



nic.br
Brazilian Network
Information Center



egi.br
Brazilian Internet
Steering Committee



registro.br cert.br cetic.br ceptro.br ceweb.br ix.br



GTER54

Do GIT ao Router

From GIT to the Router

William Prado <wprado@nic.br>



nic.br

O Desafio



Config Drift

Configurações não documentadas e mudanças manuais causam divergências.



Falta de Rastreabilidade e Versionamento

Quem mudou? Quando? Por quê? Essas perguntas ficam sem resposta.



Validação Pós-Deploy

Descobrir problemas apenas em produção aumenta riscos e downtime.

NetDevOps: DevOps para Redes

Aplicando DevOps à Infraestrutura de Rede

NetDevOps traz automação, versionamento e testes para configuração e operação de redes



Network as Code

Configurações versionadas em Git, geradas por templates, testadas antes do deploy.



Automação

CI/CD pipelines validam e aplicam mudanças automaticamente em devices.



Testing

Digital twins e testes automatizados garantem qualidade antes da produção.



Traditional Network Ops

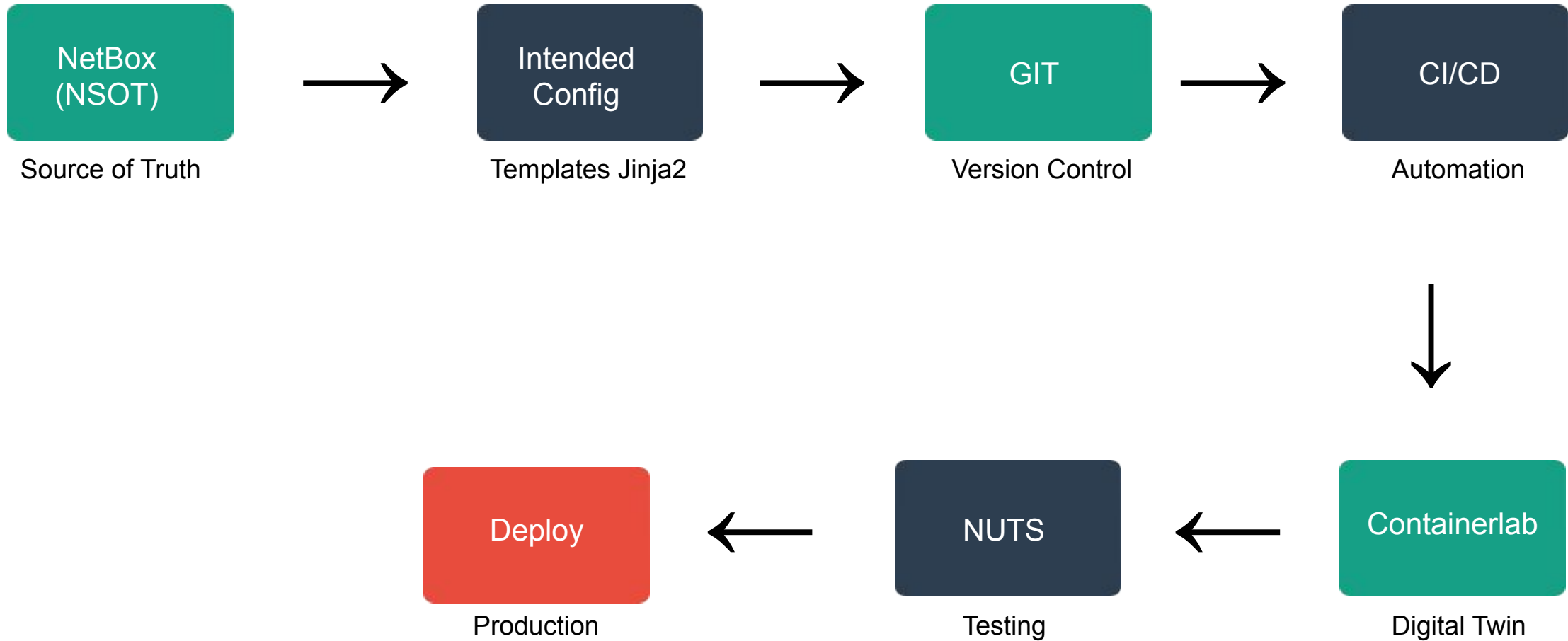
Manual, CLI, sem versionamento, validação pós-deploy.



