Network automation with Google Sheets?

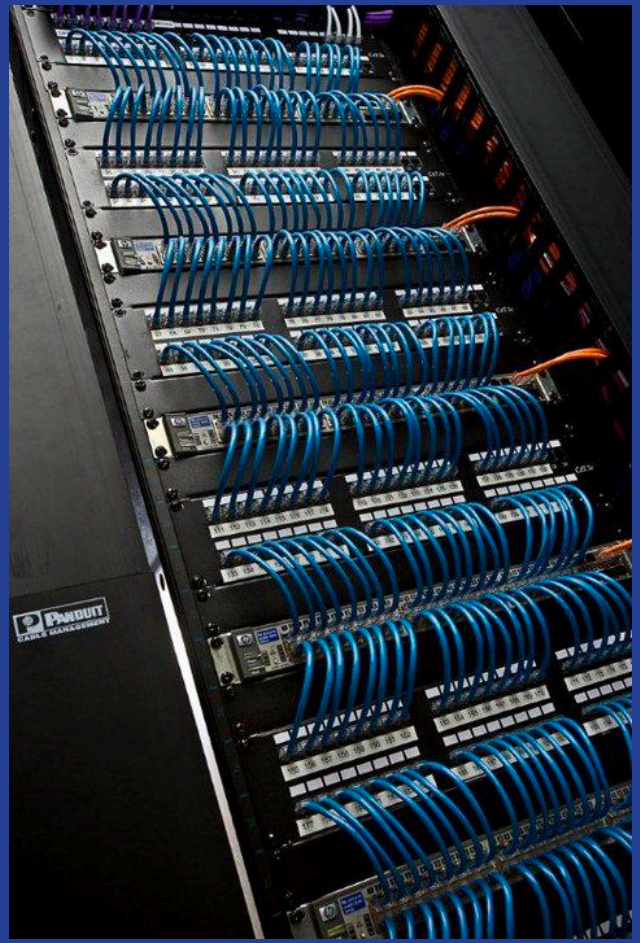
IEVGEN KOSTIUKEVYCH, EBU T&I
OCTOBER 17-18, 2019

HELLO, MY NAME IS...

Ievgen Kostiukevych
Senior IP Media Technology Architect
European Broadcasting Union, Technology & Innovation

JT-NM Tested ST 2110 Test Plan Editor
Joint Taskforce on Networked Media (JT-NM)

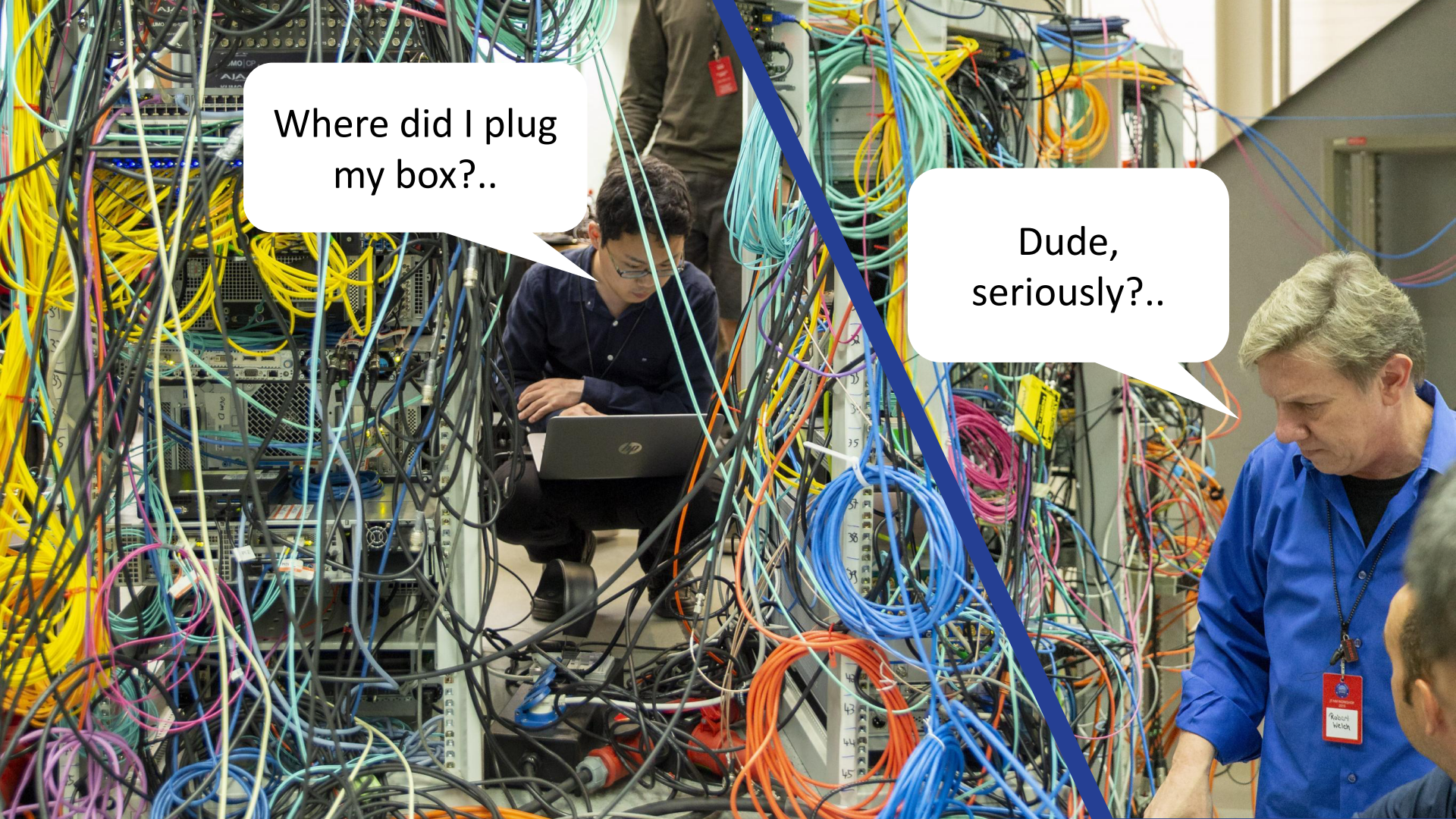
kostiukevych@ebu.ch
https://www.linkedin.com/in/kostiukevych/

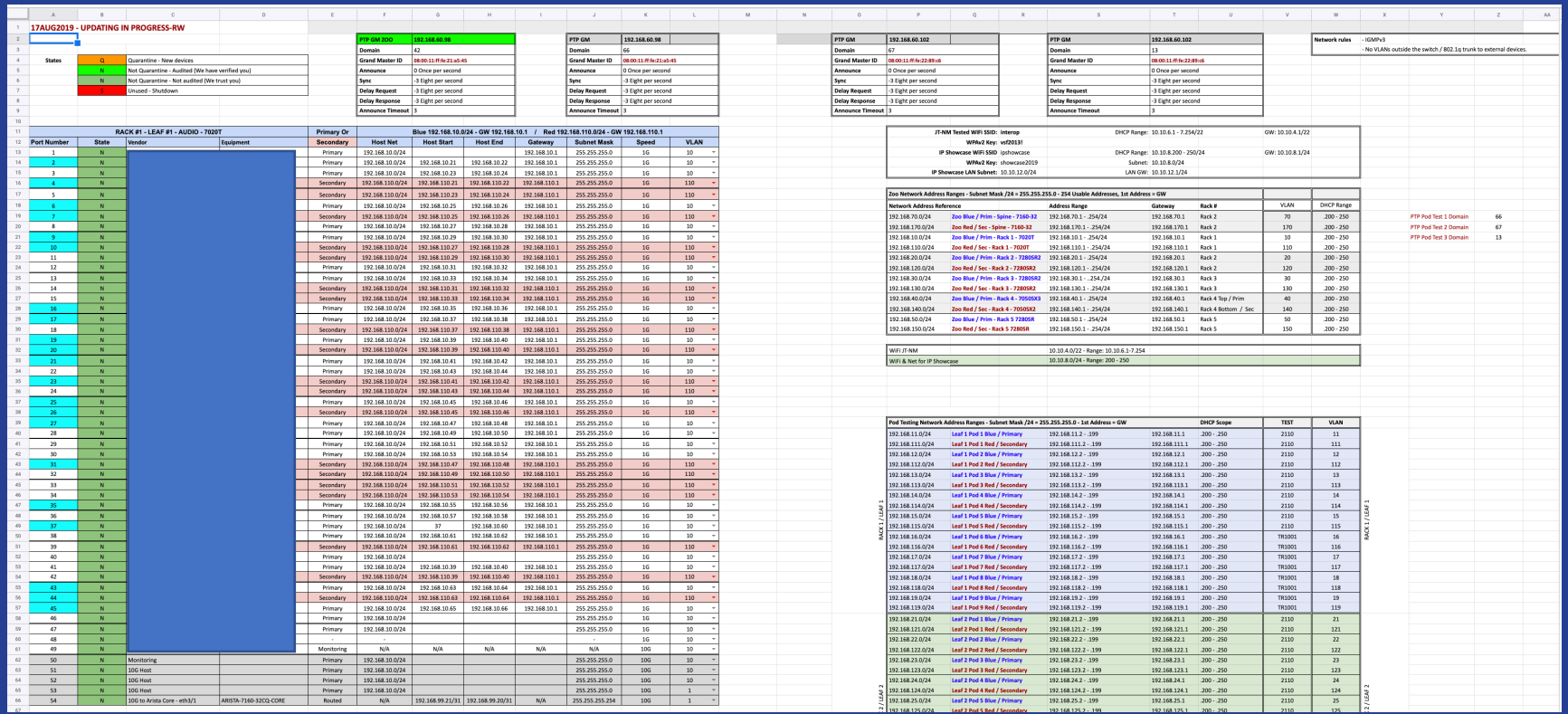


EBU

OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion


Where did I plug my box?..
Dude, seriously?..



