

CNaaS deployment and pScheduler DotX plugin development

Magnus Bergroth
Kristofer Halin
SUNET

2021-04-15 2nd European perfSONAR Users Workshop



Campus Network As A Service

Request from university

- Secure network knowledge
- Generation shift in equipment and personal
- Joint procurement

Current users

- Mälardalens University, in production 2020-02
- Stockholm University, in production 2021-04

More in the pipeline

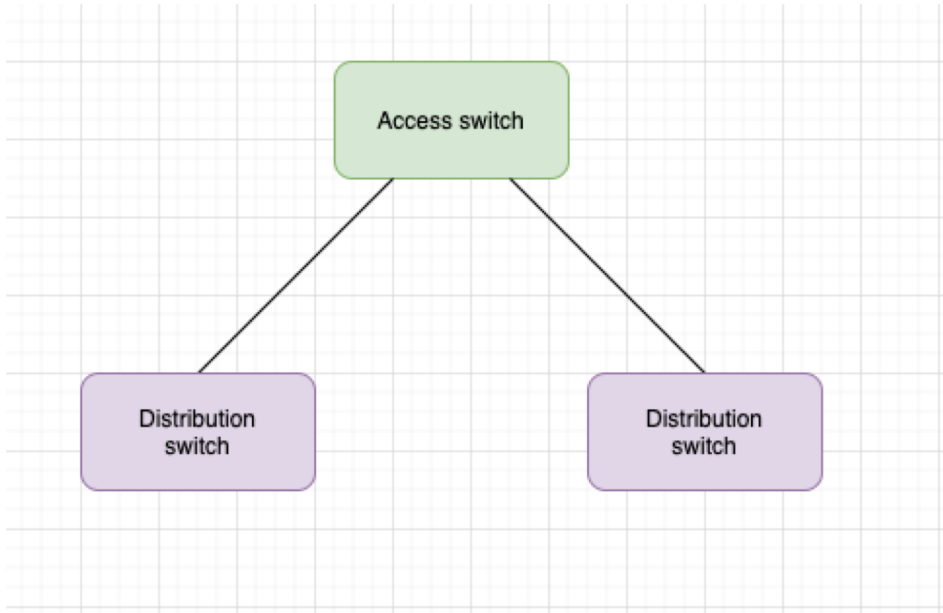


work simplification

- Build on today's network best practice, Data Center Leaf Spine
- Use only industry standards, no vendor lock in
- Automation in network deployment
- Automation in network operation
- Automation in network documentation
- Automation in network monitoring



Key drivers



- Open standards
- All links active
- Well proven technology
- No brand lock in
- No rush to fix broken hardware

