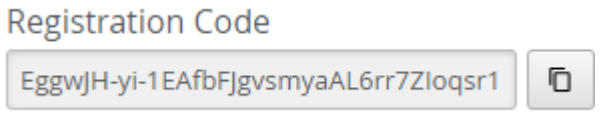
```
registration finished successfully. (regfile at /var/lib/mxagent/mxagent.reg
```

After the process is complete, the Juniper Mist Edge reboots automatically. At this point, you do not need SSH to connect to the Juniper Mist Edge. The device pulls the configuration from the Juniper Mist cloud.
After the reboot, the Juniper Mist Edge appears as connected on the Mist Edge Inventory page. An orange dot also indicates the connected status of the device.

MIST-Edge

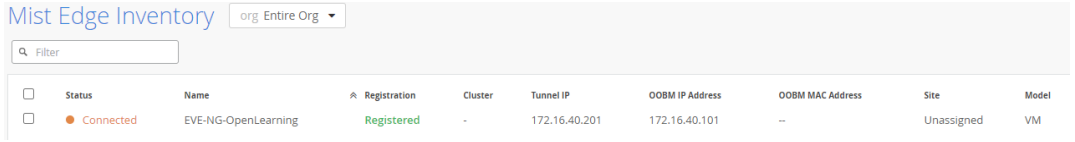
- JNPR MIST-Edge Docs: https://www.juniper.net/documentation/us/en/software/mist/mist-edge-virtual-solution/mist-edge/topics/topic-map/vm-deployment-example.html