17AUG2019 - UPDATING IN PROGRESS-RW

States		
	Q	Quarantine - New devices
	N	Not Quarantine - Audited (We have verified you)
	N	Not Quarantine - Not audited (We trust you)
	S	Unused - Shutdown

PTP GM ZOO	192.168.60.98
Domain	42
Grand Master ID	08:00:11:ff:fe:21:a5:45
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

PTP GM	192.168.60.98
Domain	66
Grand Master ID	08:00:11:ff:fe:21:a5:45
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

PTP GM	192.168.60.102
Domain	67
Grand Master ID	08:00:11:ff:fe:22:89:c6
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

PTP GM	192.168.60.102
Domain	13
Grand Master ID	08:00:11:ff:fe:22:89:c6
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

Network rules
- IGMPv3
- No VLANs outside the switch / 802.1q trunk to external devices.

RACK #1 - LEAF #1 - AUDIO - 7020T

Blue 192.168.10.0/24 - GW 192.168.10.1 / Red 192.168.110.0/24 - GW 192.168.110.1

Port Number	State	Vendor	Equipment	Primary Or Secondary	Host Net	Host Start	Host End	Gateway	Subnet Mask	Speed	VLAN
1	N			Primary	192.168.10.0/24			192.168.10.1	255.255.255.0	1G	10
2	N			Primary	192.168.10.0/24	192.168.10.21	192.168.10.22	192.168.10.1	255.255.255.0	1G	10
3	N			Primary	192.168.10.0/24	192.168.10.23	192.168.10.24	192.168.10.1	255.255.255.0	1G	10
4	N			Secondary	192.168.110.0/24	192.168.110.21	192.168.110.22	192.168.110.1	255.255.255.0	1G	110
5	N			Secondary	192.168.110.0/24	192.168.110.23	192.168.110.24	192.168.110.1	255.255.255.0	1G	110
6	N			Primary	192.168.10.0/24	192.168.10.25	192.168.10.26	192.168.10.1	255.255.255.0	1G	10
7	N			Secondary	192.168.110.0/24	192.168.110.25	192.168.110.26	192.168.110.1	255.255.255.0	1G	110
8	N			Primary	192.168.10.0/24	192.168.10.27	192.168.10.28	192.168.10.1	255.255.255.0	1G	10
9	N			Primary	192.168.10.0/24	192.168.10.29	192.168.10.30	192.168.10.1	255.255.255.0	1G	10
10	N			Secondary	192.168.110.0/24	192.168.110.27	192.168.110.28	192.168.110.1	255.255.255.0	1G	110
11	N			Secondary	192.168.110.0/24	192.168.110.29	192.168.110.30	192.168.110.1	255.255.255.0	1G	110
12	N			Primary	192.168.10.0/24	192.168.10.31	192.168.10.32	192.168.10.1	255.255.255.0	1G	10
13	N			Primary	192.168.10.0/24	192.168.10.33	192.168.10.34	192.168.10.1	255.255.255.0	1G	10
14	N			Secondary	192.168.110.0/24	192.168.110.31	192.168.110.32	192.168.110.1	255.255.255.0	1G	110
15	N			Secondary	192.168.110.0/24	192.168.110.33	192.168.110.34	192.168.110.1	255.255.255.0	1G	110
16	N			Primary	192.168.10.0/24	192.168.10.35	192.168.10.36	192.168.10.1	255.255.255.0	1G	10
17	N			Primary	192.168.10.0/24	192.168.10.37	192.168.10.38	192.168.10.1	255.255.255.0	1G	10
18	N			Secondary	192.168.110.0/24	192.168.110.37	192.168.110.38	192.168.110.1	255.255.255.0	1G	110
19	N			Primary	192.168.10.0/24	192.168.10.39	192.168.10.40	192.168.10.1	255.255.255.0	1G	10
20	N			Secondary	192.168.110.0/24	192.168.110.39	192.168.110.40	192.168.110.1	255.255.255.0	1G	110
21	N			Primary	192.168.10.0/24	192.168.10.41	192.168.10.42	192.168.10.1	255.255.255.0	1G	10
22	N			Primary	192.168.10.0/24	192.168.10.43	192.168.10.44	192.168.10.1	255.255.255.0	1G	10
23	N			Secondary	192.168.110.0/24	192.168.110.41	192.168.110.42	192.168.110.1	255.255.255.0	1G	110
24	N			Secondary	192.168.110.0/24	192.168.110.43	192.168.110.44	192.168.110.1	255.255.255.0	1G	110
25	N			Primary	192.168.10.0/24	192.168.10.45	192.168.10.46	192.168.10.1	255.255.255.0	1G	10
26	N			Secondary	192.168.110.0/24	192.168.110.45	192.168.110.46	192.168.110.1	255.255.255.0	1G	110
27	N			Primary	192.168.10.0/24	192.168.10.47	192.168.10.48	192.168.10.1	255.255.255.0	1G	10
28	N			Primary	192.168.10.0/24	192.168.10.49	192.168.10.50	192.168.10.1	255.255.255.0	1G	10
29	N			Primary	192.168.10.0/24	192.168.10.51	192.168.10.52	192.168.10.1	255.255.255.0	1G	10
30	N			Primary	192.168.10.0/24	192.168.10.53	192.168.10.54	192.168.10.1	255.255.255.0	1G	10
31	N			Secondary	192.168.110.0/24	192.168.110.47	192.168.110.48	192.168.110.1	255.255.255.0	1G	110
32	N			Secondary	192.168.110.0/24	192.168.110.49	192.168.110.50	192.168.110.1	255.255.255.0	1G	110
33	N			Secondary	192.168.110.0/24	192.168.110.51	192.168.110.52	192.168.110.1	255.255.255.0	1G	110
34	N			Secondary	192.168.110.0/24	192.168.110.53	192.168.110.54	192.168.110.1	255.255.255.0	1G	110
35	N			Primary	192.168.10.0/24	192.168.10.55	192.168.10.56	192.168.10.1	255.255.255.0	1G	10
36	N			Primary	192.168.10.0/24	192.168.10.57	192.168.10.58	192.168.10.1	255.255.255.0	1G	10
37	N			Primary	192.168.10.0/24	37	192.168.10.60	192.168.10.1	255.255.255.0	1G	10
38	N			Primary	192.168.10.0/24	192.168.10.61	192.168.10.62	192.168.10.1	255.255.255.0	1G	10
39	N			Secondary	192.168.110.0/24	192.168.110.61	192.168.110.62	192.168.110.1	255.255.255.0	1G	110
40	N			Primary	192.168.10.0/24				255.255.255.0	1G	10
41	N			Primary	192.168.10.0/24	192.168.10.39	192.168.10.40	192.168.10.1	255.255.255.0	1G	10
42	N			Secondary	192.168.110.0/24	192.168.110.39	192.168.110.40	192.168.110.1	255.255.255.0	1G	110
43	N			Primary	192.168.10.0/24	192.168.10.63	192.168.10.64	192.168.10.1	255.255.255.0	1G	10
44	N			Secondary	192.168.110.0/24	192.168.110.63	192.168.110.64	192.168.110.1	255.255.255.0	1G	110
45	N			Primary	192.168.10.0/24	192.168.10.65	192.168.10.66	192.168.10.1	255.255.255.0	1G	10
46	N			Primary	192.168.10.0/24				255.255.255.0	1G	10
47	N			Primary	192.168.10.0/24				255.255.255.0	1G	10
48	N										10
49	N			Monitoring	N/A	N/A	N/A	N/A	N/A	10G	10
50	N	Monitoring		Primary	192.168.10.0/24				255.255.255.0	10G	10
51	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	10
52	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	10
53	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	1
54	N	10G to Arista Core - eth1/1	ARISTA-7160-32CQ-CORE	Routed	N/A	192.168.99.21/31	192.168.99.20/31	N/A	255.255.255.254	10G	1

JT-NM Tested WiFi SSID:	interop	DHCP Range:	10.10.6.1 - 7.254/22	GW:	10.10.4.1/22
WPAv2 Key:	vsf2013!				
IP Showcase WiFi SSID	ipshowcase	DHCP Range:	10.10.8.200 - 250/24	GW:	10.10.8.1/24
WPAv2 Key:	showcase2019	Subnet:	10.10.8.0/24		
IP Showcase LAN Subnet:	10.10.12.0/24	LAN GW:	10.10.12.1/24		

Zoo Network Address Ranges - Subnet Mask /24 = 255.255.255.0 - 254 Usable Addresses, 1st Address = GW

Network Address Reference	Address Range	Gateway	Rack #	VLAN	DHCP Range
192.168.70.0/24	Zoo Blue / Prim - Spine - 7160-32	192.168.70.1 - 254/24	192.168.70.1	Rack 2	70	.200 - 250
192.168.170.0/24	Zoo Red / Sec - Spine - 7160-32	192.168.170.1 - 254/24	192.168.170.1	Rack 2	170	.200 - 250
192.168.10.0/24	Zoo Blue / Prim - Rack 1 - 7020T	192.168.10.1 - 254/24	192.168.10.1	Rack 1	10	.200 - 250
192.168.110.0/24	Zoo Red / Sec - Rack 1 - 7020T	192.168.110.1 - 254/24	192.168.110.1	Rack 1	110	.200 - 250
192.168.20.0/24	Zoo Blue / Prim - Rack 2 - 7280SR2	192.168.20.1 - 254/24	192.168.20.1	Rack 2	20	.200 - 250
192.168.120.0/24	Zoo Red / Sec - Rack 2 - 7280SR2	192.168.120.1 - 254/24	192.168.120.1	Rack 2	120	.200 - 250
192.168.30.0/24	Zoo Blue / Prim - Rack 3 - 7280SR2	192.168.30.1 - 254/24	192.168.30.1	Rack 3	30	.200 - 250
192.168.130.0/24	Zoo Red / Sec - Rack 3 - 7280SR2	192.168.130.1 - 254/24	192.168.130.1	Rack 3	130	.200 - 250
192.168.40.0/24	Zoo Blue / Prim - Rack 4 - 7050SX3	192.168.40.1 - 254/24	192.168.40.1	Rack 4 Top / Prim	40	.200 - 250
192.168.140.0/24	Zoo Red / Sec - Rack 4 - 7050SX2	192.168.140.1 - 254/24	192.168.140.1	Rack 4 Bottom / Sec	140	.200 - 250
192.168.50.0/24	Zoo Blue / Prim - Rack 5 7280SR	192.168.50.1 - 254/24	192.168.50.1	Rack 5	50	.200 - 250
192.168.150.0/24	Zoo Red / Sec - Rack 5 7280SR	192.168.150.1 - 254/24	192.168.150.1	Rack 5	150	.200 - 250