Resulted in

- VXLAN
- BGP
- EVPN
- EVPN/ESI multi-homeing

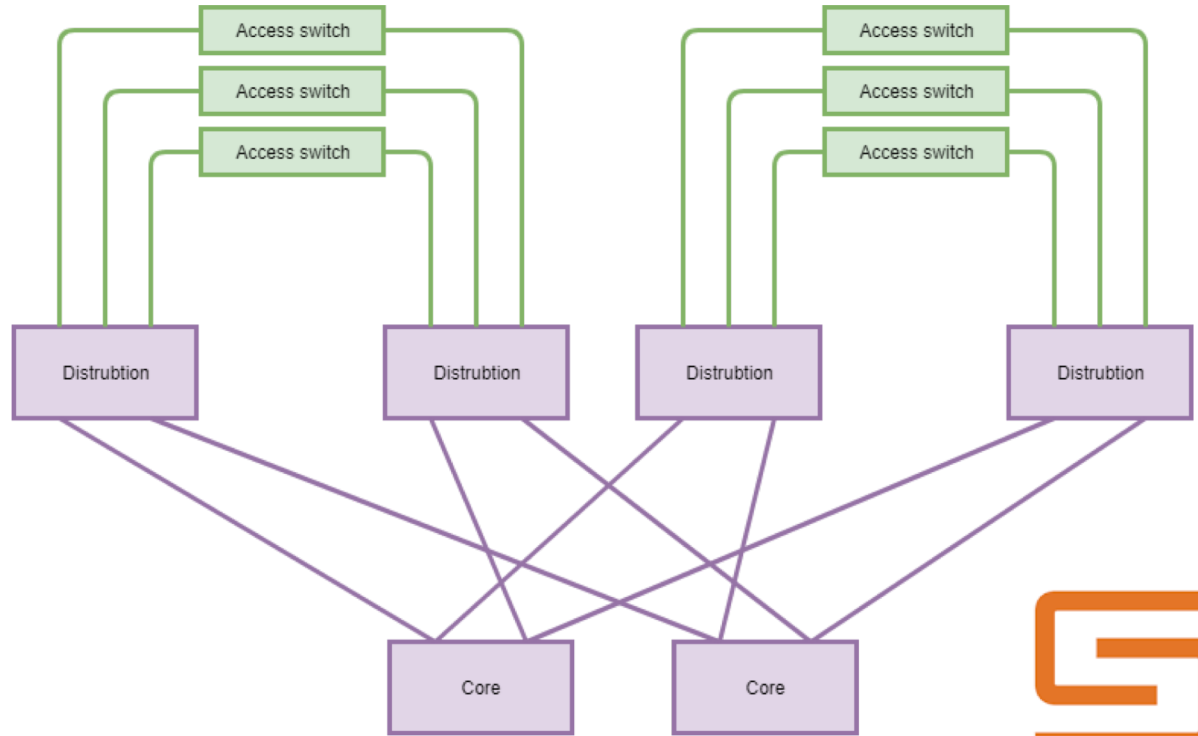
Excludes
Multichassi lag
Spanningtree
Virtual-chassis

