ROEDUNET 2026 · IAȘI

Transform-based configuration management on a 400-switch multi-vendor campus

Andrei-Alexandru Bleorțu, Mihai Bărbulescu, Costin Grigoraș, Mihai Carabaș

National University of Science and Technology POLITEHNICA Bucharest · ARNIEC / RoEduNet · CERN

407 switches, configured one command at a time



316 IOS, IOS-XE **7** C2950 (IOS 12.1) **18** NX-OS **35** Ruckus FastIron +31 unread or excluded

the old way (illustrative)

```
sw(config)# interface Gi1/0/1
sw(config-if)# spanning-tree portfast
sw(config-if)# interface Gi1/0/2
sw(config-if)# spanning-tree portfast
… 559 lines per switch
```

23
IOS hardware families

19
software versions

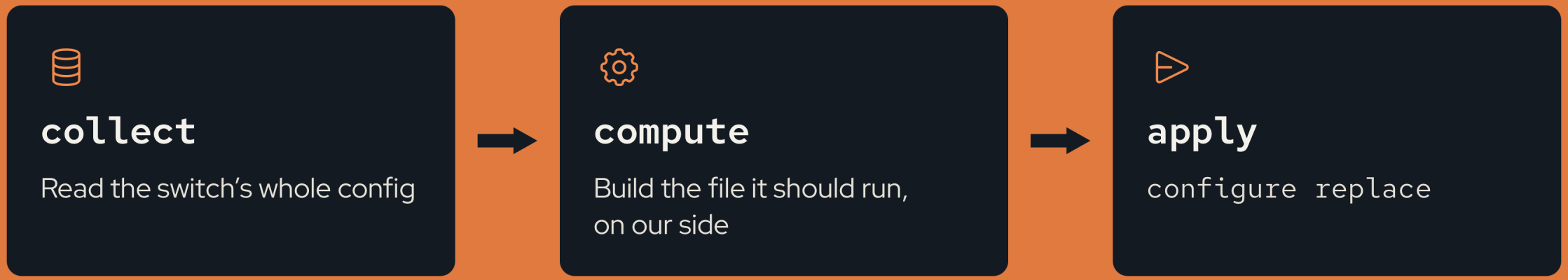
THE PROBLEM

Seven problems with sending commands

10 min	per switch, half-configured until the end
559 lines	shown as “changes” on a switch that already matched
13 + 54	rescue blocks and ignored errors: syntax tried live
port flaps	from spanning-tree lines resent on every run
2 attempts	at the new TACACS+ syntax, both abandoned
7 families	got a NetFlow config never tested on them · our mistake
no rollback	after a failed task cancelled the reboot · our mistake

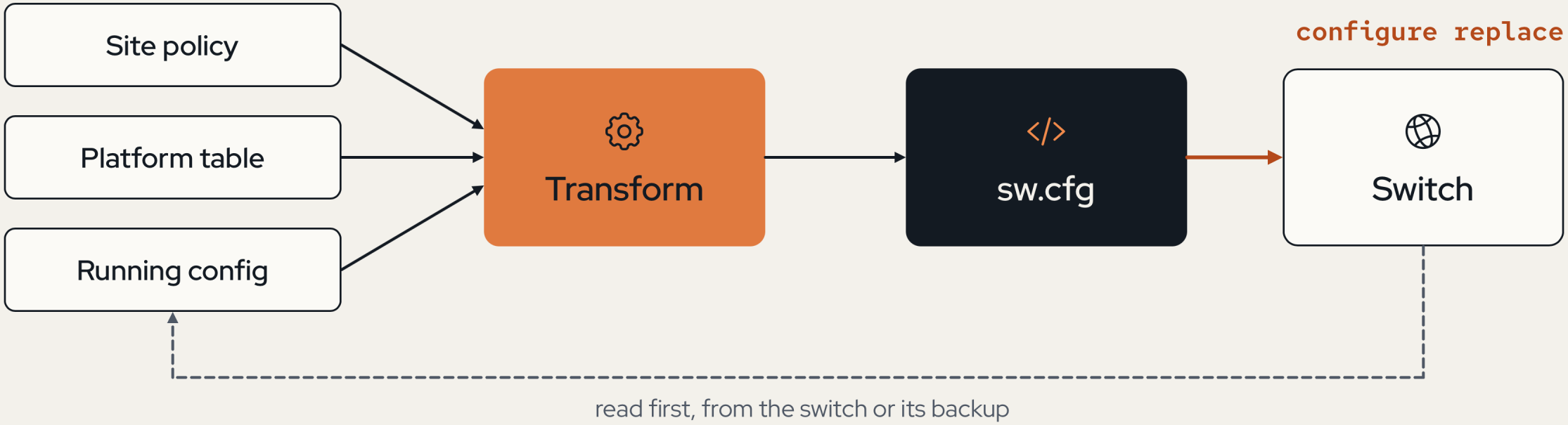
THE OPPOSITE APPROACH

Collect, compute, apply.