WiFi JT-NM	10.10.4.0/22 - Range: 10.10.6.1-7.254
WiFi & Net for IP Showcase	10.10.8.0/24 - Range: 200 - 250

Pod Testing Network Address Ranges - Subnet Mask /24 = 255.255.255.0 - 1st Address = GW

				DHCP Scope	TEST	VLAN
192.168.11.0/24	Leaf 1 Pod 1 Blue / Primary	192.168.11.2 - 199	192.168.11.1	.200 - 250	2110	11
192.168.111.0/24	Leaf 1 Pod 1 Red / Secondary	192.168.111.2 - 199	192.168.111.1	.200 - 250	2110	111
192.168.12.0/24	Leaf 1 Pod 2 Blue / Primary	192.168.12.2 - 199	192.168.12.1	.200 - 250	2110	12
192.168.112.0/24	Leaf 1 Pod 2 Red / Secondary	192.168.112.2 - 199	192.168.112.1	.200 - 250	2110	112
192.168.13.0/24	Leaf 1 Pod 3 Blue / Primary	192.168.13.2 - 199	192.168.13.1	.200 - 250	2110	13
192.168.113.0/24	Leaf 1 Pod 3 Red / Secondary	192.168.113.2 - 199	192.168.113.1	.200 - 250	2110	113
192.168.14.0/24	Leaf 1 Pod 4 Blue / Primary	192.168.14.2 - 199	192.168.14.1	.200 - 250	2110	14
192.168.114.0/24	Leaf 1 Pod 4 Red / Secondary	192.168.114.2 - 199	192.168.114.1	.200 - 250	2110	114
192.168.15.0/24	Leaf 1 Pod 5 Blue / Primary	192.168.15.2 - 199	192.168.15.1	.200 - 250	2110	15
192.168.115.0/24	Leaf 1 Pod 5 Red / Secondary	192.168.115.2 - 199	192.168.115.1	.200 - 250	2110	115
192.168.16.0/24	Leaf 1 Pod 6 Blue / Primary	192.168.16.2 - 199	192.168.16.1	.200 - 250	TR1001	16
192.168.116.0/24	Leaf 1 Pod 6 Red / Secondary	192.168.116.2 - 199	192.168.116.1	.200 - 250	TR1001	116
192.168.17.0/24	Leaf 1 Pod 7 Blue / Primary	192.168.17.2 - 199	192.168.17.1	.200 - 250	TR1001	17
192.168.117.0/24	Leaf 1 Pod 7 Red / Secondary	192.168.117.2 - 199	192.168.117.1	.200 - 250	TR1001	117
192.168.18.0/24	Leaf 1 Pod 8 Blue / Primary	192.168.18.2 - 199	192.168.18.1	.200 - 250	TR1001	18
192.168.118.0/24	Leaf 1 Pod 8 Red / Secondary	192.168.118.2 - 199	192.168.118.1	.200 - 250	TR1001	118
192.168.19.0/24	Leaf 1 Pod 9 Blue / Primary	192.168.19.2 - 199	192.168.19.1	.200 - 250	TR1001	19
192.168.119.0/24	Leaf 1 Pod 9 Red / Secondary	192.168.119.2 - 199	192.168.119.1	.200 - 250	TR1001	119
192.168.21.0/24	Leaf 2 Pod 1 Blue / Primary	192.168.21.2 - 199	192.168.21.1	.200 - 250	2110	21
192.168.121.0/24	Leaf 2 Pod 1 Red / Secondary	192.168.121.2 - 199	192.168.121.1	.200 - 250	2110	121
192.168.22.0/24	Leaf 2 Pod 2 Blue / Primary	192.168.22.2 - 199	192.168.22.1	.200 - 250	2110	22
192.168.122.0/24	Leaf 2 Pod 2 Red / Secondary	192.168.122.2 - 199	192.168.122.1	.200 - 250	2110	122
192.168.23.0/24	Leaf 2 Pod 3 Blue / Primary	192.168.23.2 - 199	192.168.23.1	.200 - 250	2110	23
192.168.123.0/24	Leaf 2 Pod 3 Red / Secondary	192.168.123.2 - 199	192.168.123.1	.200 - 250	2110	123
192.168.24.0/24	Leaf 2 Pod 4 Blue / Primary	192.168.24.2 - 199	192.168.24.1	.200 - 250	2110	24
192.168.124.0/24	Leaf 2 Pod 4 Red / Secondary	192.168.124.2 - 199	192.168.124.1	.200 - 250	2110	124
192.168.25.0/24	Leaf 2 Pod 5 Blue / Primary	192.168.25.2 - 199	192.168.25.1	.200 - 250	2110	25
192.168.125.0/24	Leaf 2 Pod 5 Red / Secondary	192.168.125.2 - 199	192.168.125.1	.200 - 250	2110	125

RACK 1 / LEAF 1

RACK 2 / LEAF 2

EBU

OPERATING EUROVISION AND EURORADIO

CLI?

```
ARISTA-7020TR-LEAF-2.Wed-13:52:01#conf
ARISTA-7020TR-LEAF-2.Wed-13:52:12(config)#int eth 31
ARISTA-7020TR-LEAF-2.Wed-13:52:17(config-if-Et31)#desc Ward Beck preMO-221 + GPIO
ARISTA-7020TR-LEAF-2.Wed-13:52:38(config-if-Et31)#int eth 32
ARISTA-7020TR-LEAF-2.Wed-13:52:43(config-if-Et32)#desc Ward Beck 32ME-N
ARISTA-7020TR-LEAF-2.Wed-13:52:48(config-if-Et32)#int eth 33
ARISTA-7020TR-LEAF-2.Wed-13:52:58(config-if-Et33)#Ward Beck AMS8-1AN
% Invalid input
ARISTA-7020TR-LEAF-2.Wed-13:53:00(config-if-Et33)#desc Ward Beck AMS8-1AN
ARISTA-7020TR-LEAF-2.Wed-13:53:06(config-if-Et33)#
```

*Taken from the JT-NM Tested March 2019

EBU
OPERATING EUROVISION AND EURORADIO

CLI?

ARISTA-7020TR-LEAF-2.Wed-13:53:06(co

Et7	DirectOut GmbH MONTONE.42	connected	routed	a-full	a-1G	1000BASE-T
Et8	Riedel RSP-2318	connected	10	a-full	a-1G	1000BASE-T
Et9	Riedel AES67-108 G2	connected	10	a-full	a-1G	1000BASE-T
Et10	Stage Tec NEXUS XFIP/RIF67	connected	311	a-full	a-1G	1000BASE-T
Et11	Stage Tec NEXUS XFIP/RIF67	connected	312	a-full	a-1G	1000BASE-T
Et12	Studio Technologies Model 5512-02	connected	10	a-full	a-1G	1000BASE-T
Et13	Studio Technologies Model 5512-02	connected	110	a-full	a-1G	1000BASE-T
Et14	Studio Technologies Model 5518	notconnect	10	auto	auto	1000BASE-T
Et15	Studio Technologies Model 5518	notconnect	110	auto	auto	1000BASE-T
Et16	Ross Video IGGY-MADI	connected	10	a-full	a-1G	1000BASE-T
Et17	Ross Video IGGY-MADI	connected	110	a-full	a-1G	1000BASE-T
Et18	Ross Video Minuet	connected	10	a-full	a-1G	1000BASE-T
Et19	Ross Video Minuet	connected	110	a-full	a-1G	1000BASE-T
Et20	Ross Video Bach openModule	connected	10	a-full	a-1G	1000BASE-T
Et21	Ross Video Bach openModule	notconnect	110	auto	auto	1000BASE-T
Et22	Ross Video Bach Liberty	connected	10	a-full	a-1G	1000BASE-T
Et23	Ross Video Bach Liberty	connected	110	a-full	a-1G	1000BASE-T
Et24	The Telos Alliance Axia iQx AoIP console	connected	10	a-full	a-1G	1000BASE-T
Et25	The Telos Alliance Telos Infinity IP Intercom Master Panel	connected	10	a-full	a-1G	1000BASE-T
Et26	The Telos Alliance Telos Infinity IP Intercom Beltpack	connected	10	a-full	a-100M	1000BASE-T
Et27	The Telos Alliance Mixed Signal xNode	connected	10	a-full	a-100M	1000BASE-T
Et28	The Telos Alliance AES/EBU xNode	connected	10	a-full	a-100M	1000BASE-T
Et29	The Telos Alliance Mic xNode	connected	10	a-full	a-100M	1000BASE-T
Et30	The Telos Alliance Analog xNode	connected	10	a-full	a-100M	1000BASE-T
Et31	Ward Beck	notconnect	10	auto	auto	1000BASE-T
Et32	Ward Beck	notconnect	10	auto	auto	1000BASE-T
Et33	Ward Beck	notconnect	10	auto	auto	1000BASE-T
Et34	1G Host	connected	10	a-full	a-100M	1000BASE-T
Et35	1G Host	connected	10	a-full	a-100M	1000BASE-T
Et36	1G Host	notconnect	10	auto	auto	1000BASE-T
Et37	1G Host	notconnect	10	auto	auto	1000BASE-T
Et38	1G Host	notconnect	10	auto	auto	1000BASE-T
Et39	1G Host	notconnect	10	auto	auto	1000BASE-T
Et40	1G Host	notconnect	10	auto	auto	1000BASE-T
Et41	1G Host	notconnect	10	auto	auto	1000BASE-T
Et42	1G Host	notconnect	10	auto	auto	1000BASE-T
Et43	1G Host	notconnect	10	auto	auto	1000BASE-T
Et44	1G Host	notconnect	10	auto	auto	1000BASE-T
Et45	1G Host	notconnect	10	auto	auto	1000BASE-T
Et46	1G Host	notconnect	10	auto	auto	1000BASE-T
Et47	1G Host	notconnect	10	auto	auto	1000BASE-T
Et48	ARISTA DEBUG - RWelch	notconnect	10	auto	auto	1000BASE-T
Et49	PacketStorm CRS (3)10GE Port B	connected	routed	full	10G	10GBASE-SRL
Et50	100G Host	notconnect	10	full	10G	Not Present
Et51	100G Host	notconnect	10	full	10G	Not Present
Et52	100G Host	notconnect	10	full	10G	Not Present
Et53	100G to Arista Spine - eth2/2 ARISTA-SPINE-1	connected	routed	full	10G	10GBASE-SRL
Et54	100G to Arista Spine - eth2/3 ARISTA-SPINE-1	connected	routed	full	10G	10GBASE-SRL
Ma1		connected	routed	a-full	a-1G	10/100/1000

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

*Taken from the JT-NM Tested March 2019

CLI?