Reference Network Architecture SUNET CNaaS

Utilizing best practice DC leaf-spine architecture

Redundant except access ports

IP only core

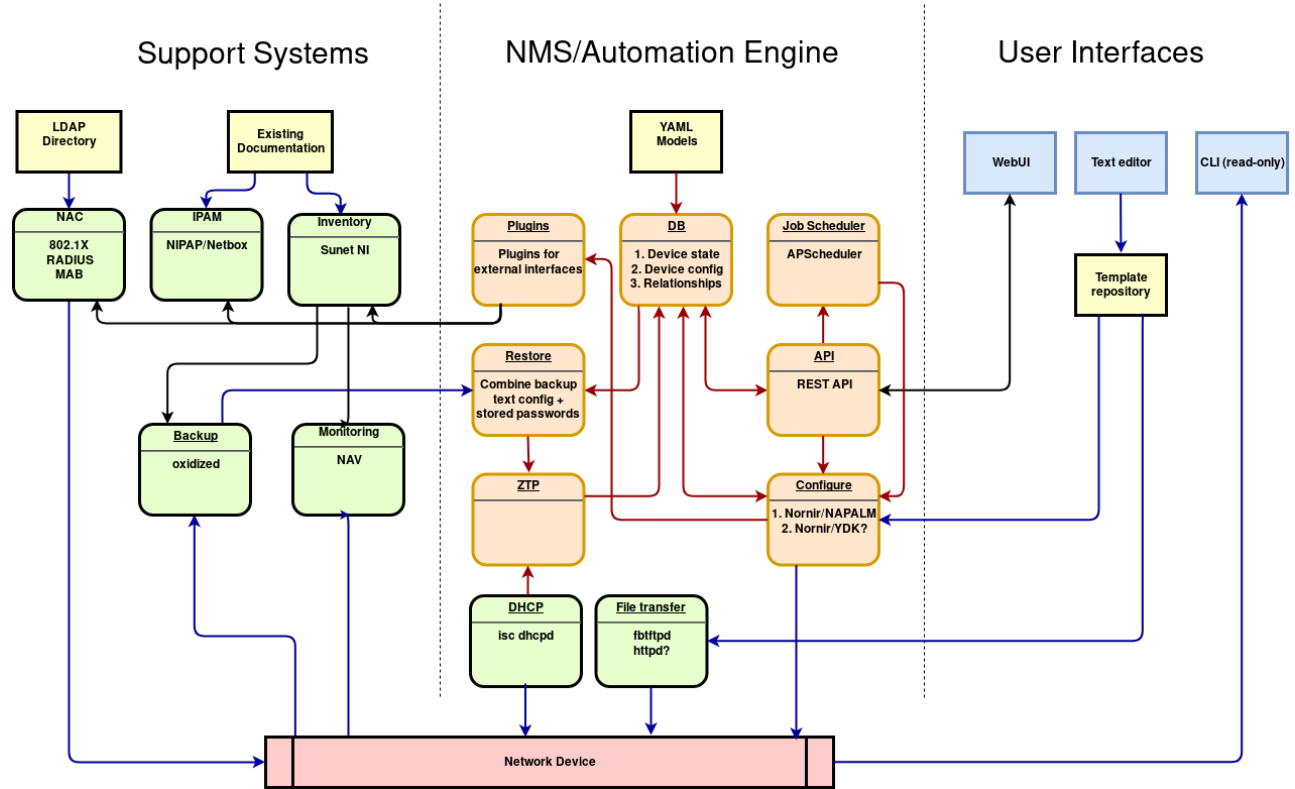


CNaaS-NMS Overview

CNaaS-NMS is open source and everything including source code, documentation etc is available to the public on Github

CNaaS-NMS is a hybrid infrastructure-as-code (IaC) and API driven automation system

The components of CNaaS-NMS are executed in separate Docker containers.



SUNET

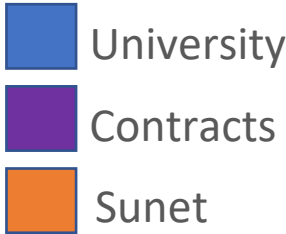
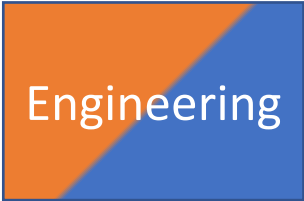
Service delivery - collaborative service



1st Line
End user support
Simple mgmt



2nd Line
Monitoring
Documentation
Incident/Problem Mgmt
Configuration Mgmt
Escalation/(Vendor mgt)



Some problems: during deployment at first Campus

- Switch came up with no access to Radius server. Put ports into fail-vlan. Stuck ip-address.
- Radius Server ran out of disk space.
- eduroam on fixed network ports. Radius chain broken.
- Large radius certificates, jumbo frames
- Firewall filters deny and allow
- DHCP server running out of scope

Monitoring: verify that the network is performing

- Verifying access to the network
 - Dot1x, user + password, local account and from eduroam other provider
 - Dot1x, certificates
 - MAB, login with mac-address
 - Fail vlan
- Vlan assignment
 - User gets to correct VLAN
 - Gets IP-address from correct vlan pool.

Monitoring: verify that the network is performing

- Network access
 - Verify DNS
 - Verify access to public services
 - When connected to “Passage system” VLAN, public services should not be allowed
 - Verify access to local services
 - When connected to Student network access to economy servers should be restricted
- Dot1x, user + password, local account and from eduroam other provider