NetDevOps

Automatizado, versionado, testado, validação pré-deploy.

Pipeline End-to-End



NetBox: Network Source of Truth



Dados Estruturados

Devices, interfaces, IPs, VLANs, etc



Config Contexts

Estrutura de Dados personalizada para representar sua Rede (ospf.json)



Templates Jinja2

Renderização das configurações do NetBox. (nokia.j2)



Branching

Estados da Rede representados por branches: main, staging, lab, etc



Resultado

Intended Configs geradas automaticamente via Script

GIT: Data Source + Trigger

netbox-data-source

Config Contexts

config-context/ospf.json

Templates

templates/nokia.j2

Intended Config

intended/*.cfg

Scripts

generate_intended.py

CI/CD - Control Plane

Intended config: git push → branch feature

Pull Request: branch feature → development

Start CI pipeline - validation.yml

Pull Request: development → main

Start CD pipeline - deploy.yml

Rastreabilidade Total

Commit → Config → Router

ContainerLAB: Digital Twin

Por que Digital Twin?

Validar mudanças em ambiente idêntico à produção antes de aplicar nos routers reais.

Topologia

3x Nokia SR Linux (PE-101, PE-102, PE-103)

Protocols

OSPF Area 0 com full mesh

Startup Config

Intended configs carregadas automaticamente

gter54.clab.yaml

```
topology:
  nodes:
    sp1-srlinux-pe-101:
      kind: nokia_srlinux
      startup-config:
        ../netbox-data-source/intended/sp1-srlinux-pe-101.cfg
```

NUTS - Network Unit Testing System

Framework de Testes Automatizados

- Validação completa do Digital Twin antes do deploy em Produção;
- Utiliza Nornir no backend para acesso aos devices;
- Existem varios test cases prontos, mas é possível criar os testes personalizados;
- SR Linux utiliza NAPALM-SRLINUX com GNMI para testes;

Interface Tests

Status operacional
Velocidade configurada
MTU correto
Endereçamento IP

ICMP Tests

Ping loopback local
Ping entre PEs
P2P connectivity

OSPF Tests

Neighbors UP
Area 0.0.0.0
FULL STATE

Garante a qualidade do deploy.

Se os testes passam no Digital Twin → Seguro para produção

Workflow

1

NetBox gera intended configs a partir de dados estruturados e templates.

2

Git versiona configurações e config-contexts, sendo source of truth e trigger.

3

CI/CD é chamado/trigger automaticamente em push/merge para validar ou fazer o deploy.

4

ContainerLab cria Digital Twin idêntico à produção com configurações aplicadas.

5

NUTS executa testes automatizados validando interfaces, conectividade e OSPF.

6

Deploy Real aplica configurações nos routers apenas se todos os testes passarem.

Fluxo de Branches



CI - Continuous Integration

Deploy ContainerLab + NUTS Testing



CD - Continuous Deployment

Deploy automático nos routers

CI/CD Pipeline com Gitea Actions

DEVELOPMENT

- ✓ Clone do repositório
- ✓ Setup Python + UV
- ✓ Deploy ContainerLab
- ✓ Testes: Interfaces
- ✓ Testes: ICMP
- ✓ Testes: OSPF
- ✓ Destroy + Cleanup

MAIN

- ✓ Deploy
- ✓ Ansible/Nornir
- ✓ Aplicação nos devices

✨ Production Ready

Self-hosted runner: Gitea + act_runner em Docker ou Nativo com acesso privilegiado para ContainerLab

E na prática?