THE TRANSFORM

One complete file per switch



We rewrite only the lines the policy manages. Everything else is copied unchanged, in place.

THE TRANSFORM

Every line is managed or carried

```
interface GigabitEthernet1/0/5
 description EC105 printer                    carried
 spanning-tree portfast edge                  managed
no cdp run                                    managed: removed
access-list 1 permit 192.0.2.10               managed
access-list 10 permit 192.0.2.0 0.0.0.255     carried
```

managed — the policy writes it

carried — the switch's own, copied unchanged

THE TRANSFORM

Same setting, different spelling

Same C3750X, two software versions

```
15.0(2)     spanning-tree portfast
15.2(4)     spanning-tree portfast edge
```

Domain name

```
older IOS    ip domain-name
IOS-XE 17    ip domain name
```

```
switchport trunk encapsulation dot1q
```



accepted by **6 of 23** families

A platform table picks the spelling in advance. Nothing is tried on the live switch.

THE TRANSFORM · WHERE THE TABLE COMES FROM

Three ways we learned what each switch accepts

1 · ask its parser

```
spanning-tree portfast ?
  edge
  <cr>
Ctrl-U line erased
```

70 commands on one switch of each family. Nothing is entered.

2 · read the backups

stores	plain	edge
C1000	0	102
C9200L	37	0
C2960S	22	0

Count which spelling every switch of a family already stores.

3 · learn from failures

```
tacacs server X
 ^
% Invalid input detected
at '^' marker
```

A failed replace names the line. The C2960 keeps the old TACACS+ syntax.

When the answers disagree, **what the switch stores wins**.

THE TRANSFORM

Three times it refuses to produce a file

ExcludedSwitch

```
excluded:
  - core-sw-01
```

On the never-touch list

IncompleteConfig

```
interface Gi1/0/47
 switchport mode acc
```

The capture is cut off

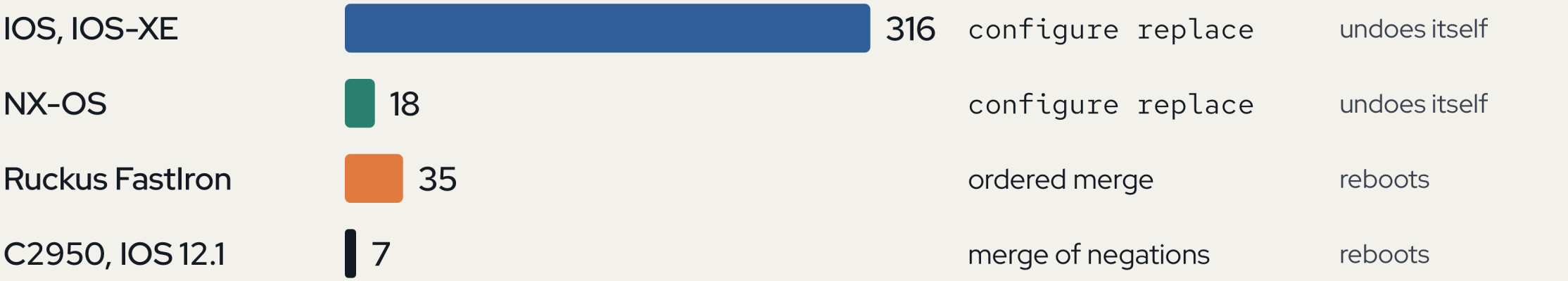
LockoutRisk

```
~~tacacs server …~~
~~ access-class 1 in~~
```

The file would cut off our own access

DEPLOYMENT

How each platform gets its file



Lose the switch mid-change, and it recovers by itself on every platform.

Five commands, two safety timers

```
sw# write memory
sw# reload in 12
sw# configure replace http://…/sw.cfg list time 7
Rollback Done
sw# configure confirm
sw# reload cancel    ! only after a new login works
```



0 min · start

7 min · the switch undoes the change

12 min · it reboots

DEPLOYMENT · LOST LOGINS

A switch that forwards but won’t let you in

```
sw# configure replace … list time 7
Rollback aborted after 5 passes
sw# configure revert now
% Authorization failed
```

✕ Our open session still works. It proves nothing.

