

Building Digital Twins

with Containerlab – using ntop tools and Wireshark for traffic analysis.

> Mischa Diehm CTO narrowin.ch



Who?

Mischa Diehm

- Founder of narrowin
- Network design and development
- Computer and network infrastructure

narrowin

- Networking and security
- Micro-/Endpoint segmentation
- Lightweight Network Explorer

https://demo.narrowin.ch





eniwa

— ℇոՑሠ

Kantonsspital Baselland ganz nah

What can I use a Digital Twin of my Network for?



Education

Running in containerlabs



Introducing Containerlab

_	<	>								
💧 conti	ainerlak	, ,					D Q Search	1		C sri-labs/cd
Home Insta	llation			er manual (Lab example:	s Release note			
Home										
										Features Use cases
				,	$\neg \square$					
				, c	>					
release v0.64.0	downloads 2	Ak M follow C	maineriati (1) disc	ord 255 online						
With the grow										
	vina num	ber of conta	inerized Netv	work Operatir	na Svstems arows t	he demand to e	asily run them in	the user-def	ined.	
versatile lab to	ving num opologie	ber of conta s.	ainerized Netv					the user-def		
versatile lab to	ving num opologie v, contain	ber of conta s. er orchestra	ainerized Netw	work Operatir docker-com	ng Systems grows t pose are not a goo			the user-def		
versatile lab tr Unfortunately, easily create of	ving num opologie , contain connecti	ber of conta s. er orchestra ons betweer	inerized Netv ition tools like h the containe	work Operatir docker-com ers which defi	ng Systems grows t pose are not a goo ine a topology.			the user-def		
versatile lab te Unfortunately easily create c Containerlab p virtual wiring te	ving num opologie , contain connecti provides between	ber of conta s. er orchestra ons betweel a CLI for or them to cre	ainerized Netw ation tools like in the containe chestrating as ate lab topole	work Operatir docker-com ers which defi nd managing ogies of users	ng Systems grows t pose are not a goo ine a topology. container-based n ; choice and manag	he demand to e d fit for that pur atworking labs. I jes labs lifecycle		the user-def not allow a u ainers, builds		
versatile lab ti Unfortunately, easily create o Containerlab j virtual wiring b	ving num opologie , contain connecti provides between	ber of conta s. er orchestra ons betweel a CLI for or them to cre	inerized Netv ition tools like n the containe chestrating a rate lab topole	work Operatir I docker-com ers which defi nd managing ogies of users	ng Systems grows t pose are not a goo ine a topology. container-based n : choice and manag	he demand to e d fit for that pur atworking labs. I jes labs lifecycle		i the user-def o not allow a u ainers, builds		
versatile lab te Unfortunately easily create o Containerlab j virtual wiring t	ving num opologie , contain connecti provides between between spine	ber of conta s. er orchestra ons between a CLI for or them to cre	ation tools like In the containe chestrating a Rate lab topole	work Operatin e docker-com ars which defi nd managing ogies of users	ng Systems grows t pose are not a goo ine a topology. container-based n : choice and manag	he demand to e d fit for that pur atworking labs. les labs lifecycle	asily run them in pose, as they do it starts the cont	i the user-def p not allow a u ainers, builds		
versatile lab tr Unfortunately easily create c Containerlab j virtual wiring t	ving num opologie r, contain connecti provides between between	ber of conta s. er orchestra ons betweet a CLI for on them to cre	ation tools like ation tools like the containe chestrating ar ate lab topole	work Operatin e docker-com ers which defi nd managing ogies of users	ng Systems grows t pose are net a goo ne a topology. container-based n choice and manaç	he demand to e d fit for that pur atworking labs. I ses labs lifecycle	asily run them in pose, as they do t starts the cont SRLinux	the user-def e not allow a u ainers, builds		
versatile lab tr Unfortunately easily create c Containerlab (virtual wiring t S	ving num opologie , contain connecti provides between between spine	ber of conta s. er orchestra ons between a CLI for or them to cre	ation tools like In the containe chestrating a rate lab topole	vork Operatin e docker-com ers which defi nd managing ogies of users	ng Systems grows t pose are not a goo ne a topology. container-based n choice and manag	he demand to e d fit for that pur atworking labs. labs lifecycle	asily run them in pose, as they do t starts the cont BRLinux	the user-def o not allow a u ainers, builds		
versatile lab te Unfortunately easily create of Containerlab j virtual wiring t	ving num opologie ; contain connecti provides between spine	ber of conta s. er orchestra a CLI for or them to cre	ainerized Netwation tools like at the contained of the co	work Operatin	ng Systems grows t pose are not a goo ne a topology. container-based n c choice and manag	he demand to e d fit for that pur etworking labs. les labs lifecycle	asily run them in pose, as they do t starts the cont SRLinux AristacEOS	the user-def o not allow a u ainers, builds		
versatile lab te Unfortunately easily create o Containerlab virtual wiring t S S	ving num opologie , contain connecti provides between super spine spine	ber of conta s. er orchestra ons between a CLI for or them to cre	ation tools like In the containe chestrating at ate lab topok	work Operatin	ng Systems grows t pose are not a goo ne a topology. container-based n choice and manag	he demand to e d fit for that pur atworking labs. les labs lifecycle SR Linux	asily run them in pose, as they do t starts the cont SR Linux AristacEOS	the user-def e not allow a u ainers, builds		
versatile lab t Unfortunately easily create d Containerlab virtual wiring t S S	ving num opologie ; contain connecti provides between spine spine	ber of conta s. er orchestra ons between a CLI for or them to cre	tion tools like the contained chestrating ar ate lab topok	work Operatin docker-com ars which defi ind managing ogles of users interpretation gless	ng Systems grows t pose are not a goo ne a topology. container-based n choice and manag	the demand to e d fit for that pur stworking labs. I es labs lifecycle sR linux sR Linux rbitrarily co	asily run them in pose, as they do t starts the cont - - - - - - - - - - - - - - - - - - -	the user-def e not allow a u ainers, builds nux pologi		
versatile iak u Unfortunately easily create c Containerlab virtual wiring t S S Containerlab 1	ving num opologie , contain connecti provides between spine spine spine	ber of conta s. er orchestra ons between a CLI for or them to cre CLO on the conta	ation tools like in the contained chestrating at ate lab topole	work Operatin docker-com ars which defind and managing ogies of users	ng Systems grows to pose are not a goo ne a topology. container-based n s choice and manac	the demand to e d fit for that pur stworking labs. I es labs lifecycle site interview site inter	asily run them in pose, as they do t starts the cont BRUNAX Arita eCOS SRUMAX Arita eCOS	nux pologi contallow a u ainers, builds pologi contactor contactor pologi		
versatile lab t Unfortunately easily create c Containerlab ; virtual wiring t S S S Containerlab 1 designs, such	ving num opologie , contain connecti provides between super spine ear spine focuses as:	ber of conta s. er orchestra a CLI for on them to cre b ccLO on the conta	ation tools like in the contained chestrating at ate lab topole	work Operatin el docker-com ars which defi nd managing ogies of users	ng Systems grows to pose are not a goo container-based ni choice and manaç	the demand to e d fit for that pur stworking labs. I es labs lifecycle sit linue sit linue sit linue sit linue re typically used	asily run them in pose, as they do t starts the cont SRUmax Anita eCOS sRU Anita eCOS	nux pologi creatures and		

