Track 2: Cloud and Wired networking

Thijs Walcarius

imec – Ghent University, Belgium

Porto Roadshow
Porto - February 18, 2020
Table of Contents

CLOUD

• **ESpec**: The Experiment Specification

• Easy deployment of **OpenStack** with EnOS

Wired Networking

• Workflow for scaling up experiments

• Scaling up networking experiments with **Shared LAN’s**

• **Examples** of Advanced Networking Experiments
The Experiment Specification
What is an Experiment Specification?

Espec bundles:

Resource Specification

Files to be uploaded

Commands to be executed

Extras: SSH, RNG
Deploying OpenStack with EnOS
Deploying OpenStack with EnOS

EnOS allows you to **Deploy**, **Customize** and **Benchmark** OpenStack

- Developed by Inria
- Wrapper around **Kolla-Ansible**
- Deploys all OS-services as Docker containers
- ESpec generates the EnOS config file for bootstrapping the deployment
OpenStack experiment architecture

private

public

NAT

192.168.0.202
192.168.0.201
192.168.0.200

compute

network

control

VM

Floating "public IP": 192.168.10.XYZ

192.168.10.1
Resources on EnOS

Tutorial


EnOS documentation

https://enos.readthedocs.io/

EnOS ESpec repository

https://gitlab.ilabt.imec.be/ilabt/enos-espec/
Workflow for creating and scaling up experiments
Recommended workflow

1. Build the smallest reasonable topology
2. Orchestrate and automate
3. Iterate and save your progress
4. Scale your experiment
1. Build the smallest reasonable topology
2. Automate as you go

Use Configuration Management Systems to automate installation and configuration of software

Many tools available for this job: Ansible, Chef, Puppet, …
3. Save your progress

Log all of your experimental artifacts for every experiment that works

- RSpec
- image
- install script
- custom software
- measurements
- etc.

Use version control to store your artifacts

Always know the last configuration that worked
4. Scale your experiment

Only scale up when your smallest reasonable experiment is working

Adapt your request RSpec to add more nodes

- Roll your own scaling script: mostly copy/pasting with minimal editing required
- Use geni-lib
Scaling up experiments with shared LAN’s
Shared LAN

Shared LAN’s allow you to add extra servers to an existing network in an experiment

Step 1: Right click on network and choose “Share/Unshare Lan”
Shared LAN

**Step 2:** Design a new experiment with extra servers
Shared LAN

Step 3: Fix duplicate IP-addresses
Step 5: Configure link in new experiment to connect to existing Shared LAN

**Properties of link0**

- **Link type:**
  - lan

**NOTE:** Known supported link types for this link are: gre-tunnel, egre-tunnel, lan

**NOTE:** (e)gre-tunnel only works when all nodes have a public IPv4 address

- **Connect to existing Shared LAN:**
  - demo_shared_lan

- **Make into a new Shared LAN:**
  - None selected
  - Use real VLAN
  - No MAC learning
Shared LAN

Step 6: Start the new experiment
Test your links!
Examples of advanced networking experiments
Creating multi-testbed experiments

1. Start with creating a backbone
2. Scale up with shared LAN’s on each site after that
route add -net 192.168.1.0/24 gw 192.168.0.2
Advanced SDN experiment
Documentation

FED4FIRE+

Testbeds Overview
https://www.fed4fire.eu/testbeds/

Technical Documentation
https://doc.fed4fire.eu/
This project has received funding from the European Union’s Horizon 2020 research and innovation programme, which is co-funded by the European Commission and the Swiss State Secretariat for Education, Research and Innovation, under grant agreement No 732638.

WWW.FED4FIRE.EU