Video no youtube:



@NetworkAutomationBR

Netbox - Devices

[Main](#) wprado Admin

Devices [+ Add](#) [Import](#) [Export](#)

Results **3** [Filters](#)

[Configure Table](#)

<input type="checkbox"/>	NAME	STATUS	TENANT	SITE	ROLE	MANUFACTURER	TYPE	IP ADDRESS	
<input type="checkbox"/>	sp1-srlinux-pe-101	Active	production	POP-SP1	PE	Nokia	SRLINUX	2001:db8:cafe::101/64	
<input type="checkbox"/>	sp2-srlinux-pe-102	Active	production	POP-SP2	PE	Nokia	SRLINUX	2001:db8:cafe::102/64	
<input type="checkbox"/>	sp3-srlinux-pe-103	Active	production	POP-SP3	PE	Nokia	SRLINUX	2001:db8:cafe::103/64	

Showing 1-3 of 3 [Per Page](#)

[+ Add Components](#) [Edit Selected](#) [Rename Selected](#) [Delete Selected](#)

Netbox - Interfaces

Search...

Main

wprado Admin

Devices / POP-SP1

dcim.device:1

sp1-srlinux-pe-101

Created 2025-12-03 19:33 · Updated 2025-12-09 22:09

+ Add Components

Bookmark

Subscribe

Edit

Delete

Device

Interfaces 6

Config Context

Render Config

Contacts

Images

Journal

Changelog

Quick search

Configure Table

	NAME	ENABLED	TYPE	PARENT	DESCRIPTION	IP ADDRESSES	CABLE	CONNECTION	TAGS	
	ethernet-1/1	✓	25GBASE-LR (25GE)	—	—		#1	sp2-srlinux-pe-102 > ethernet-1/1	—	<div>+ </div>
	ethernet-1/1.0	✓	Virtual	ethernet-1/1	—	10.0.0.13/30	—	—	ospf	<div>+ </div>
	ethernet-1/2	✓	25GBASE-LR (25GE)	—	—		#2	sp3-srlinux-pe-103 > ethernet-1/1	—	<div>+ </div>
	ethernet-1/2.0	✓	Virtual	ethernet-1/2	—	10.0.0.5/30	—	—	ospf	<div>+ </div>
	lo0.0	✓	Virtual	—	—	10.255.255.1/32	—	—	ospf	<div>+ </div>
	mgmt0	✓	1000BASE-T (1GE)	—	—	172.24.0.101/24 2001:db8:cafe::101/64	—	—	—	<div>+ </div>

Showing 1-6 of 6

Per Page

Netbox - Config Context

Devices / POP-SP1

dcim.device:1

sp1-srlinux-pe-101

Created 2025-12-03 19:33 · Updated 2025-12-09 22:09

+ Add Components

Bookmark

Subscribe

Clone

Edit

Delete

Device

Interfaces 6

Config Context

Render Config

Contacts

Images

Journal

Changelog

Rendered Context

JSON

YAML

```
{
  "ospf": {
    "area": "0.0.0.0",
    "enabled": "enable",
    "instance": "main",
    "version": "ospf-v2"
  },
  "router_id": "10.255.255.1"
}
```

Local Context

JSON

YAML

```
{
  "router_id": "10.255.255.1"
}
```

ⓘ The local config context overwrites all source contexts.

Source Contexts

JSON

YAML

ospf

1000

```
{
  "ospf": {
    "area": "0.0.0.0",
    "enabled": "enable",
    "instance": "main",
    "version": "ospf-v2"
  }
}
```