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

IDEA

- A centralized, but dynamic and collaborative repository for often changed switch parameters.
- A centralized, but dynamic and collaborative view of troubleshooting data from the switch, the “show” commands.
- A real time update or periodic polling of both.
- Had to support Arista switches

EBU
OPERATING EUROVISION AND EURORADIO

POSSIBLE SOLUTIONS

- **CLI?**
- **Ansible?**
- **Python?**
  - CLI (Telnet/SSH)?
  - JSON?
  - Native API?

EBU
OPERATING EUROVISION AND EURORADIO

PYTHON

- telnetlib
- paramiko
- netmiko
- native programmatic API library (pyeapi)

TELNETLIB

```python
import getpass
import sys
import telnetlib

ip_address = "192.168.122.71"
user = 'admin'
password = 'admin'

tn = telnetlib.Telnet(ip_address)

tn.read_until("Username: ")
tn.write(user + "\n")
tn.read_until("Password: ")
tn.write(password + "\n")

tn.write("enable\n")
tn.write("super_secure_enable_pass\n")
tn.write("show version\n")

print(tn.read_all())
```

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

PARAMIKO

```python
import paramiko
import time

ip_address = "192.168.122.72"
username = "admin"
password = "admin"

ssh_client = paramiko.SSHClient()
ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
ssh_client.connect(hostname=ip_address, username=username, password=password)

remote_connection = ssh_client.invoke_shell()

remote_connection.send("enable\n")
remote_connection.send("super_secure_enable_pass\n")
remote_connection.send("show version\n")

time.sleep(1)
output = remote_connection.recv(65535)
print(output)

ssh_client.close
```

EBU
OPERATING EUROVISION AND EURORADIO

NETMIKO

```python
from netmiko import ConnectHandler

S1 = {
    'device_type': 'arista_eos',
    'ip': '192.168.122.72',
    'username': 'admin',
    'password': 'admin',
}


net_connect = ConnectHandler(**S1)
output = net_connect.send_command('show version')
print(output)

```

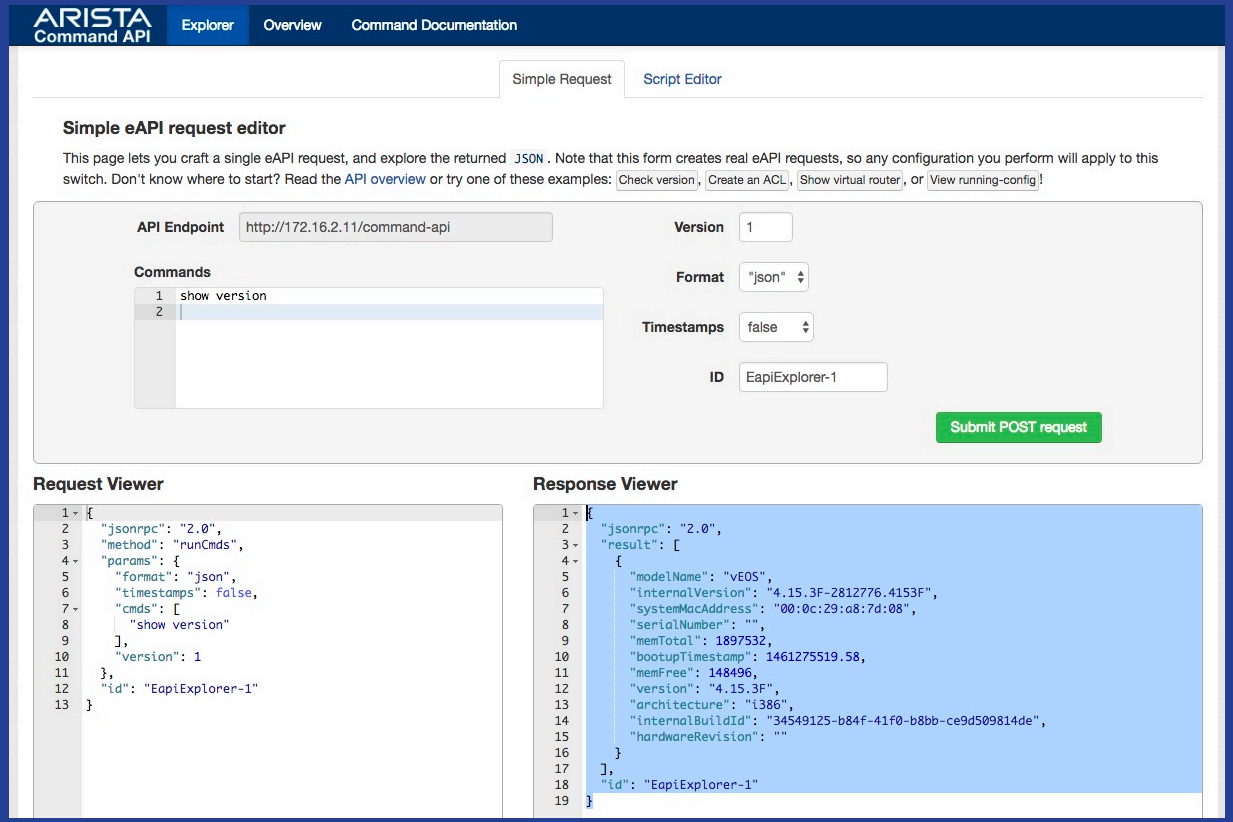
EBU
OPERATING EUROVISION AND EURORADIO

NATIVE PROGRAMMATIC API LIBRARY (PYEAPI)

```
1   import pyeapi
2   connect = pyeapi.client.connect(
3       transport='https',
4       host='192.168.22.72',
5       username='admin',
6       password='admin')
7   connectedSwitch = pyeapi.client.Node(connect)
8
9   connectedSwitch.enable('show version')
10
```