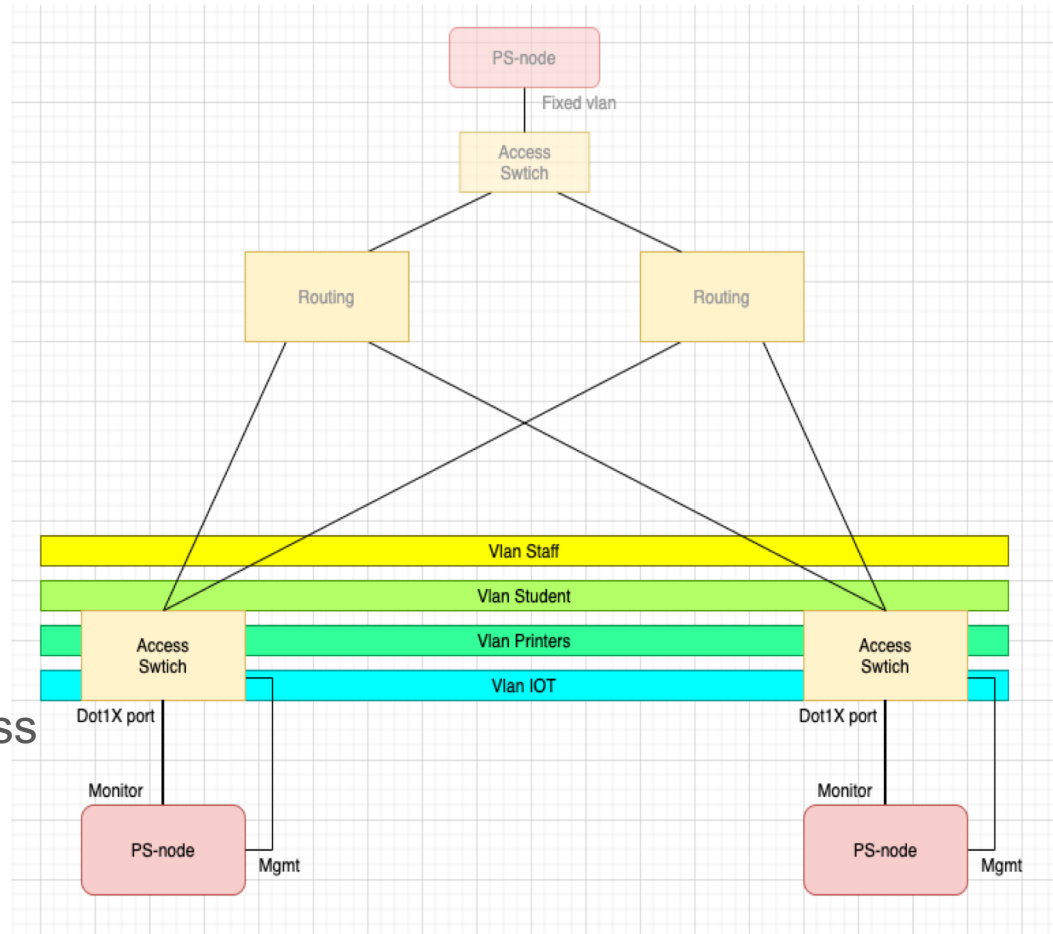


PF tests

- Request node to authenticate to VLAN (test in it self)
- Run test, twamp, traceroute, ping
- Test Multicast 😊

Ip-address changes depending on VLAN.

Requires Mgmt port with fixed address
probe – probe req synchronization
Use fixed probe as reference point



Probe hardware requirements

- No power adapter,
- Remote power cycle
- power via POE
- Dual NICs
- Rack mountable
- No Fans
- Run Perfsonar
- Cheap, many probes