https://containerlab.dev

«Containerlab provides a CLI and GUI for orchestrating and managing container-based networking labs.

It starts the containers, builds a virtual wiring between them to create lab topologies of users' choice and manages labs lifecycle.»

- ✓ Covers many vendors
- ✓ Declarative by nature
 - Easy topology definition
- ✓ Scales really well

Containerlab workflow



Where do I get a container Image? Containerized NOSes



- Sourced by the vendor
- Fast and easy to use

The trend is to move away from VM packaging towards containers. Still, many NOS are VM-based.

Where do I get a container Image? Containerizing VM-based NOSes



- Use vrnetlab to containerize
- Launch topologies with VM-based NOS within the same topology definition file, alongside containerized NOS.
- > 30 NOS kinds supported

https://github.com/hellt/vrnetlab

https://containerlab.dev/manual/vrnetlab/#supportedvm-products

Important: Containerlab uses original vrnetlab project fork hellt/vrnetlab. Container built with upstream vrnetlab project will not be compatible with Containerlab.

Containerlab basics: Topology file definition

```
topology:
kinds:
  mikrotik_ros:
    image: ghcr.io/narrowin/vrnetlab_mikrotik_routeros:7.18
  linux:
    image: ghcr.io/network-unit-testing-system/nuts-testclient:0.0.2
    env:
      ADMIN_PASSWORD: admin
nodes:
  # SWITHCES
  sw-acc1:
    kind: mikrotik_ros
    mgmt-ipv4: 10.10.1.11
    startup-config: startup-configs/sw-acc1.rsc
    env:
      CLAB_MGMT_PASSTHROUGH: "true"
  # ENDPOINTS / CLIENTS
  linux1:
    kind: linux
    mgmt-ipv4: 10.10.1.101
    exec:
      - ip address add 10.1.1.1/24 dev eth1
  linux2:
    kind: linux
    mgmt-ipv4: 10.10.1.102
    exec:
      - ip address add 10.1.1.2/24 dev eth1
   ntap1:
```

clab deploy	deploy the topology (start the lab).				
clab destroy	shut down the lab.				
ssh clab-mylab-mkt1	connect to the node.				