EBU
OPERATING EUROVISION AND EURORADIO

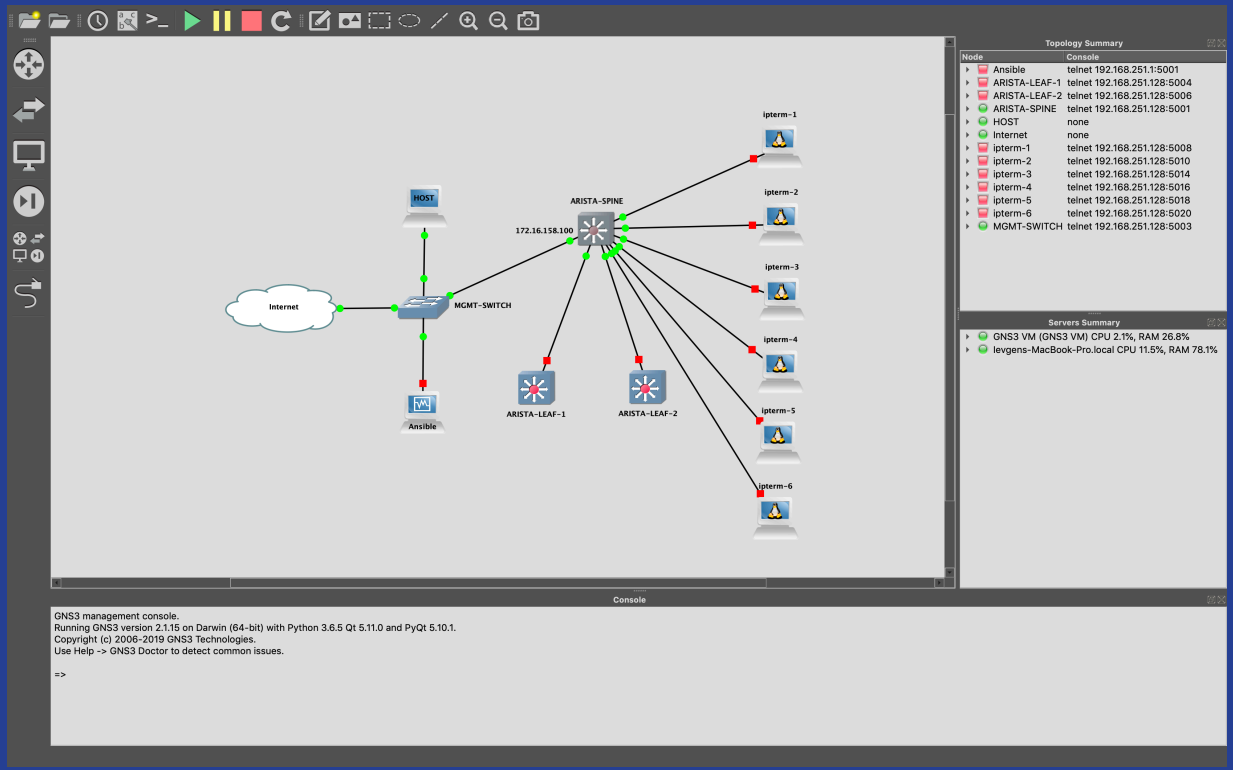
ARISTA API EXPLORER



EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

TESTING ENVIRONMENT



EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

GOOGLE API AUTHENTICATION

```python
# ================== Google sheet access ====================


def googleAuthorize():
    # Sets access to Google spreadsheets and Google drive
    scope = ['https://spreadsheets.google.com/feeds',
             'https://www.googleapis.com/auth/drive']
    # Imports the JSON access token and extracts credential
    credentials = ServiceAccountCredentials.from_json_keyfile_name(
        args.api, scope)
    # Athorization with extracted credentials
    gc = gspread.authorize(credentials)
    logger.info(' - Google Cloud API authorization successful')
    return(gc)


# Initial authorization at program start
gc = googleAuthorize()

# Extracts data from the spreadsheet
switchConfigs = gc.open(config['spreadsheet'])
vlanList = switchConfigs.worksheet('Vlan List')
vlanPorts = switchConfigs.worksheet(
    'Interfaces VLAN Allocation and Descriptions')
interfacesMacTable = switchConfigs.worksheet('MAC addresses table')
interfacesStatusTable = switchConfigs.worksheet('Interfaces status table')
```

EBU
OPERATING EUROVISION AND EURORADIO

SWITCH CONNECTION SESSION

```
# First - connect to the switch and check connection and eAPI
connect = pyeapi.client.connect(
    transport='https', host=config['ip'], username=config['username'], password=config['password'])
logger.info(
    ' - Trying to connect to {} using provided username and password'.format(config['ip']))
connectedSwitch = pyeapi.client.Node(connect)
hostname = connectedSwitch.enable('show hostname')
logger.info(' - Succsessfully connected to ' +
            hostname[0]['result']['hostname'])
version = connectedSwitch.enable('show version')
logger.info(pp.pprint(version[0]['result']))
# Second - read initial data from the switch
vlans = connectedSwitch.api('vlans')
portsVlans = connectedSwitch.api('switchports')
portsDescription = connectedSwitch.api('interfaces')

# Start executing selected tasks when continious flag is not set

if args.vlans_list and not args.continious:
    createVlans()

if args.interfaces_description and not args.continious:
    setInterfaceDescriptions()

if args.interfaces_vlans and not args.continious:
    setInterfaceVlans()

if args.interfaces_status and not args.continious:
    getInterfacesState()

if not args.continious:
    logger.info('============================================')
    logger.info(' - All tasks finished. Exiting...')
```

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

WRITING TO A SWITCH

```
def setInterfaceDescriptions():
    # Reads interfaces description from the spreadsheet and writes to the switch
    logger.info('============================================')
    logger.info(' - Updating interfaces descriprions...')
    for port in vlanPorts.get_all_records():
        portsDescription.set_description('Ethernet {}'.format(
            port['Port']), value=port['Description'])
    logger.info(' - Interfaces descriptions updated')


def createVlans():
    # Reads vlan list and names from the spreadsheet and writes to the switch
    logger.info('============================================')
    logger.info(' - Updating vlan table...')
    vlans = connectedSwitch.api('vlans')
    vlans.autorefresh = True
    for vlan in vlanList.get_all_records():
        vlans.create(vlan['Vlan ID'])
        vlans.set_name(vlan['Vlan ID'], name=vlan['Vlan Name'])
    logger.info(' - Vlan table updated')


def setInterfaceVlans():
    # Reads interfaces vlans list from the spreadsheet and writes to the switch
    logger.info('============================================')
    logger.info(' - Updating interface vlans...')
    currentVlans = portsVlans.getall()
    for port in vlanPorts.get_all_records():
        try:
            if str(port['Vlan']) != str(currentVlans['Ethernet{}'.format(port['Port'])]['access_vlan']):
                logger.warning(
                    '   - Modifying port {} - new access Vlan {}'.format(port['Port'], port['Vlan']))
                portsVlans.set_access_vlan('Ethernet {}'.format(
                    port['Port']), value=port['Vlan'])
            else:
                pass
        except KeyError:
            pass
    logger.info(' - Interface vlans updated')
```

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

PROTOTYPE CONTROL FRONT-END

Switches

Файл Изменить Вид Вставка Формат Данные Инструменты Доп

Vlan ID

	A	B	C	D	E
1	Vlan ID	Vlan Name			
2	1	default			
3	10	TEN			
4	20	TWENTY			
5	30	THIRTY			
6	40	FOURTY			
7	50	FIFTY			
8	60	SIXTY			
9	70	SEVENTY			
10	100	Management			
11	200	Zoo			
12	210	POD_1			
13	220	POD_2			
14	230	POD_3			
15	240	POD_4			
16	250	POD_5			

Vlan List | Interfaces VLAN Allocation and Descriptions

Switches

Файл Изменить Вид Вставка Формат Данные Инструменты Доп

Port

	A	B	C	D	E
1	Port	Vlan	Description		
2	1	1	Device 1		
3	2	20	Device 2		
4	3	200	Device 3		
5	4	20	Device 4		
6	5	100	Device 5		
7	6	20	Device 6		
8	7	10	Device 7		
9	8	220	Device 8		
10	9	20	Device 9		
11	10	30	Device 10		
12	11	20	Device 11		
13	12	240	Device 12		

Vlan List | Interfaces VLAN Allocation and Descriptions

EBU

OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

READING FROM A SWITCH

```
def getInterfacesState():
    # Reads interfaces statuses and mac address table from the switch and writes to the spreadsheet
    logger.info('==========================================')
    logger.info(' - Updating interfaces status table..')
    getIterfaces = connectedSwitch.enable('show interfaces status')
    interfacesStatus = getIterfaces[0]['result']['interfaceStatuses']
    macAdrTable = connectedSwitch.enable('show mac address-table')
    unicastMacAdrTable = macAdrTable[0]['result']['unicastTable']['tableEntries']

    header = ['autoNegotiateActive', 'autoNegotigateActive', 'bandwidth',
              'description', 'duplex', 'interfaceType', 'lineProtocolStatus', 'linkStatus', 'macAddress', 'entryType']
    # Define the cell range
    cellRange = interfacesStatusTable.range('A2:L256')

    # Sort by interface name
    interfaceList = []
    for interface in interfacesStatus.keys():
        interfaceList.append(interface)
    interfaceList.sort()

    # Flatten the list of dicts into a list of values in order
    flattened_test_data = []
    # ==================
    for i in interfaceList:
        for entry in unicastMacAdrTable:
            for value in entry.values():
                if value == i:
                    interfacesStatus[i].update(
                        {'macAddress': entry['macAddress']})
                    interfacesStatus[i].update(
                        {'entryType': entry['entryType']})
    # ==================

    for i in interfaceList:
        flattened_test_data.append(i)
        try:
            flattened_test_data.append(
                interfacesStatus[i]['vlanInformation']['vlanId'])
        except KeyError:
            flattened_test_data.append('N/A')
        for j in header:
            try:
                flattened_test_data.append(interfacesStatus[i][j])
            except KeyError:
                flattened_test_data.append('N/A')
    # Send flattened list to the cell range to be re-rendered as a table
    for i, cell in enumerate(cellRange):
        try:
            cell.value = flattened_test_data[i]
        except IndexError:
            pass

    interfacesStatusTable.update_cells(cellRange)
    logger.info(' - Interfaces status table updated')
```

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

PROTOTYPE “TELEMETRY” FRONT-END

Switches

Файл Изменить Вид Вставка Формат Данные Инструменты Дополнения Справка Последнее изменение: python-api 9 дней назад

Настройки Дост

Interface

Interface	Vlan ID	autoNegotiateActive	autoNegotigateActive	bandwidth	description	duplex	interfaceType	lineProtocolStatus	linkStatus	MAC Address	Entry Type
Ethernet1	1	FALSE	FALSE	0	Device 1	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet10	30	FALSE	FALSE	0	Device 10	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet11	20	FALSE	FALSE	0	Device 11	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet12	240	FALSE	FALSE	0	Device 12	duplexFull	EbraTestPhyPort	up	connected	76:d0:3c:48:d8:31	dynamic
Ethernet2	20	FALSE	FALSE	0	Device 2	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet3	200	FALSE	FALSE	0	Device 3	duplexFull	EbraTestPhyPort	up	connected	96:b6:4c:f8:8c:1a	dynamic
Ethernet4	20	FALSE	FALSE	0	Device 4	duplexFull	EbraTestPhyPort	up	connected	1a:b2:af:cc:7a:c6	dynamic
Ethernet5	100	FALSE	FALSE	0	Device 5	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet6	20	FALSE	FALSE	0	Device 6	duplexFull	EbraTestPhyPort	up	connected	9e:94:82:cc:ed:ac	dynamic
Ethernet7	10	FALSE	FALSE	0	Device 7	duplexFull	EbraTestPhyPort	up	connected	ba:ac:81:78:3e:1a	dynamic
Ethernet8	220	FALSE	FALSE	0	Device 8	duplexFull	EbraTestPhyPort	up	connected	N/A	N/A
Ethernet9	20	FALSE	FALSE	0	Device 9	duplexFull	EbraTestPhyPort	up	connected	7e:05:67:4c:9f:5b	dynamic
Management1	N/A	TRUE	TRUE	1000000000	MGMT	duplexFull	10/100/1000	up	connected	N/A	N/A