Raspberry Pi

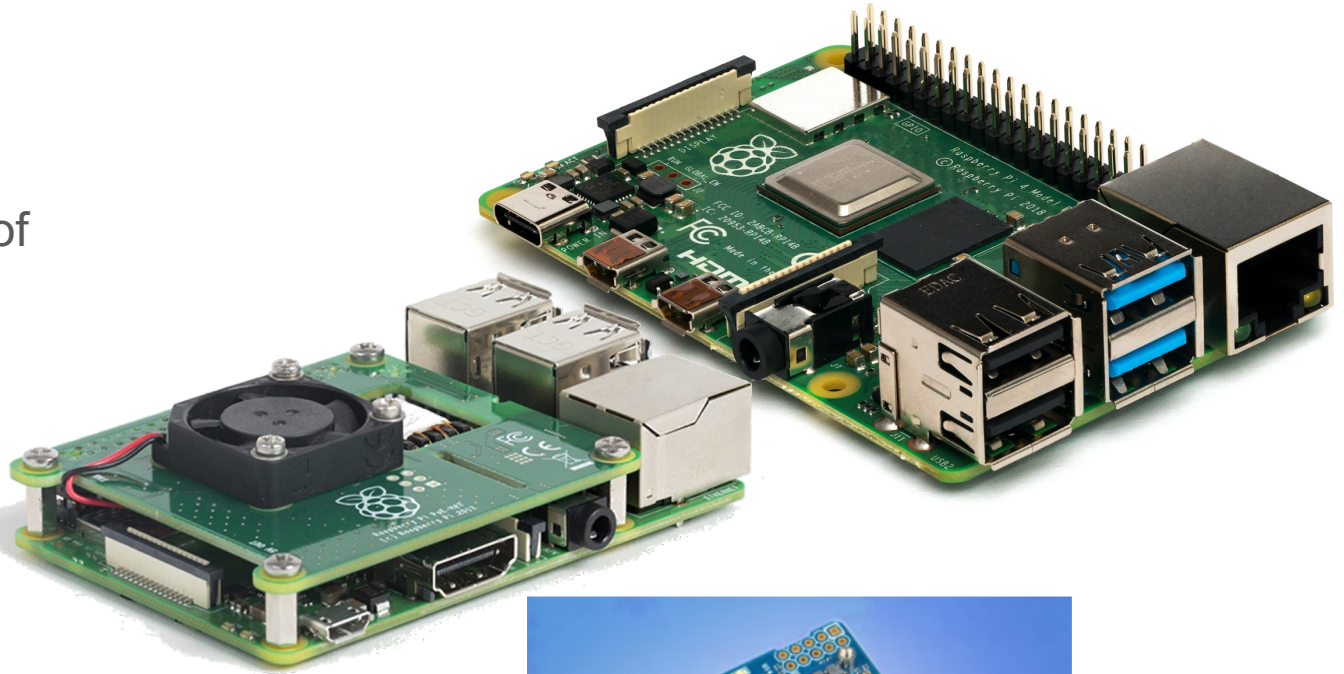
Does the job, sort of

- one NIC
- no PoE

POE Hat has a fan

NanoPi R2S

no Poe
not that available



Parts Cnaas Probe