Netbox - Render Config

Rendered Config

[Download](#)



```
set / network-instance default type default
set / interface lo0 admin-state enable
set / interface lo0 subinterface 0 ipv4 address 10.255.255.1/32
set / interface lo0 subinterface 0 ipv4 admin-state enable
set / network-instance default interface lo0.0
set / interface ethernet-1/1 admin-state enable
set / interface ethernet-1/1 subinterface 0 ipv4 address 10.0.0.13/30
set / interface ethernet-1/1 subinterface 0 ipv4 admin-state enable
set / network-instance default interface ethernet-1/1.0
set / interface ethernet-1/2 admin-state enable
set / interface ethernet-1/2 subinterface 0 ipv4 address 10.0.0.5/30
set / interface ethernet-1/2 subinterface 0 ipv4 admin-state enable
set / network-instance default interface ethernet-1/2.0
set / network-instance default protocols ospf instance main admin-state enable
set / network-instance default protocols ospf instance main version ospf-v2
set / network-instance default protocols ospf instance main router-id 10.255.255.1
set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/1.0 admin-state enable
set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/1.0 interface-type point-to-point
set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/2.0 admin-state enable
set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/2.0 interface-type point-to-point
set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 admin-state enable
set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 interface-type point-to-point
set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 passive true
```

Netbox Scripts - Intended Config

Script

Source

Jobs 15

Script Data

Device*

sp1-srlinux-pe-101



This field is required.

Selecione o dispositivo

☐

Dry run

Apenas visualizar (não enviar para Git)

Script Execution Parameters

Schedule at

YYYY-MM-DD hh:mm:ss

Schedule execution of script to a set time (current time: 2025-12-13 13:59:53 UTC)

Recurs every

Interval at which this script is re-run (in minutes)

☐

Commit changes

Commit changes to the database (uncheck for a dry-run)

Cancel

▶ Run Script

Netbox Scripts - Intended Config

LINE	TIME	LEVEL	OBJECT	MESSAGE
1	2025-12-13 14:01:14	Info	—	=====
2	2025-12-13 14:01:14	Info	—	📡 Device: sp1-srlinux-pe-101
3	2025-12-13 14:01:14	Info	—	Site: POP-SP1
4	2025-12-13 14:01:14	Info	—	Tipo: SRLINUX
5	2025-12-13 14:01:14	Info	—	=====
6	2025-12-13 14:01:14	Success	—	✓ Config Template: Nokia - SR Linux
7	2025-12-13 14:01:14	Info	—	🔄 Renderizando configuração...
8	2025-12-13 14:01:18	Success	—	✓ Configuração renderizada com sucesso!
9	2025-12-13 14:01:18	Info	—	_____
10	2025-12-13 14:01:18	Info	—	Total de linhas: 23
11	2025-12-13 14:01:18	Info	—	=====
12	2025-12-13 14:01:18	Info	—	📦 ENVIANDO PARA GITEA
13	2025-12-13 14:01:18	Info	—	=====
14	2025-12-13 14:01:18	Info	—	🔗 URL: http://192.168.246.95:3000
15	2025-12-13 14:01:18	Info	—	📁 Repo: wprado/gter54
16	2025-12-13 14:01:18	Info	—	🌿 Branch: feature
17	2025-12-13 14:01:18	Info	—	📄 Arquivo: netbox-data-source/intended/sp1-srlinux-pe-101.cfg

GITEA - Intended Config - Branch feature

23 lines | 1.7 KiB | INI

[Raw](#)[Permalink](#)[Blame](#)[History](#)

```
1 set / network-instance default type default
2 set / interface lo0 admin-state enable
3 set / interface lo0 subinterface 0 ipv4 address 10.255.255.1/32
4 set / interface lo0 subinterface 0 ipv4 admin-state enable
5 set / network-instance default interface lo0.0
6 set / interface ethernet-1/1 admin-state enable
7 set / interface ethernet-1/1 subinterface 0 ipv4 address 10.0.0.1/30
8 set / interface ethernet-1/1 subinterface 0 ipv4 admin-state enable
9 set / network-instance default interface ethernet-1/1.0
10 set / interface ethernet-1/2 admin-state enable
11 set / interface ethernet-1/2 subinterface 0 ipv4 address 10.0.0.5/30
12 set / interface ethernet-1/2 subinterface 0 ipv4 admin-state enable
13 set / network-instance default interface ethernet-1/2.0
14 set / network-instance default protocols ospf instance main admin-state enable
15 set / network-instance default protocols ospf instance main version ospf-v2
16 set / network-instance default protocols ospf instance main router-id 10.255.255.1
17 set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/1.0 admin-state enable
18 set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/1.0 interface-type point-to-point
19 set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/2.0 admin-state enable
20 set / network-instance default protocols ospf instance main area 0.0.0.0 interface ethernet-1/2.0 interface-type point-to-point
21 set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 admin-state enable
22 set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 interface-type point-to-point
23 set / network-instance default protocols ospf instance main area 0.0.0.0 interface lo0.0 passive true
```