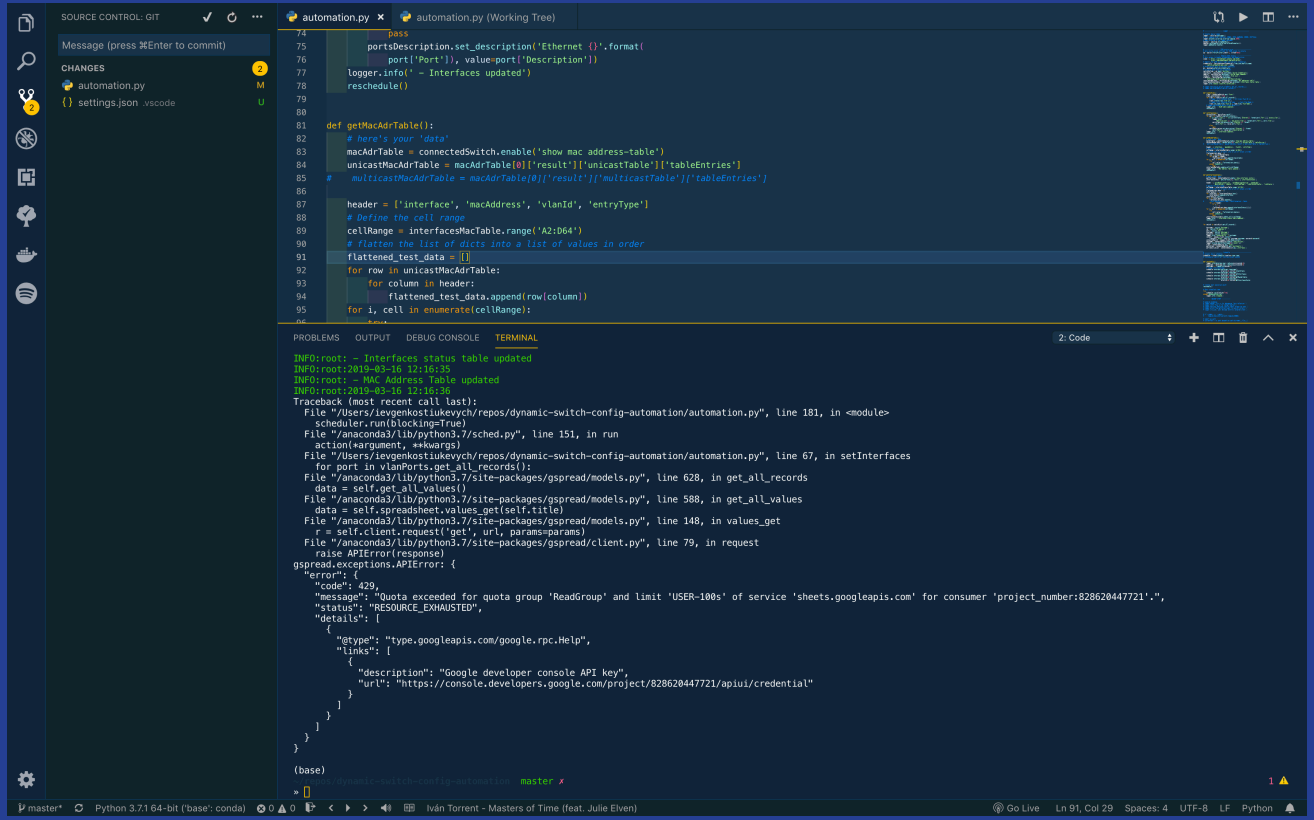
EBU

OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion


WHAT ABOUT REAL TIME
UPDATE?

WHILE TRUE:



EBU

OPERATING EUROVISION AND EURORADIO

WORKAROUND

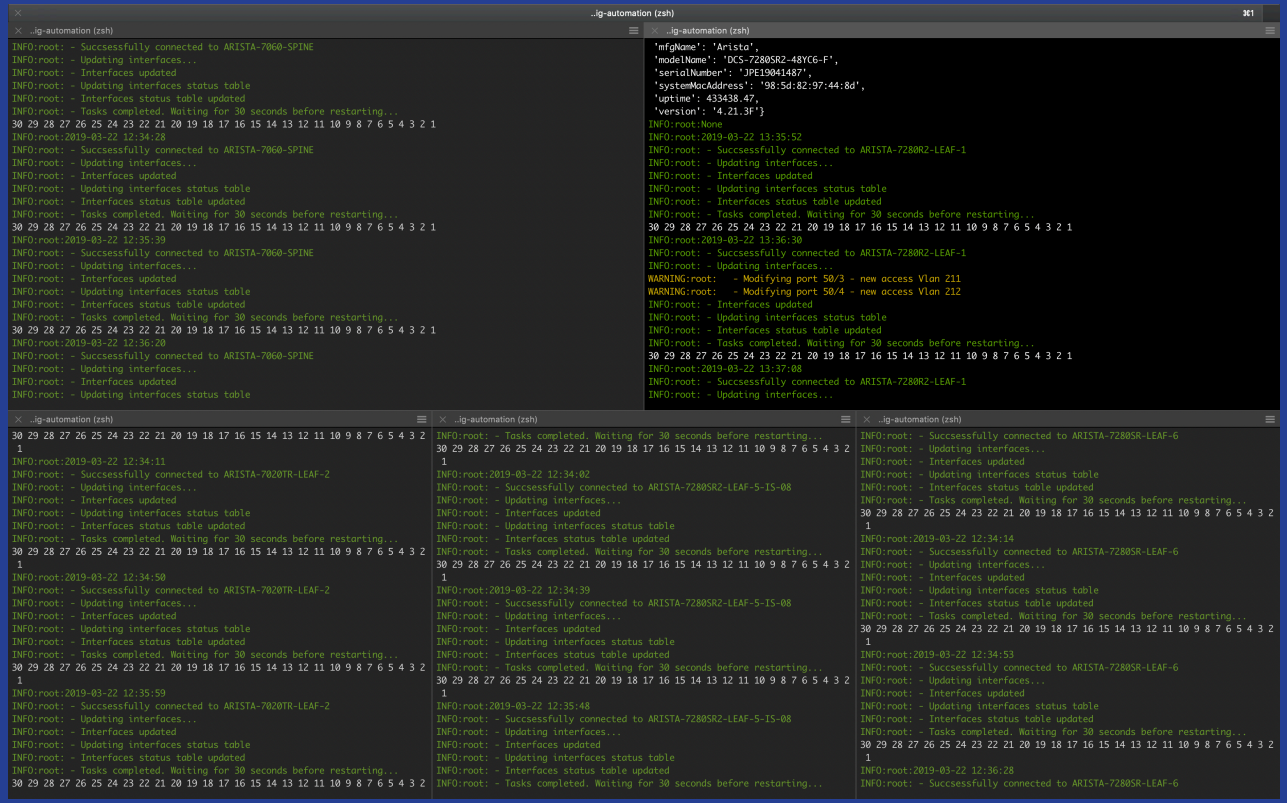
```
# Tasks repetition when continious flag is set
# Each task can be interrupted by the Google API exhaustion error. 1 minute wait is introduced.

try:
    while args.continious:
        logger.info(' - ' + str(datetime.now().replace(microsecond=0)))
        logger.info(' - Succsessfully connected to ' +
                    hostname[0]['result']['hostname'])
        if args.vlans_list:
            try:
                createVlans()
            except gspread.exceptions.APIError:
                wait('api')
                gc = googleAuthorize()
        if args.interfaces_description:
            try:
                setInterfaceDescriptions()
            except gspread.exceptions.APIError:
                wait('api')
                gc = googleAuthorize()
        if args.interfaces_vlans:
            try:
                setInterfaceVlans()
            except gspread.exceptions.APIError:
                wait('api')
                gc = googleAuthorize()
        if args.interfaces_status:
            try:
                getInterfacesState()
            except gspread.exceptions.APIError:
                wait('api')
                gc = googleAuthorize()
        wait('wait')
except KeyboardInterrupt:
    logger.info('Stopped')
```