✓ A **new** login succeeds: only then cancel the reboot.

THE OPERATOR'S VIEW

You approve a diff, not a list of commands

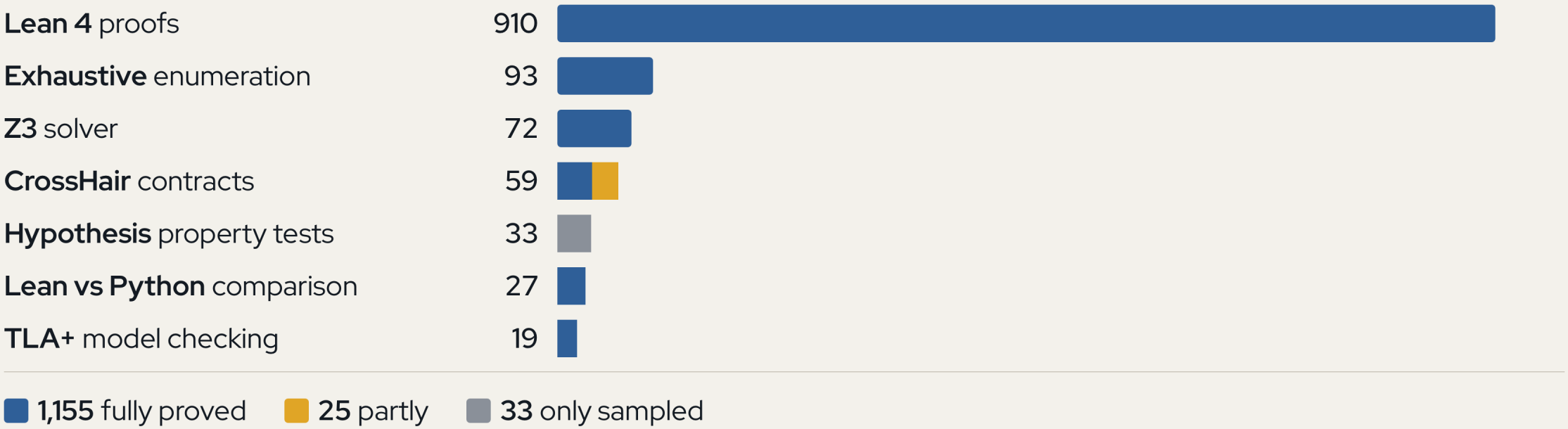
backup → live check → validated → deployed

on the switch now	what it should run
`interface Gi1/0/5`	`interface Gi1/0/5`
` description EC105`	` description EC105`
` spanning-tree portfast`	` spanning-tree portfast` **edge**
` switchport mode access`	` switchport mode access`

The highlighted line is all that **configure replace** will change.

ASSURANCE

1,213 claims checked by seven tools



ASSURANCE

The first full run found 18 real bugs



20 claims failed. **18** were bugs, fixed with a test each. 2 were a wrong rule and dead code.

THE WORST ONE · C2950 MERGE

```
tacacs-server host A key NEW
no tacacs-server host A
```

Added with the new key, then removed. The switch would keep only its local account.

RESULTS · 28 AUGUST 2026

One rollout across the campus

313
switches

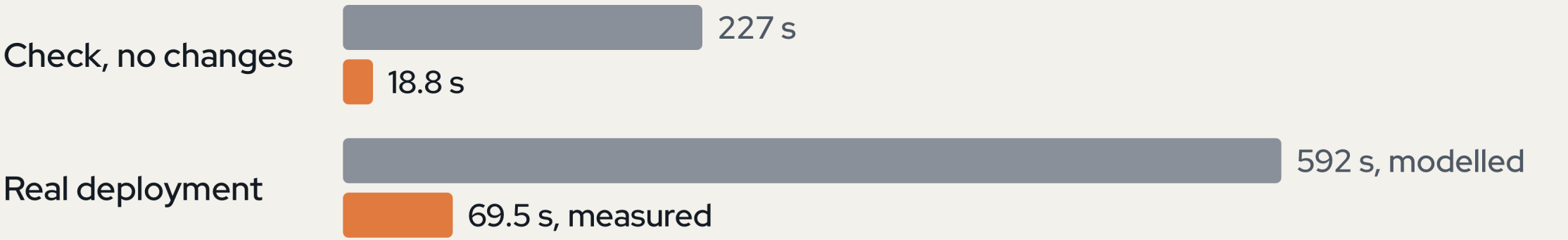
46
minutes

0
failures

5–6 seconds per switch in a batch. **1,542** deployments in the first five weeks.