GITEA - Pull Request para Branch Development

± 1 changed files with 1 additions and 1 deletions



2 netbox-data-source/intended/sp1-srlinux-pe-101.cfg






		@@ -4,7 +4,7 @@ set / interface lo0 subinterface 0 ipv4 address 10.255.255.1/32
4	4	set / interface lo0 subinterface 0 ipv4 admin-state enable
5	5	set / network-instance default interface lo0.0
6	6	set / interface ethernet-1/1 admin-state enable
7	-	set / interface ethernet-1/1 subinterface 0 ipv4 address 10.0.0.13/30
7	+	set / interface ethernet-1/1 subinterface 0 ipv4 address 10.0.0.1/30
8	8	set / interface ethernet-1/1 subinterface 0 ipv4 admin-state enable
9	9	set / network-instance default interface ethernet-1/1.0
10	10	set / interface ethernet-1/2 admin-state enable



GITEA - Pull Request para Branch Development

Mudanca de Bloco P2P SR1 para SR2 #13

 Merged **wprado** merged 4 commits from `feature` into `development` 19 minutes ago

 Conversation **0**  Commits **4**  Files Changed **2**



wprado commented 19 minutes ago









Owner



No description provided.



wprado added 4 commits 19 minutes ago

-   delete video youtube
-   NetBox: sp1-srlinux-pe-101 - 2025-12-13 20:37:24
-   NetBox: sp1-srlinux-pe-101 - 2025-12-13 20:43:17
-   NetBox: sp2-srlinux-pe-102 - 2025-12-13 20:43:40

c53c903c52

5b9babcab0

327a4b4403

 f8733211d4








wprado merged commit **93132102e7** into `development` 19 minutes ago

Pipeline CI - validation.yml [JOBS]

test

Running



>  Set up job	-1s
>  Clean up old workspace	0s
>  Checkout repository	2s
>  Setup Python venv	1s
>  Clean up old labs	21s
<input type="radio"/> Deploy lab	0s
<input type="radio"/> Wait for lab to be ready	0s
<input type="radio"/> Run Interface tests	0s
<input type="radio"/> Run ICMP tests	0s
<input type="radio"/> Run OSPF tests	0s
<input type="radio"/> Destroy lab	0s
<input type="radio"/> Cleanup	0s
<input type="radio"/> Complete job	0s

Pipeline CI - validation.yml - Interface Tests [PASSED]

✓ Run Interface tests

10s

```
1 ===== test session starts =====
2 platform linux -- Python 3.14.0, pytest-7.4.4, pluggy-1.6.0 -- /tmp/gter54-dev-208/.venv/bin/python3
3 cachedir: .pytest_cache
4 rootdir: /tmp/gter54-dev-208/nuts
5 configfile: pytest.ini
6 plugins: nuts-3.5.1
7 collecting ...
8   collecting 45 items
9   collected 45 items
10
11 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR1_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 2%]
12 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR1_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 4%]
13 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR1_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 6%]
14 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR2_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 8%]
15 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR2_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 11%]
16 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR2_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 13%]
17 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR3_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 15%]
18 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR3_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 17%]
19 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_enabled[SR3_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 20%]
20 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR1_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 22%]
21 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR1_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 24%]
22 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR1_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 26%]
23 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR2_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 28%]
24 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR2_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 31%]
25 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR2_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 33%]
26 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR3_0] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 35%]
27 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR3_1] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 37%]
28 tests/interfaces/test_sr_interfaces.yaml::TestNapalmInterfaces::test_is_up[SR3_2] <- ../.venv/lib/python3.14/site-packages/nuts/base_tests/napalm_interfaces.py PASSED [ 40%]
```