- mxagent register -- registration-code <CODE>

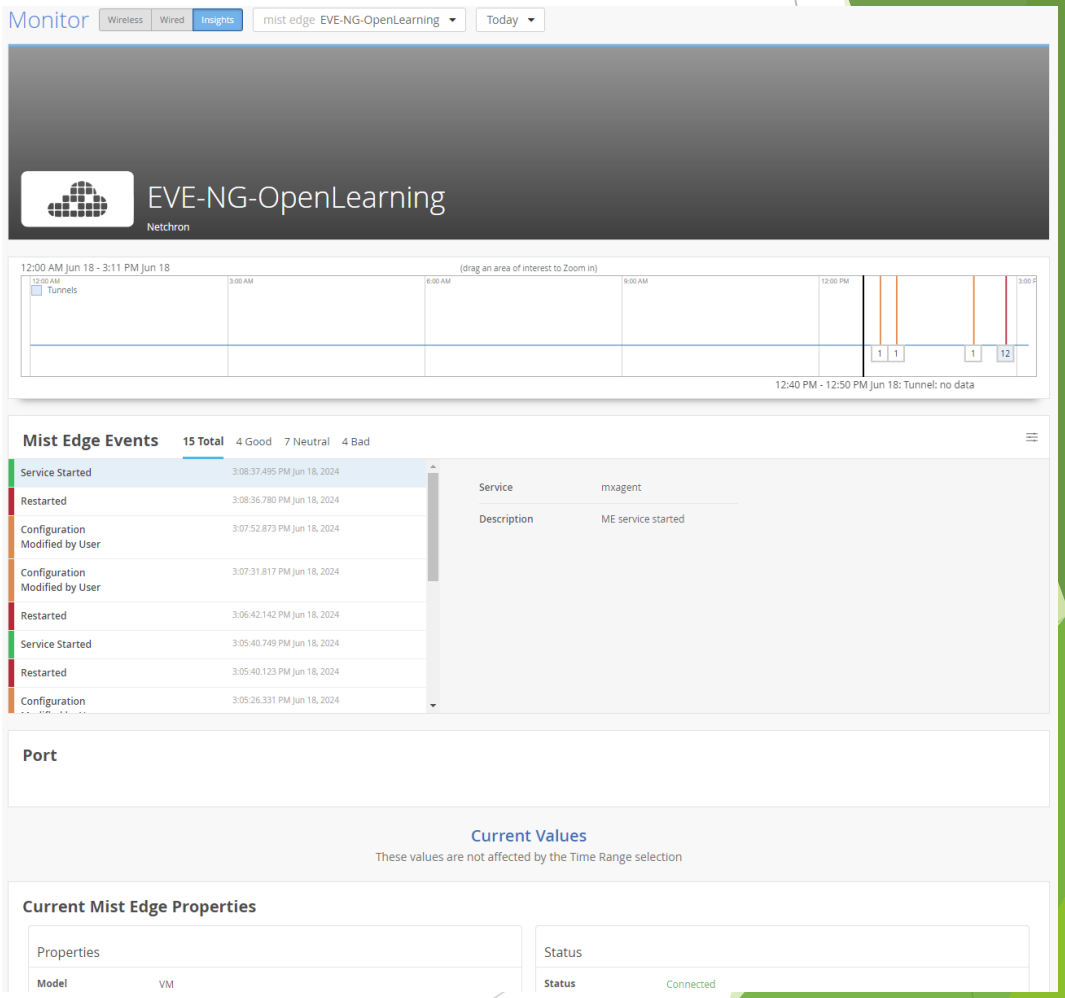


- Wait 5min, then Reboot once

MIST-Edge

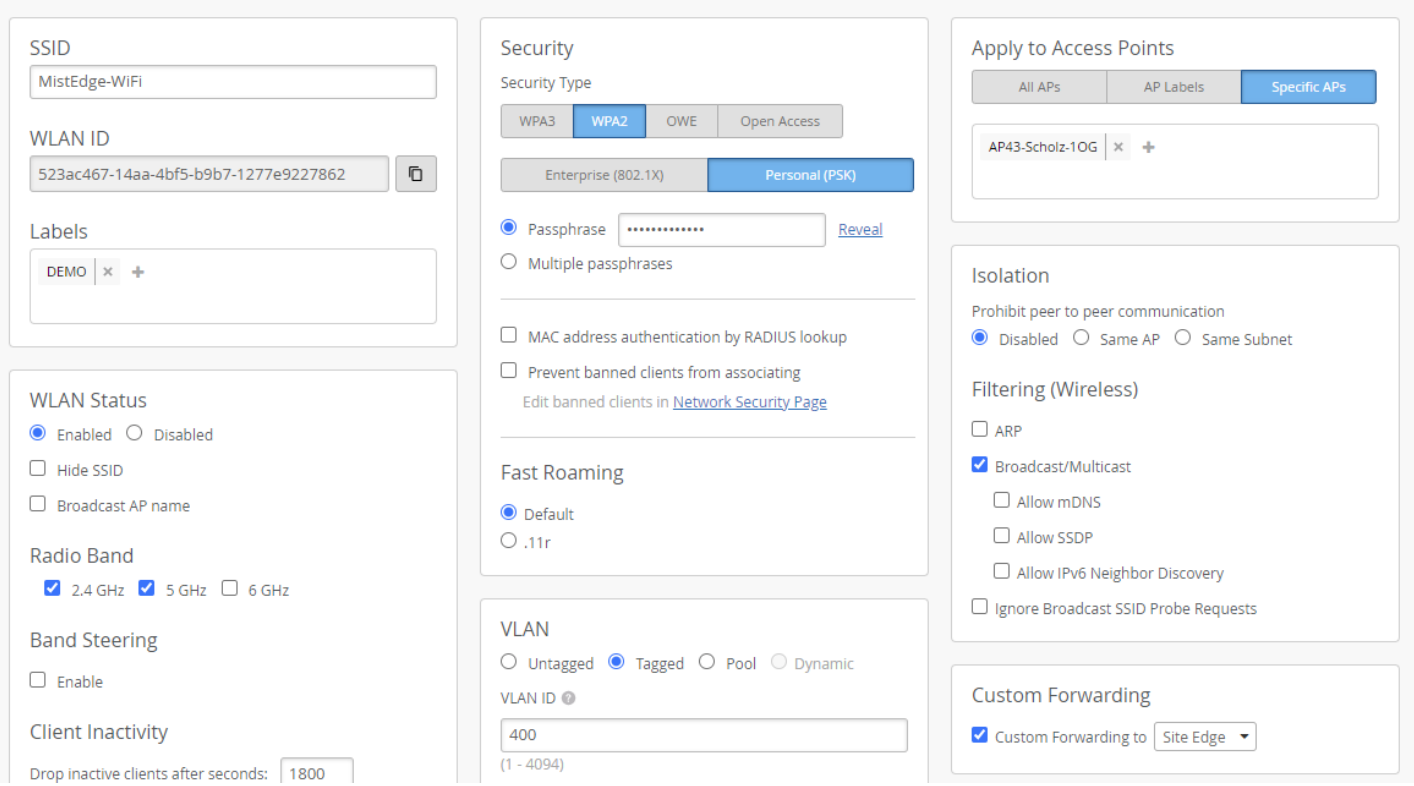


- The Edge will be shown as registered and „connected“
- Assign your Edge to a Site ;)
- Your AP‘s can now connect to it (Template)