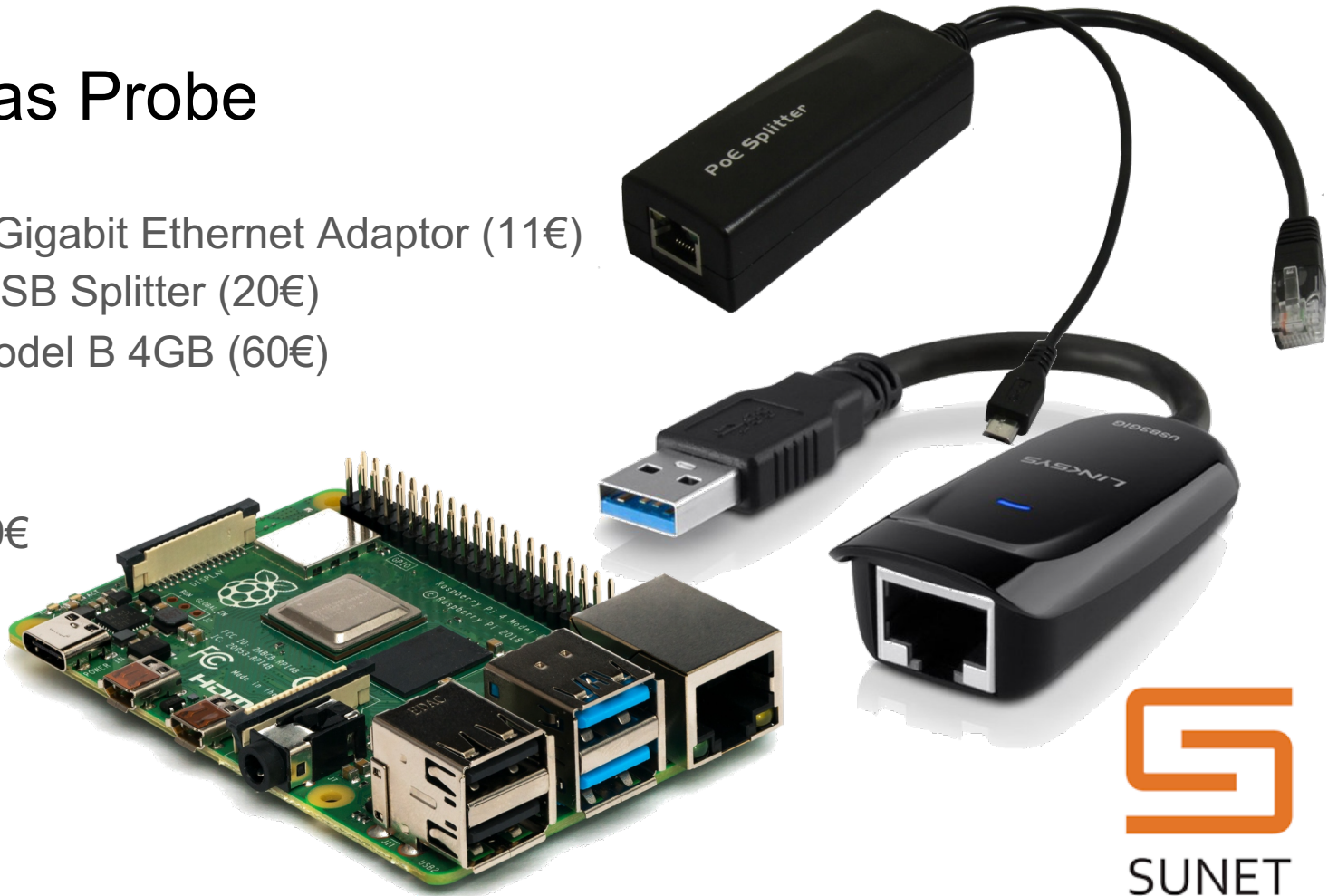
Linksys USB3 Gigabit Ethernet Adaptor (11€)

PoE to micro USB Splitter (20€)

Raspberry 4 Model B 4GB (60€)

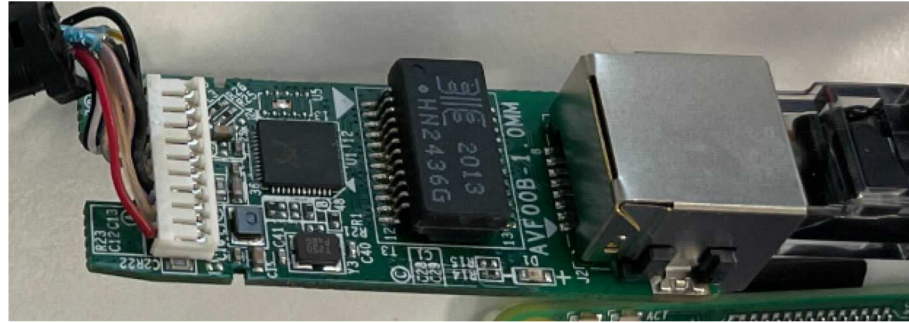
SD-card (8€)