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

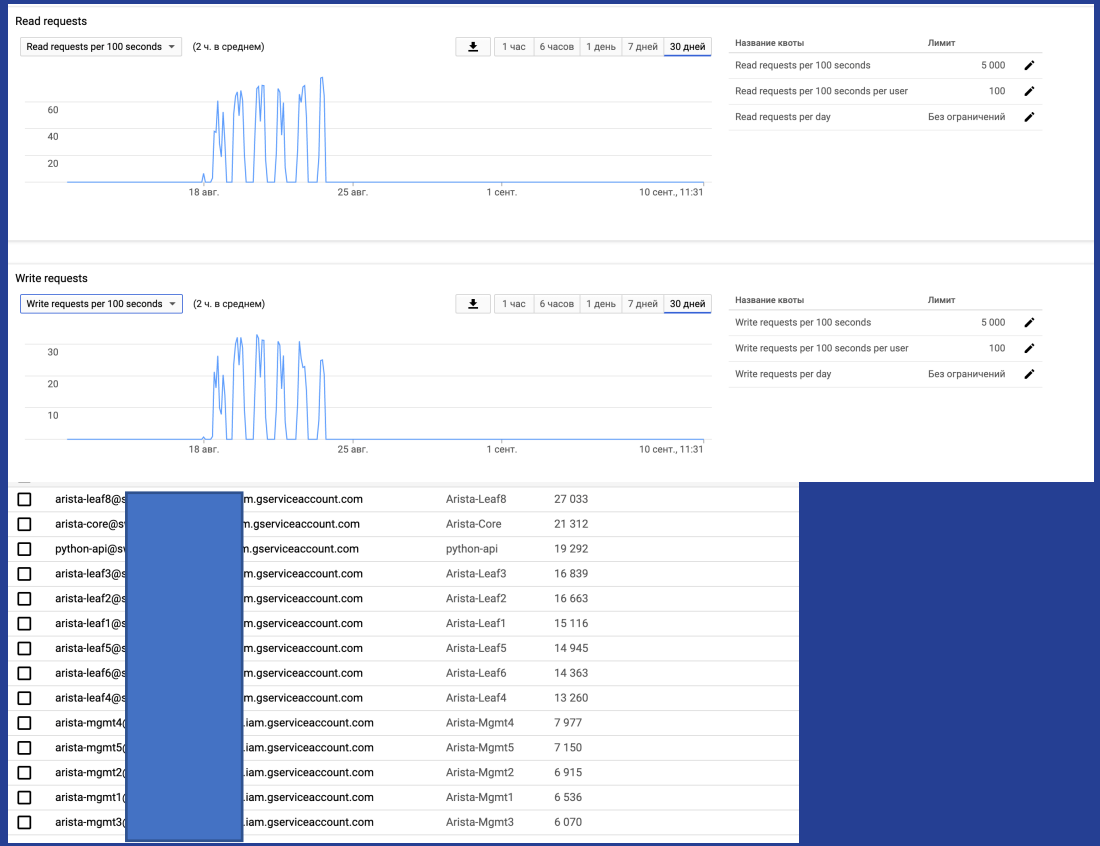
WORKAROUND



*Taken from the JT-NM Tested March 2019

EBU
OPERATING EUROVISION AND EURORADIO

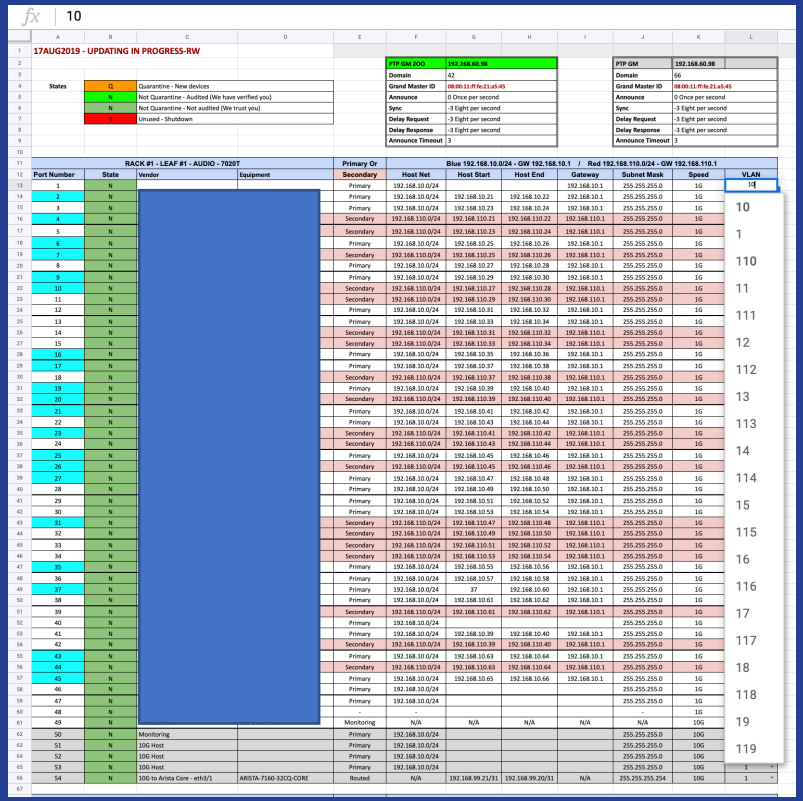
API USAGE MONITORING



EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

PRODUCTION CONTROL FRONT-END



17AUG2019 - UPDATING IN PROGRESS-RW

States		
	Q	Quarantine - New devices
	N	Not Quarantine - Audited (We have verified you)
	N	Not Quarantine - Not audited (We trust you)
	S	Unused - Shutdown

PTP GM ZOO	192.168.60.98
Domain	42
Grand Master ID	08:00:11:ff:fe:21:a5:45
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

PTP GM	192.168.60.98
Domain	66
Grand Master ID	08:00:11:ff:fe:21:a5:45
Announce	0 Once per second
Sync	-3 Eight per second
Delay Request	-3 Eight per second
Delay Response	-3 Eight per second
Announce Timeout	3

RACK #1 - LEAF #1 - AUDIO - 7020T

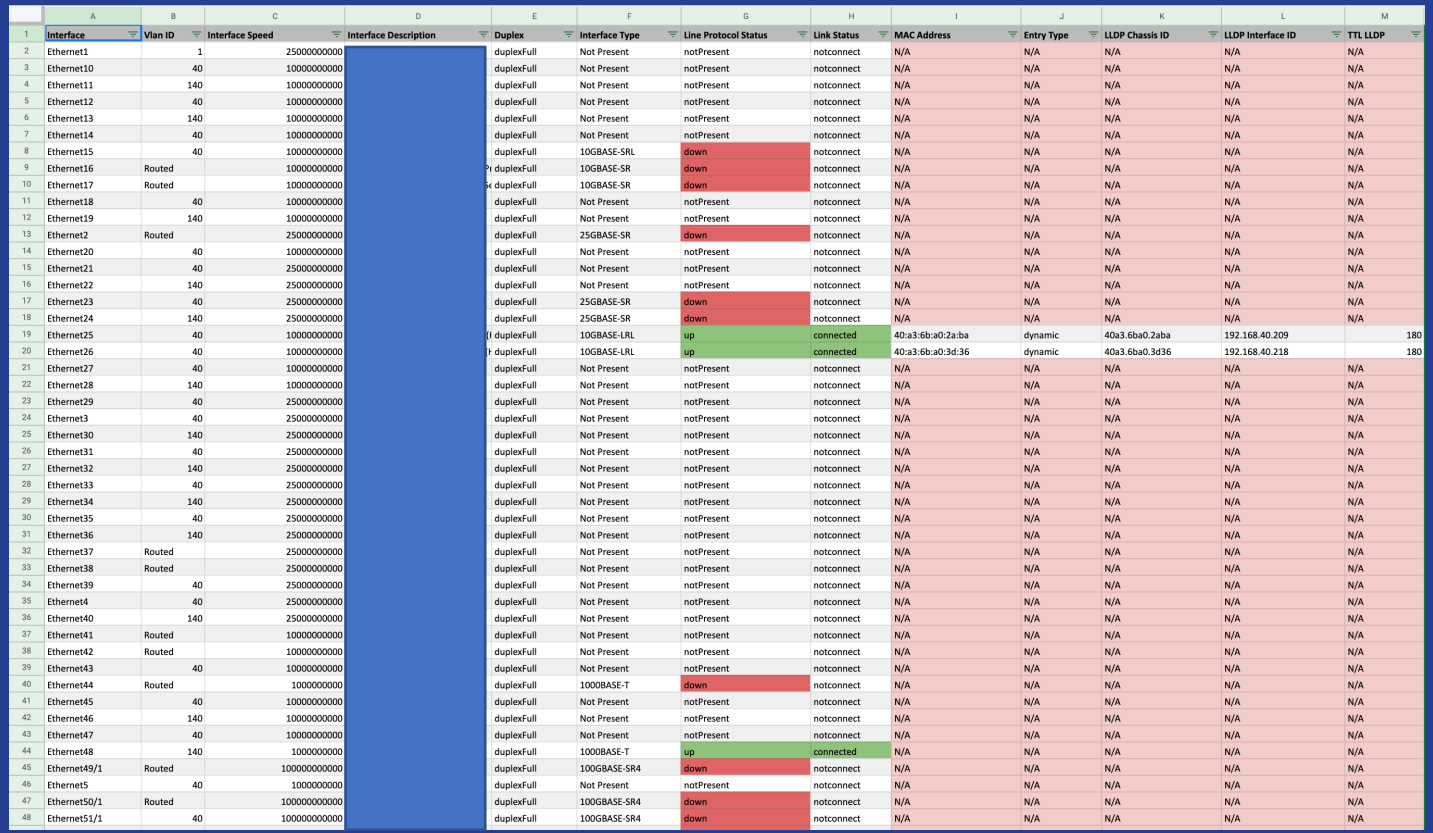
Blue 192.168.10.0/24 - GW 192.168.10.1 / Red 192.168.110.0/24 - GW 192.168.110.1