Containerlab creates static entries in the /etc/hosts file and sets up /etc/ssh_config.d/ to allow you to use SSH.

Live Demo / Screencast

Pray to the demo gods



Debugging with Wireshark

Command Line

Executing the capture script

~/bin/clab_pcap.sh cs.foo clab-s3n-sw-acc2 ether2

... execs:

ssh cs.foo 'sudo ip netns exec clab-s3n-sw-acc2 tshark -l -i ether2 -w -' | /usr/bin/wireshark -k -i -

GUI

- Edgeshark general stand-alone virtual network/communication diagnosis tool
- Can comfortably capture live container network traffic in Wireshark, using the csharg external capture plugin for Wireshark

Ntop tooling

NTOPNG

- Containerizes setups: <u>https://github.com/ntop/docker-ntop</u>
- adapt: docker-compose.yml

services:

...

...

ntopng:

- image: ntop/ntopng:stable
- + image: ntopng-dev
- command: ['-i', 'tcp://*:5556c', '-F', 'clickhouse', '--disable-login']

+ command: ['--license-mgr', '/etc/ntopng-Im-client.conf', '-w', '127.0.0.1:3333','-i', 'ntap:5678:secret', '-i', 'tcp://*:5556c', '-F', 'clickhouse', '--disable-login', '0']

NTAP

Change Dockerfile.ntap.dev

-ENTRYPOINT ["/run.sh"] +ENTRYPOINT ["/bin/bash"]

- Tweak clab node definition

ntap1: kind: linux image: ntap-clab mgmt-ipv4: 10.10.1.103 binds: - ../setup-bridge-eth1-eth2.sh:/setup-bridge.sh cmd: ntap_remote -i eth2 -c 10.10.1.1:5678 -k secret exec: - bash /setup-bridge.sh

Transform Real Network Into Digital Twin

- Map your production network topolgy to containerlab
 - Use a software like the narrowin LNE that can generate contrainlab topos for you
 - Write/wait for tooling that taps into e.g. your SoT like netbox and does the limbo
- Use your production running configs in containerlab
 - Interface name mappings
 - Can be done if supported with interface aliases in containerlab
 - Renaming of interfaces inside the NOS itself
 - $\circ~$ HW related features possibly NOT available in virtualized NOS
 - MLAG (multi-chassis link aggregation)
 - Mirror/span ports
 - Switch stacks
- Virtual wiring leads to link states always being up (watch out when testing fail-over scenarios)
- Some NOS features might work differently on virtual NOS than on real HW (e.g. logging in CHR)

NIWOTIGN

Netmap Inventory Segmentation History Assessment 🐼

OT-WAN Lab with network services



Some useful remarks for your labs:

- Dynamic inventory automatically created for anisble and nornir
 - Labels will be translated into group membership (hopefully soon also for nornir)
- Run your labs without any local dependencies
 - $\circ~$ Local with devpod
 - Remote with github codespaces
- Share access to your labs with sshx a secure web-based, collaborative terminal
- External connectivity: <u>https://containerlab.dev/lab-examples/ext-bridge/</u>
- Containerlab API: <u>https://github.com/srl-labs/clab-api-serve</u>

Live Demo / Screencast

Pray to the demo gods



Lab examples for inspiration

- <u>https://containerlab.dev/lab-examples/lab-examples/</u> huge number of very advanced labs
- <u>https://ccie-sp.gitbook.io/ccie-spv5.1-labs</u> all labs for Cisco CCIE Service Provider v5.1
- <u>https://github.com/srl-labs/srl-telemetry-lab</u> The lab topology consists of a Clos topology, plus a Streaming Telemetry stack comprised of gnmic, prometheus and grafana applications.
- <u>https://github.com/narrowin/ansible-mikrotik/</u> Automating MikroTik Device Management with Ansible
- <u>https://containerlab.dev/manual/topo-def-file/</u> containerlab docs -> absolutely exceptional!
- <u>https://www.youtube.com/@RomanDodin</u> great vidoes on many aspects of containerlab

Thanks – stay in touch



mischa.diehm@narrowin.ch