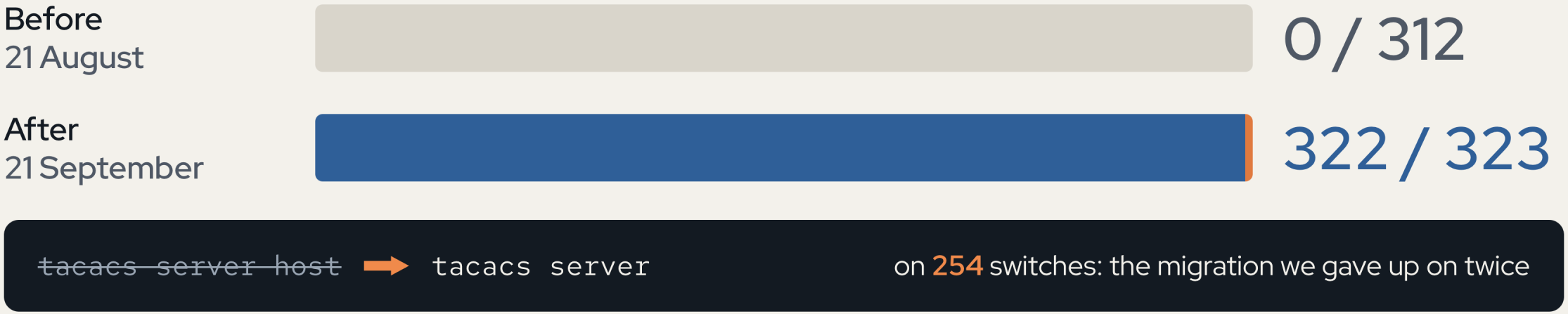
RESULTS · SAME TEN SWITCHES

Old way against new way



RESULTS

Switches that match the policy



DISCUSSION

Can you use this on your network?

- ✓ **Yes:** Cisco IOS 12.2(55)+, NX-OS
- ✓ **Should work:** Junos commit confirmed
- ~ **Extra work:** FastIron, very old IOS

Honest limits

One campus only

We check syntax, not meaning

The old way's time is modelled

TAKEAWAYS

Compute the whole file, let the switch apply it

1. Record each platform's syntax in advance; never try it live.
2. Rewrite only what the policy manages; carry everything else.
3. Recover on the switch: replace timer, reload, and a new login.
4. 313 switches in 46 minutes, without a failure.

Thank you. Questions?

andrei.bleortu@stud.acs.upb.ro

How this differs from existing tools

Tool	What it does · how we differ
NAPALM	Replaces with a candidate the user must produce · we generate it from the running config
Managed / unmanaged (ipSpace)	Regex split, managed part appended at the end · we place lines, prove unmanaged ones survive, check lockout
hier_config, Golden Config	Hand-written negation rules per OS · the device computes the difference
Robotron, NSO	Their model is the source of truth · ours is the running config
Batfish, Campion	Analyse meaning with their own grammars · we use the switch's parser: syntax, not meaning
Cedar	Lean proofs plus differential tests · same approach, plus TLA+, Z3 and property tests

What the Lean and TLA+ models cover

Lean 4 · 12 modules, 871 theorems

- The parser neither drops nor invents a line
- Prefixes match only on whole words
- Carried lines keep their order, never lost or doubled
- The lockout check fires exactly when a hazard exists
- The file goes only to its switch, during its deployment

TLA+ · deployment handlers

- 15 invariants, 4 liveness properties
- At most one deployment per switch; a stalled run ends
- Smallest domain: two sessions, two switches, two files, two jobs
- Not covered: two switches deploying at once

How the old real run was modelled

Check mode, plus write costs measured on an **unused port** of each switch, then restored.

0.7 s extra per call

0.35 s per line sent

Fitted on one set of calls, tested on another: off by a median **3.7%** (worst 13.1%), too low on 8 of 10 switches.

Biases

24-port switches favour the old playbook: its calls grow with ports.

The model leaves out **write memory** and the reload.

78% of the old check time went to the configuration modules.

THE TRANSFORM

Write it the way the switch stores it

Same bytes

show prints

^C



we write

0x03

A banner delimiter is one byte, not two characters.

Same place

```
…
logging host 192.0.2.10
ntp server 192.0.2.5
line con 0
…
```

A new line goes where the other switches keep it, not at the top.

