IETF Hackathon

IETF 123

19–20 July 2025

Madrid, Spain

YANG model for management of Network Tester

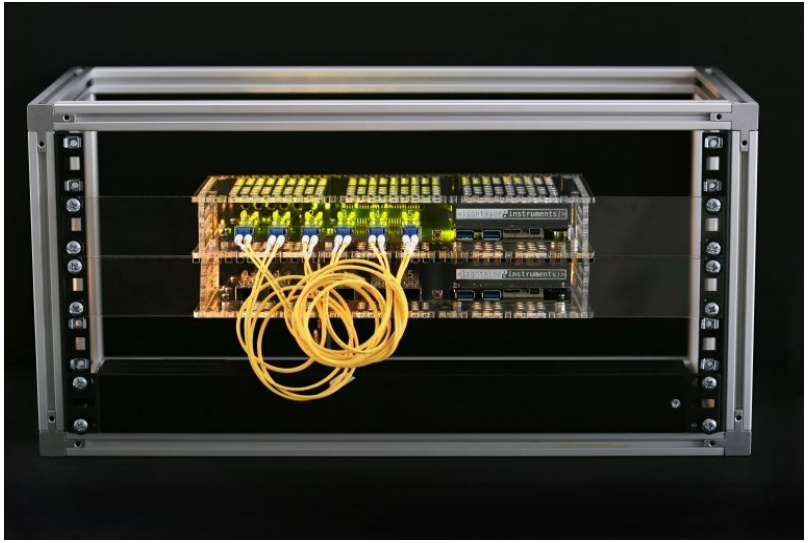
Quiz: Which 2 of the 5 network testers below have standard based management interfaces and protocols?