Pipeline CI - validation.yml - ICMP Tests [PASSED]

✓ Run ICMP tests

1m34s

```
1 ===== test session starts =====
2 platform linux -- Python 3.14.0, pytest-7.4.4, pluggy-1.6.0 -- /tmp/gter54-dev-208/.venv/bin/python3
3 cachedir: .pytest_cache
4 rootdir: /tmp/gter54-dev-208/nuts
5 configfile: pytest.ini
6 plugins: nuts-3.5.1
7 collecting ...
8   collecting 3 items
9   collecting 12 items
10  collecting 24 items
11  collecting 36 items
12  collected 45 items
13
14 tests/icmp/test_sr1_loopback_local.yaml::TestSRLinuxPing::test_ping_success[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 2%]
15 tests/icmp/test_sr1_loopback_local.yaml::TestSRLinuxPing::test_ping_no_loss[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 4%]
16 tests/icmp/test_sr1_loopback_local.yaml::TestSRLinuxPing::test_ping_max_drop[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 6%]
17 tests/icmp/test_sr1_loopback_to_sr2.yaml::TestSRLinuxPing::test_ping_success[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 8%]
18 tests/icmp/test_sr1_loopback_to_sr2.yaml::TestSRLinuxPing::test_ping_no_loss[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 11%]
19 tests/icmp/test_sr1_loopback_to_sr2.yaml::TestSRLinuxPing::test_ping_max_drop[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 13%]
20 tests/icmp/test_sr1_loopback_to_sr3.yaml::TestSRLinuxPing::test_ping_success[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 15%]
21 tests/icmp/test_sr1_loopback_to_sr3.yaml::TestSRLinuxPing::test_ping_no_loss[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 17%]
22 tests/icmp/test_sr1_loopback_to_sr3.yaml::TestSRLinuxPing::test_ping_max_drop[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 20%]
23 tests/icmp/test_sr1_to_sr2_p2p.yaml::TestSRLinuxPing::test_ping_success[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 22%]
24 tests/icmp/test_sr1_to_sr2_p2p.yaml::TestSRLinuxPing::test_ping_no_loss[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 24%]
25 tests/icmp/test_sr1_to_sr2_p2p.yaml::TestSRLinuxPing::test_ping_max_drop[SR1_] <- custom_tests/sr_linux_ping_napalm.py PASSED [ 26%]
```

Pipeline CI - validation.yml - OSPF Tests [FAILED]

test


Failure

Run OSPF tests

```
1 ===== test session starts =====
2 platform linux -- Python 3.14.0, pytest-7.4.4, pluggy-1.6.0 -- /tmp/gter54-dev-182/.venv/bin/python3
3 cachedir: .pytest_cache
4 rootdir: /tmp/gter54-dev-182/nuts
5 configfile: pytest.ini
6 plugins: nuts-3.5.1
7 collecting ...
8   collecting 6 items
9   collected 18 items
10
11 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 5%]
12 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 11%]
13 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 16%]
14 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 22%]
15 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 27%]
16 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 33%]
17 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 38%]
18 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 44%]
19 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 50%]
20 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 55%]
21 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 61%]
22 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 66%]
23 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 72%]
24 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 77%]
25 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 83%]
26 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 88%]
27 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py FAILED [ 94%]
28 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py FAILED [100%]
29
30 ===== FAILURES =====
31
32 TestSRLinuxOSPFNeighbor.test_neighbor_exists[SR1_0]
33 custom_tests/sr_linux_ospf_napalm.py:82: in test_neighbor_exists
34     assert neighbor_id in single_result.result, \
35 E   AssertionError: Neighbor 2.2.2.2 não encontrado no host SR1. Neighbors disponiveis: ['10.255.255.2', '10.255.255.3']
```