Same idea, different commands

NX-OS

```
reload in 1200
configure replace …
 commit-timeout 600
```

Both switches of a pair together

Ruckus FastIron

```
reload after …
no <old line>
<new line>
```

No replace: an ordered merge

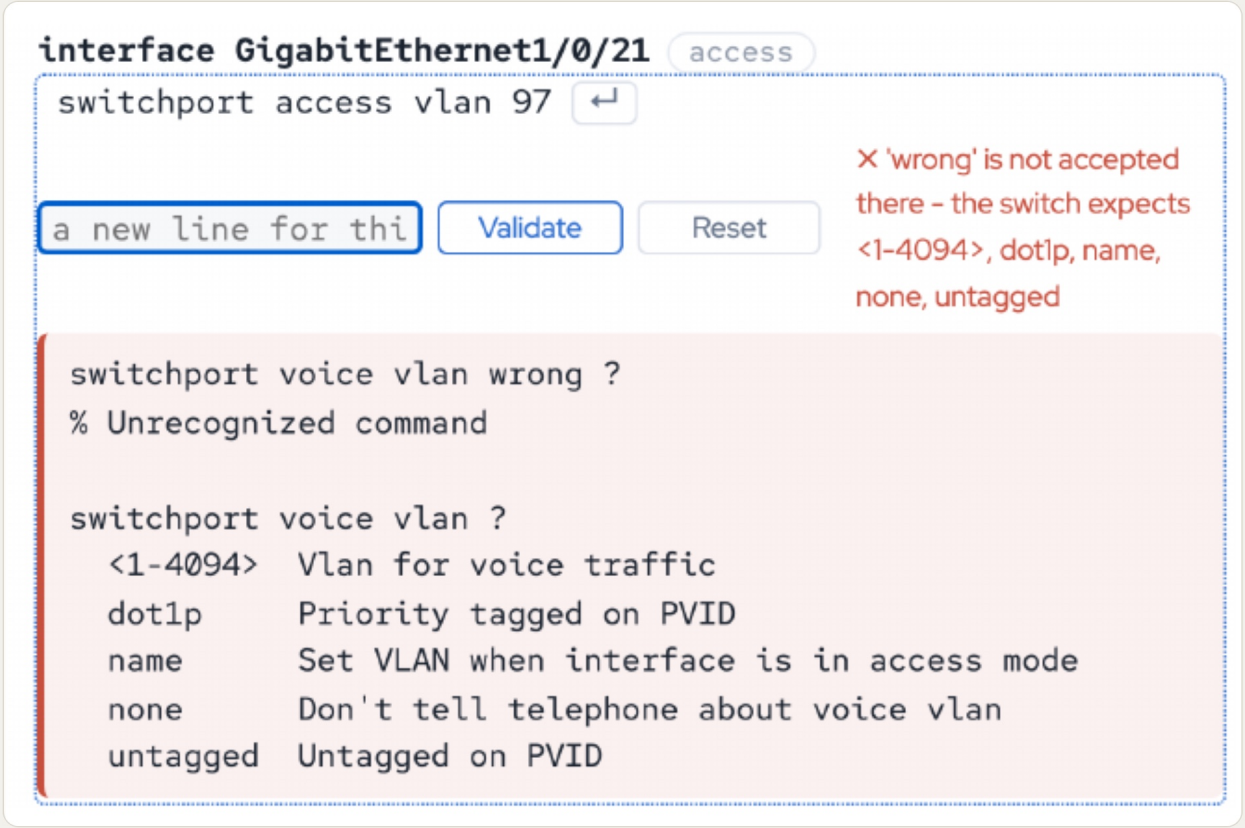
Catalyst 2950

```
reload in …
no <old line>
<new line>
```

From 2001: negations over SSH

THE OPERATOR'S VIEW

The switch itself checks your edit



? Type the line, ask the switch

Ctrl-U Erase it: never entered

<cr> Offered? The line is valid

RESULTS

The price: knowing every platform’s quirks

23
hardware families

19
software versions

70
commands probed on each family

```
one quirk, learned from a failed replace on a C3560
username admin secret 5 $1$<8-char salt>$…
ERROR: The secret you entered is not a valid encrypted secret
```

This family wants a 4-character salt, not 8.

DISCUSSION

Two things that still surprised us

Proofs don’t know the switch

```
cli alias name write …
verify-only: accepted
configure replace: refused
```

NX-OS’s own check missed a reserved word.

Paired switches still blip

6–26 s

of vPC downtime on three pairs, from one MTU change.