A

B

C

D

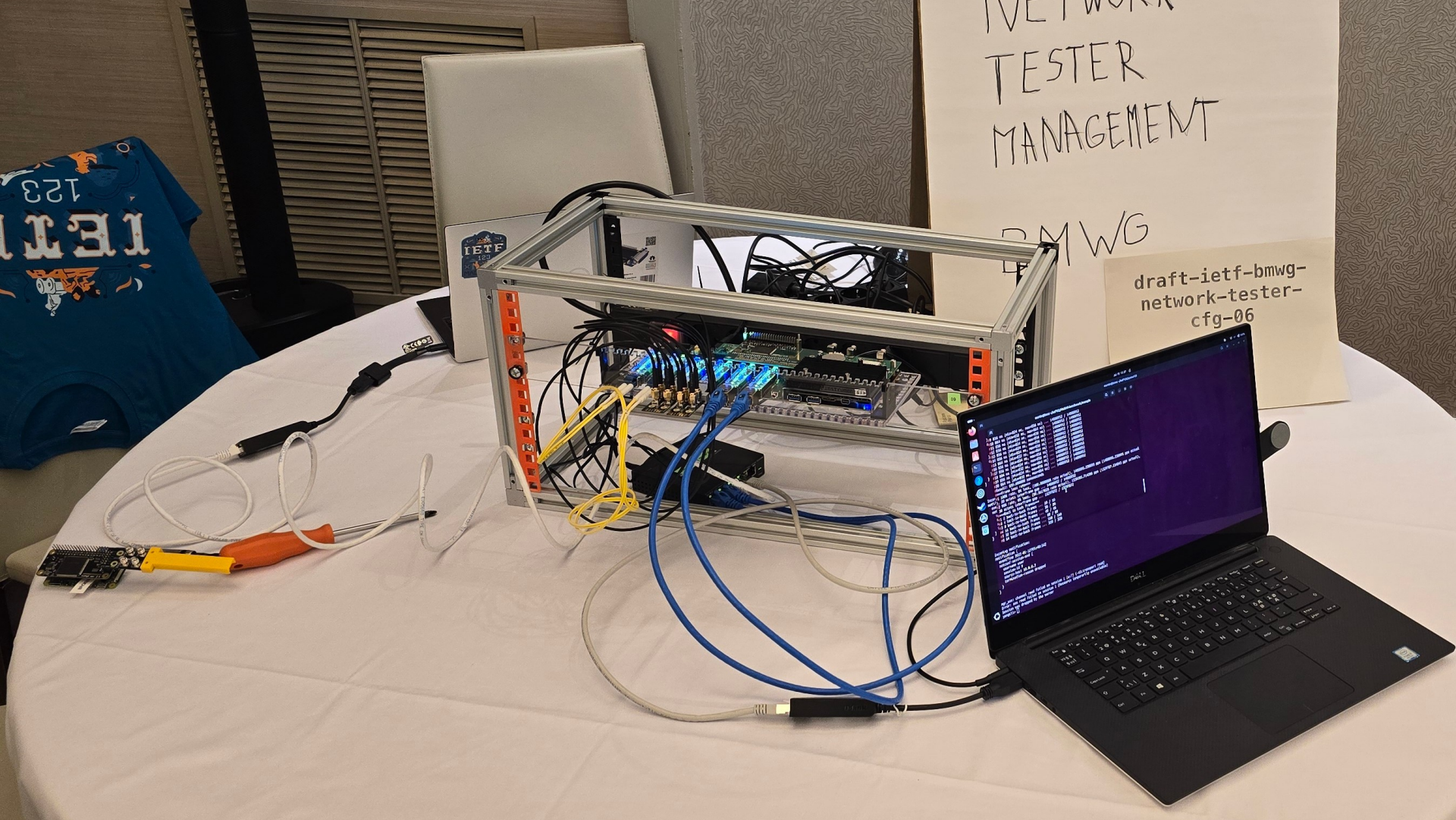


E

Champion(s)
Vladimir Vassilev (vladimir@lightside-instruments.com)
Project(s)
Model implementation with software and programmable logic for 1Gb Ethernet
RFC2544 benchmark test in python
RFC2889 benchmark test in python
TG configuration to PCAP and PCAP to TG configuration conversion scripts

Specifications
draft-ietf-bmwg-network-tester-cfg
Repositories
Scripting - YANG/NETCONF benchmark orchestration code
Software - YANG/NETCONF device side code
FPGA -HDL
Hardware - board design
Getting started - walk-through

- State of the draft (bmwg-network-tester-cfg):
  - Stable only change -05 to -06 was „e0“ -> „eth0“ as interface name in example
  - Aiming for Working Group Last Call on Monday
- State of the implementation:
  - Full support of the single stream TG capability
  - First opensource RFC2544 implementation in python
  - Both static and dynamic tesframe
  - Pre-silicon gate level simulation with cocotb


TESTER
MANAGEMENT
BMWG
draft-ietf-bmwg-
network-tester-
cfg-06

Testcase 1 (trivial): **Send 1B 64 octet sized RFC2544 specified testframes with minimal interframe gap**

```
create /interfaces/interface[name='eth0']/traffic-generator --
frame-size=64 interframe-gap=20 frame-
data=6CA96F0000026CA96F000010800450000 2ED4A500000A115816C0000201
C0000202C020007001A000001020304050607080 90A0B0C0D0E0F101112
total-frames=1000000000
commit
```

Testcase 2: **Same as #1 but 1 in million is with bad CRC**

Solution 1: **Superposition of traffic generators**

```
create /interfaces/interface[name='eth0']/traffic-generator --
frame-size=64 interframe-gap=20 interburst-gap=84 frames-per-
burst=999999 frame-
data=6CA96F0000026CA96F00000108004500002ED4A500000A115816C0000201
C0000202C0200007001A00000102030405060708090A0B0C0D0E0F101112
total-frames=999999999
create /interfaces/interface[name='eth0.1']/traffic-generator --
frame-size=64 interframe-gap=83999916 frame-
data=6CA96F0000026CA96F00000108004500002ED4A500000A115816C0000201
C0000202C0200007001A00000102030405060708090A0B0C0D0E0F101112FBADB
EEF
total-frames=999999999
```

Solution 2: **Multistream mode**

```
create /interfaces/interface[name='eth0']/traffic-generator -- total-frames=1000000000
create /interfaces/interface[name='eth0']/traffic-generator/stream[id='1'] --
frame-size=64 interframe-gap=20 frame-
data=6CA96F0000026CA96F0000108004500002ED4A500000A115816C0000201C000
0202C0200007001A000001020304050607080 90A0B0C0D0E0F101112
interstream-gap=20 frames-per-stream=999999
create /interfaces/interface[name='eth0']/traffic-generator/stream[id='2'] --
frame-size=64 interframe-gap=20 frame-
data=6CA96F0000026CA96F0000108004500002ED4A500000A115816C0000201C000
0202C0200007001A000001020304050607080 90A0B0C0D0E0F101112FBADBEEF
interstream-gap=20 frames-per-stream=1
commit
```

END