Total cost of 99€

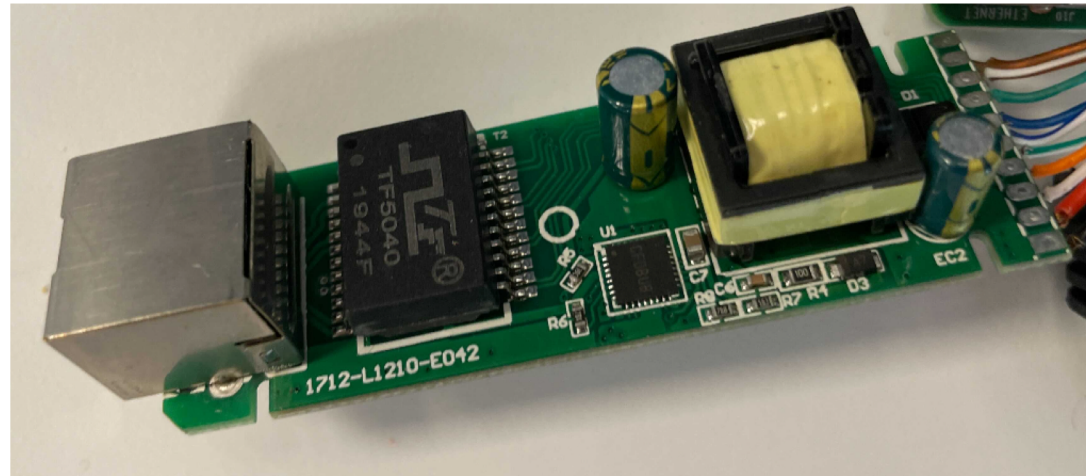


PCB

USB NIC

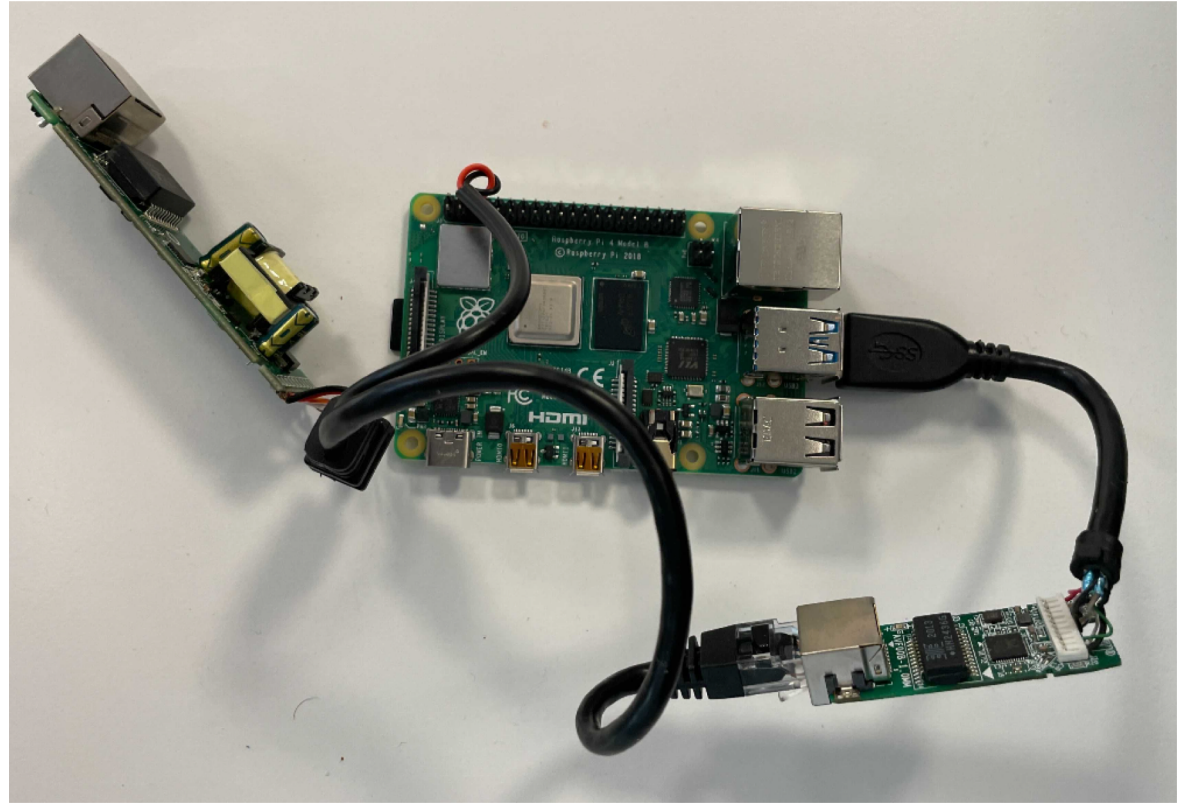


POE Splitter



Complete probe

Solder the POE splitter cables to the raspberry
+5V and GND



Case 3D printed



SUNET

Installed in rack

Rack mounted

Dual Ethernet

POE on MGMT port

reset disable POE

Monitor port build in port.



Iperf3 between two Raspberry

```
pi@raspberrypi:~ $ iperf3 -c 192.168.100.10
Connecting to host 192.168.100.10, port 5201
[ 5] local 192.168.100.11 port 56628 connected to 192.168.100.10 port 5201
[ ID] Interval      Transfer   Bitrate   Retr  Cwnd
[ 5] 0.00-1.00 sec  110 MBytes 919 Mbits/sec  0   368 KBytes
[ 5] 1.00-2.00 sec  112 MBytes 937 Mbits/sec  0   368 KBytes
[ 5] 2.00-3.00 sec  112 MBytes 941 Mbits/sec  0   368 KBytes
[ 5] 3.00-4.00 sec  112 MBytes 937 Mbits/sec  0   368 KBytes
[ 5] 4.00-5.00 sec  112 MBytes 937 Mbits/sec  0   368 KBytes
[ 5] 5.00-6.00 sec  112 MBytes 937 Mbits/sec  0   368 KBytes
[ 5] 6.00-7.00 sec  112 MBytes 941 Mbits/sec  0   513 KBytes
[ 5] 7.00-8.00 sec  111 MBytes 935 Mbits/sec  0   513 KBytes
[ 5] 8.00-9.00 sec  112 MBytes 938 Mbits/sec  0   513 KBytes