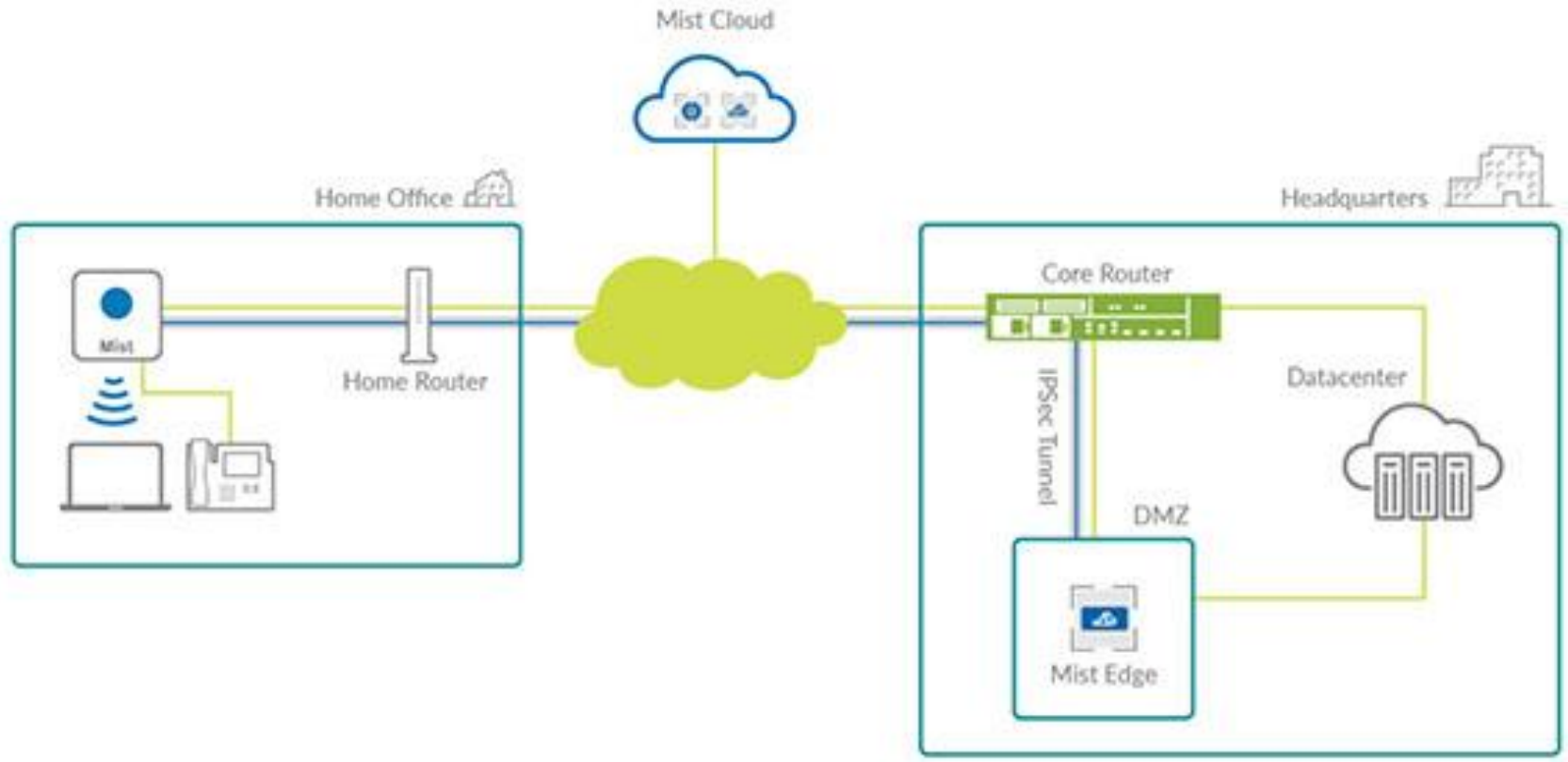
MIST-Edge

- Custom Forwarding to „Site Edge“



MIST-Edge

- Congrats – your Lab can now be accessed via your Home-WiFi ☺



Q&A

Christian Scholz (Social: @Netzwerkonkel)

„That guy who loves Networking,
EVE-NG and IPv6“

Teamlead CC-NIA | Lead Architect
[Axians Networks & Solutions - Germany]

Juniper Ambassador | Juniper MistFit
JNCIE-SEC #374 | JNCIE-ENT #947

Blog: netzwerkonkel.eu (jncie.eu)
Mail: chs@ip4.de