NUTS - test_data - Error

development development gter54 / nuts / tests / ospf / test_sr1_ospf_neighbors.yaml

 wprado 3a082d6f56 NUTS

15 lines | 394 B | YAML | Executable File

```
1 ---
2 - test_module: custom_tests.sr_linux_ospf_napalm
3   test_class: TestSRLinuxOSPFNeighbor
4   test_data:
5     - host: SR1
6       instance: main
7       neighbor_id: 2.2.2.2
8       expected_interface: ethernet-1/1.0
9       description: "SR1 - Neighbor SR2"
10    - host: SR1
11      instance: main
12      neighbor_id: 3.3.3.3
13      expected_interface: ethernet-1/2.0
14      description: "SR1 - Neighbor SR3"
```

NUTS - test_data - Error

development gter54 / nuts / tests / ospf / test_sr1_ospf_neighbors.yaml

wprado 3a082d6f56 NUTS

15 lines | 394 B | YAML | Executable File

```
1 ---
2 - test_module: custom_tests.sr_linux_ospf_napalm
3   test_class: TestSRLinuxOSPFNeighbor
4   test_data:
5     - host: SR1
6       instance: main
7       neighbor_id: 2.2.2.2
8       expected_interface: ethernet-1/1.0
9       description: "SR1 - Neighbor SR2"
10    - host: SR1
11      instance: main
12      neighbor_id: 3.3.3.3
13      expected_interface: ethernet-1/2.0
14      description: "SR1 - Neighbor SR3"
```

←

←

ESPERADO:
- 10.255.255.2
- 10.255.255.3

NUTS - Atualizando o teste_data

development ▾



gter54 / nuts / tests / ospf / test_sr1_ospf_neighbors.yaml



wprado

655d1a6f27



Update nuts/tests/ospf/test_sr1_ospf_neighbors.yaml

15 lines | 404 B | YAML | Executable File

```
1  ---
2  - test_module: custom_tests.sr_linux_ospf_napalm
3    test_class: TestSRLinuxOSPFNeighbor
4    test_data:
5      - host: SR1
6        instance: main
7        neighbor_id: 10.255.255.2
8        expected_interface: ethernet-1/1.0
9        description: "SR1 - Neighbor SR2"
10     - host: SR1
11       instance: main
12       neighbor_id: 10.255.255.3
13       expected_interface: ethernet-1/2.0
14       description: "SR1 - Neighbor SR3"
```

Pipeline CI - validation.yml - OSPF Tests [PASSED]

✓ Run OSPF tests

```
1 ===== test session starts =====
2 platform linux -- Python 3.14.0, pytest-7.4.4, pluggy-1.6.0 -- /tmp/gter54-dev-185/.venv/bin/python3
3 cachedir: .pytest_cache
4 rootdir: /tmp/gter54-dev-185/nuts
5 configfile: pytest.ini
6 plugins: nuts-3.5.1
7 collecting ...
8   collecting 6 items
9   collected 18 items
10
11 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 5%]
12 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 11%]
13 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 16%]
14 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 22%]
15 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR1_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 27%]
16 tests/ospf/test_sr1_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR1_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 33%]
17 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 38%]
18 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 44%]
19 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 50%]
20 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 55%]
21 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR2_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 61%]
22 tests/ospf/test_sr2_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR2_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 66%]
23 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 72%]
24 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_exists[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 77%]
25 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 83%]
26 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_state_full[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 88%]
27 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR3_0] <- custom_tests/sr_linux_ospf_napalm.py PASSED [ 94%]
28 tests/ospf/test_sr3_ospf_neighbors.yaml::TestSRLinuxOSPFNeighbor::test_neighbor_interface[SR3_1] <- custom_tests/sr_linux_ospf_napalm.py PASSED [100%]
29
30 ===== 18 passed in 17.81s =====
```