[ 5] 9.00-10.00 sec 112 MBytes 939 Mbits/sec  0   513 KBytes
```

```
-----
[ ID] Interval      Transfer   Bitrate   Retr
[ 5] 0.00-10.00 sec 1.09 GBytes 936 Mbits/sec  0      sender
[ 5] 0.00-10.00 sec 1.09 GBytes 935 Mbits/sec                receiver
```

```
Accepted connection from 192.168.100.11, port 56626
[ 8] local 192.168.100.10 port 5201 connected to 192.168.100.11 port 56628
[ ID] Interval      Transfer   Bitrate
[ 8] 0.00-1.00 sec  109 MBytes 911 Mbits/sec
[ 8] 1.00-2.00 sec  112 MBytes 938 Mbits/sec
[ 8] 2.00-3.00 sec  112 MBytes 938 Mbits/sec
[ 8] 3.00-4.00 sec  112 MBytes 938 Mbits/sec
[ 8] 4.00-5.00 sec  112 MBytes 938 Mbits/sec
[ 8] 5.00-6.00 sec  112 MBytes 936 Mbits/sec
[ 8] 6.00-7.00 sec  112 MBytes 937 Mbits/sec
[ 8] 7.00-8.00 sec  112 MBytes 938 Mbits/sec
[ 8] 8.00-9.00 sec  112 MBytes 938 Mbits/sec
[ 8] 9.00-10.00 sec  112 MBytes 938 Mbits/sec
[ 8] 10.00-10.00 sec  160 KBytes 917 Mbits/sec
```

```
-----
[ ID] Interval      Transfer   Bitrate
[ 8] 0.00-10.00 sec 1.09 GBytes 935 Mbits/sec                receiver
```

```
-----
Server listening on 5201
-----
```





SUNET