Port Number	State	Vendor	Equipment	Primary Or Secondary	Host Net	Host Start	Host End	Gateway	Subnet Mask	Speed	VLAN
1	N			Primary	192.168.10.0/24			192.168.10.1	255.255.255.0	1G	10
2	N			Primary	192.168.10.0/24	192.168.10.21	192.168.10.22	192.168.10.1	255.255.255.0	1G	
3	N			Primary	192.168.10.0/24	192.168.10.23	192.168.10.24	192.168.10.1	255.255.255.0	1G	
4	N			Secondary	192.168.110.0/24	192.168.110.21	192.168.110.22	192.168.110.1	255.255.255.0	1G	
5	N			Secondary	192.168.110.0/24	192.168.110.23	192.168.110.24	192.168.110.1	255.255.255.0	1G	
6	N			Primary	192.168.10.0/24	192.168.10.25	192.168.10.26	192.168.10.1	255.255.255.0	1G	
7	N			Secondary	192.168.110.0/24	192.168.110.25	192.168.110.26	192.168.110.1	255.255.255.0	1G	
8	N			Primary	192.168.10.0/24	192.168.10.27	192.168.10.28	192.168.10.1	255.255.255.0	1G	
9	N			Primary	192.168.10.0/24	192.168.10.29	192.168.10.30	192.168.10.1	255.255.255.0	1G	
10	N			Secondary	192.168.110.0/24	192.168.110.27	192.168.110.28	192.168.110.1	255.255.255.0	1G	
11	N			Secondary	192.168.110.0/24	192.168.110.29	192.168.110.30	192.168.110.1	255.255.255.0	1G	
12	N			Primary	192.168.10.0/24	192.168.10.31	192.168.10.32	192.168.10.1	255.255.255.0	1G	
13	N			Primary	192.168.10.0/24	192.168.10.33	192.168.10.34	192.168.10.1	255.255.255.0	1G	
14	N			Secondary	192.168.110.0/24	192.168.110.31	192.168.110.32	192.168.110.1	255.255.255.0	1G	
15	N			Secondary	192.168.110.0/24	192.168.110.33	192.168.110.34	192.168.110.1	255.255.255.0	1G	
16	N			Primary	192.168.10.0/24	192.168.10.35	192.168.10.36	192.168.10.1	255.255.255.0	1G	
17	N			Primary	192.168.10.0/24	192.168.10.37	192.168.10.38	192.168.10.1	255.255.255.0	1G	
18	N			Secondary	192.168.110.0/24	192.168.110.37	192.168.110.38	192.168.110.1	255.255.255.0	1G	
19	N			Primary	192.168.10.0/24	192.168.10.39	192.168.10.40	192.168.10.1	255.255.255.0	1G	
20	N			Secondary	192.168.110.0/24	192.168.110.39	192.168.110.40	192.168.110.1	255.255.255.0	1G	
21	N			Primary	192.168.10.0/24	192.168.10.41	192.168.10.42	192.168.10.1	255.255.255.0	1G	
22	N			Primary	192.168.10.0/24	192.168.10.43	192.168.10.44	192.168.10.1	255.255.255.0	1G	
23	N			Secondary	192.168.110.0/24	192.168.110.41	192.168.110.42	192.168.110.1	255.255.255.0	1G	
24	N			Secondary	192.168.110.0/24	192.168.110.43	192.168.110.44	192.168.110.1	255.255.255.0	1G	
25	N			Primary	192.168.10.0/24	192.168.10.45	192.168.10.46	192.168.10.1	255.255.255.0	1G	
26	N			Secondary	192.168.110.0/24	192.168.110.45	192.168.110.46	192.168.110.1	255.255.255.0	1G	
27	N			Primary	192.168.10.0/24	192.168.10.47	192.168.10.48	192.168.10.1	255.255.255.0	1G	
28	N			Primary	192.168.10.0/24	192.168.10.49	192.168.10.50	192.168.10.1	255.255.255.0	1G	
29	N			Primary	192.168.10.0/24	192.168.10.51	192.168.10.52	192.168.10.1	255.255.255.0	1G	
30	N			Primary	192.168.10.0/24	192.168.10.53	192.168.10.54	192.168.10.1	255.255.255.0	1G	
31	N			Secondary	192.168.110.0/24	192.168.110.47	192.168.110.48	192.168.110.1	255.255.255.0	1G	
32	N			Secondary	192.168.110.0/24	192.168.110.49	192.168.110.50	192.168.110.1	255.255.255.0	1G	
33	N			Secondary	192.168.110.0/24	192.168.110.51	192.168.110.52	192.168.110.1	255.255.255.0	1G	
34	N			Secondary	192.168.110.0/24	192.168.110.53	192.168.110.54	192.168.110.1	255.255.255.0	1G	
35	N			Primary	192.168.10.0/24	192.168.10.55	192.168.10.56	192.168.10.1	255.255.255.0	1G	
36	N			Primary	192.168.10.0/24	192.168.10.57	192.168.10.58	192.168.10.1	255.255.255.0	1G	
37	N			Primary	192.168.10.0/24	37	192.168.10.60	192.168.10.1	255.255.255.0	1G	
38	N			Primary	192.168.10.0/24	192.168.10.61	192.168.10.62	192.168.10.1	255.255.255.0	1G	
39	N			Secondary	192.168.110.0/24	192.168.110.61	192.168.110.62	192.168.110.1	255.255.255.0	1G	
40	N			Primary	192.168.10.0/24				255.255.255.0	1G	
41	N			Primary	192.168.10.0/24	192.168.10.39	192.168.10.40	192.168.10.1	255.255.255.0	1G	
42	N			Secondary	192.168.110.0/24	192.168.110.39	192.168.110.40	192.168.110.1	255.255.255.0	1G	
43	N			Primary	192.168.10.0/24	192.168.10.63	192.168.10.64	192.168.10.1	255.255.255.0	1G	
44	N			Secondary	192.168.110.0/24	192.168.110.63	192.168.110.64	192.168.110.1	255.255.255.0	1G	
45	N			Primary	192.168.10.0/24	192.168.10.65	192.168.10.66	192.168.10.1	255.255.255.0	1G	
46	N			Primary	192.168.10.0/24				255.255.255.0	1G	
47	N			Primary	192.168.10.0/24				255.255.255.0	1G	
48	N			-	-	-	-	-	-	1G	
49	N			Monitoring	N/A	N/A	N/A	N/A	N/A	10G	
50	N	Monitoring		Primary	192.168.10.0/24				255.255.255.0	10G	
51	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	
52	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	
53	N	10G Host		Primary	192.168.10.0/24				255.255.255.0	10G	
54	N	10G to Arista Core - eth3/1	ARISTA-7160-32CQ-CORE	Routed	N/A	192.168.99.21/31	192.168.99.20/31	N/A	255.255.255.254	10G	

10
1
110
11
111
12
112
13
113
14
114
15
115
16
116
17
117
18
118
19
119

EBU
OPERATING EUROVISION AND EURORADIO

PRODUCTION “TELEMETRY” FRONT-END



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Interface	Vlan ID	Interface Speed	Interface Description	Duplex	Interface Type	Line Protocol Status	Link Status	MAC Address	Entry Type	LLDP Chassis ID	LLDP Interface ID	TTL LLDP
2	Ethernet1	1	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
3	Ethernet10	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
4	Ethernet11	140	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
5	Ethernet12	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
6	Ethernet13	140	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
7	Ethernet14	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
8	Ethernet15	40	10000000000		duplexFull	10GBASE-SRL	down	notconnect	N/A	N/A	N/A	N/A	N/A
9	Ethernet16	Routed	10000000000		duplexFull	10GBASE-SR	down	notconnect	N/A	N/A	N/A	N/A	N/A
10	Ethernet17	Routed	10000000000		duplexFull	10GBASE-SR	down	notconnect	N/A	N/A	N/A	N/A	N/A
11	Ethernet18	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
12	Ethernet19	140	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
13	Ethernet2	Routed	25000000000		duplexFull	25GBASE-SR	down	notconnect	N/A	N/A	N/A	N/A	N/A
14	Ethernet20	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
15	Ethernet21	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
16	Ethernet22	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
17	Ethernet23	40	25000000000		duplexFull	25GBASE-SR	down	notconnect	N/A	N/A	N/A	N/A	N/A
18	Ethernet24	140	25000000000		duplexFull	25GBASE-SR	down	notconnect	N/A	N/A	N/A	N/A	N/A
19	Ethernet25	40	10000000000		duplexFull	10GBASE-LRL	up	connected	40:a3:6b:a0:2a:ba	dynamic	40a3.6ba0.2aba	192.168.40.209	180
20	Ethernet26	40	10000000000		duplexFull	10GBASE-LRL	up	connected	40:a3:6b:a0:3d:36	dynamic	40a3.6ba0.3d36	192.168.40.218	180
21	Ethernet27	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
22	Ethernet28	140	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
23	Ethernet29	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
24	Ethernet3	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
25	Ethernet30	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
26	Ethernet31	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
27	Ethernet32	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
28	Ethernet33	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
29	Ethernet34	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
30	Ethernet35	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
31	Ethernet36	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
32	Ethernet37	Routed	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
33	Ethernet38	Routed	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
34	Ethernet39	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
35	Ethernet4	40	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
36	Ethernet40	140	25000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
37	Ethernet41	Routed	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
38	Ethernet42	Routed	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
39	Ethernet43	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
40	Ethernet44	Routed	1000000000		duplexFull	1000BASE-T	down	notconnect	N/A	N/A	N/A	N/A	N/A
41	Ethernet45	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
42	Ethernet46	140	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
43	Ethernet47	40	10000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
44	Ethernet48	140	1000000000		duplexFull	1000BASE-T	up	connected	N/A	N/A	N/A	N/A	N/A
45	Ethernet49/1	Routed	100000000000		duplexFull	100GBASE-SR4	down	notconnect	N/A	N/A	N/A	N/A	N/A
46	Ethernet5	40	1000000000		duplexFull	Not Present	notPresent	notconnect	N/A	N/A	N/A	N/A	N/A
47	Ethernet50/1	Routed	100000000000		duplexFull	100GBASE-SR4	down	notconnect	N/A	N/A	N/A	N/A	N/A
48	Ethernet51/1	40	100000000000		duplexFull	100GBASE-SR4	down	notconnect	N/A	N/A	N/A	N/A	N/A

EBU
OPERATING EUROVISION AND EURORADIO

©Ievgen Kostiukevych, kostiukevych@ebu.ch, Special for AES NY AoIP Pavilion

CLI ARGS

```
usage: automation.py [-h] [-vL] [-iD] [-iS] [-iV] [-c] config api [time]

positional arguments:
  config                provide the config file
  api                   provide the API json file
  time                  amount of seconds to wait before repeating tasks,
                        default is 10 seconds

optional arguments:
  -h, --help            show this help message and exit
  -vL, --vlans_list     read vlans from the spreadsheet and create in the
                        switch
  -iD, --interfaces_description
                        read interfaces descriptions from the spreadsheet and
                        update in the switch
  -iS, --interfaces_status
                        read interfaces status from the switch and update in
                        the spreadsheet
  -iV, --interfaces_vlans
                        read interfaces vlans from the spreadsheet and update
                        in the switch
  -c, --continious      repeat activated tasks after 10 seconds (default,
                        unless time is specified)
```

EBU
OPERATING EUROVISION AND EURORADIO

USEFUL LINKS

- https://github.com/ktbyers/netmiko
- https://github.com/paramiko/paramiko
- https://docs.python.org/3.7/library/telnetlib.html
- https://pyeapi.readthedocs.io/en/latest/index.html
- https://github.com/ievgen-kostiukevych/Google-Spreadsheet-Driven-Arista-eAPI-Automation-Demo
- https://www.gns3.com/

EBU
OPERATING EUROVISION AND EURORADIO

Thank you for your time!

Ievgen Kostiukevych

kostiukevych@ebu.ch

https://www.linkedin.com/in/kostiukevych/