Pipeline CI - validation.yml - [ALL JOBS PASSED]

test

Success





> ✓ Set up job	-1s
> ✓ Clean up old workspace	0s
> ✓ Checkout repository	2s
> ✓ Setup Python venv	0s
> ✓ Clean up old labs	21s
> ✓ Deploy lab	1m3s
> ✓ Wait for lab to be ready	30s
> ✓ Run Interface tests	8s
> ✓ Run ICMP tests	1m40s
> ✓ Run OSPF tests	19s
> ✓ Destroy lab	3s
> ✓ Cleanup	0s
> ✓ Complete job	0s

GITEA - Pull Request - Branch MAIN

New Pull Request

Select the branch to merge into and the branch to pull from.




 merge into: wprado:main ▾

 pull from: wprado:development ▾

New Pull Request

16 Commits

main ... developmen

Author	SHA1	Message	Date
 wprado	431c1672f0	Update nuts/tests/ospf/test_sr3_ospf_neighbors.yaml ✓	1 hour ago  

All checks were successful

✓ Network Validation / test (push) Successful in 4m5s [Details](#)

GITEA - Pull Request - Branch MAIN

All checks were successful

[Hide all checks](#)

✓ Network Validation / test (push) Successful in 4m5s

[Details](#)

✓ This pull request can be merged automatically.

🔒 There is no key available to sign this commit.

⚠ This branch is out-of-date with the base branch

Update branch by merge



🔗 Create merge commit



▶ [View command line instructions](#)

Pipeline CD - deploy.yml

✓ Production Deploy

7s

🚀 PRODUCTION DEPLOYMENT 🚀

📦 Project: GTER54 Network Automation Lab

🔧 Deployment Method: Ansible/Python

🎯 Deploying configurations...

📡 [Ansible] Connecting to devices...

✓ sp1-srlinux-pe-101 - Configuration applied

✓ sp2-srlinux-pe-102 - Configuration applied

✓ sp3-srlinux-pe-103 - Configuration applied

🔍 Verifying OSPF neighbors...

✓ All OSPF neighbors UP

✓ DEPLOY SUCCESSFUL! ✓

🎉 Production environment updated!

🌟 Ready for GTER54 demonstration!

Pipeline CD - deploy.yml

deploy

Success



- | | |
|-------------------------|----|
| > ✓ Set up job | 0s |
| > ✓ Checkout repository | 1s |
| > ✓ Production Deploy | 7s |
| > ✓ Cleanup | 0s |
| > ✓ Complete job | 0s |

Resumo



Rastreabilidade

Cada mudança rastreada do Git ao router com histórico completo.



Segurança

Validação prévia em Digital Twin antes de aplicar em produção.



Velocidade

Deploy automatizado reduz tempo e elimina erros manuais.



Auditoria

Registro completo de quem mudou, quando e por quê.



Consistência


Configurações padronizadas geradas por templates.



Confiabilidade

Testes automatizados garantem qualidade antes do deploy.


GITHUB - <https://github.com/wsdoprado/gter54>

 **gter54** Public

Pin Watch 0 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

 **wsdoprado** Update project title in README.md 66b66f7 · 15 hours ago 10 Commits

containerlab	Project	15 hours ago
netbox-data-source	Fix template syntax and remove unnecessary lines	15 hours ago
nuts	Project	15 hours ago
workflows	Project	15 hours ago
.gitignore	Initial commit	last week
LICENSE	Project	15 hours ago
README.md	Update project title in README.md	15 hours ago
pyproject.toml	Project	15 hours ago

README Apache-2.0 license

GTER54 - Do Git ao Router

Projeto de automação de rede demonstrando um pipeline completo desde o Git até o deployment em roteadores, apresentado no GTER54.

About

GTER54 - do GIT ao Router

- Readme
- Apache-2.0 license
- Activity
- 0 stars
- 0 watching
- 0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages

Python 87.6% Jinja 12.4%

Suggested workflows

Based on your tech stack

Obrigado
www.ix.br

Dezembro, 16, 2025

nic.br egi.br
www.nic.br | www.cgi.br