Northern.tech

No Reboot Required: Over-the-Air Updates for Immutable Filesystems

Jana Perić, Customer Engineer for Mender.io
Embedded Linux Conference Europe 2026

Why shift towards read-only rootfs

SECURITY

Integrity

Tampering is detected and blocked: `dm-verity`, secure boot, signed images

REGULATION

Compliance

EU CRA: integrity protection and security updates

FLEET RELIABILITY

No drift

Every device stays exactly as shipped

A one-line fix shouldn't cost 500 MB and a reboot

THE FIX

```
/usr/bin/sensor-check.sh
   INTERVAL=10
-  TIMEOUT=5
+  TIMEOUT=30
   RETRIES=3
```

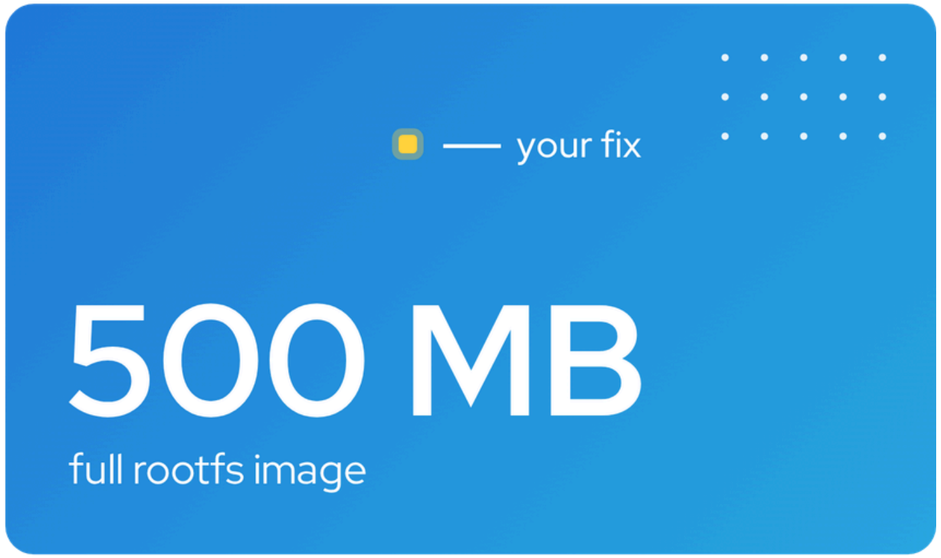
1 line

a few bytes of change

what we ship today



THE UPDATE



Reboot

~60 s downtime

× **every device** in the fleet

Northern.tech

Can we have both?

WE WANT

A sealed OS

Read-only, cryptographically verified, identical on every device

+

WE ALSO NEED

Fast updates

Small, frequent changes – applied live, no reboot

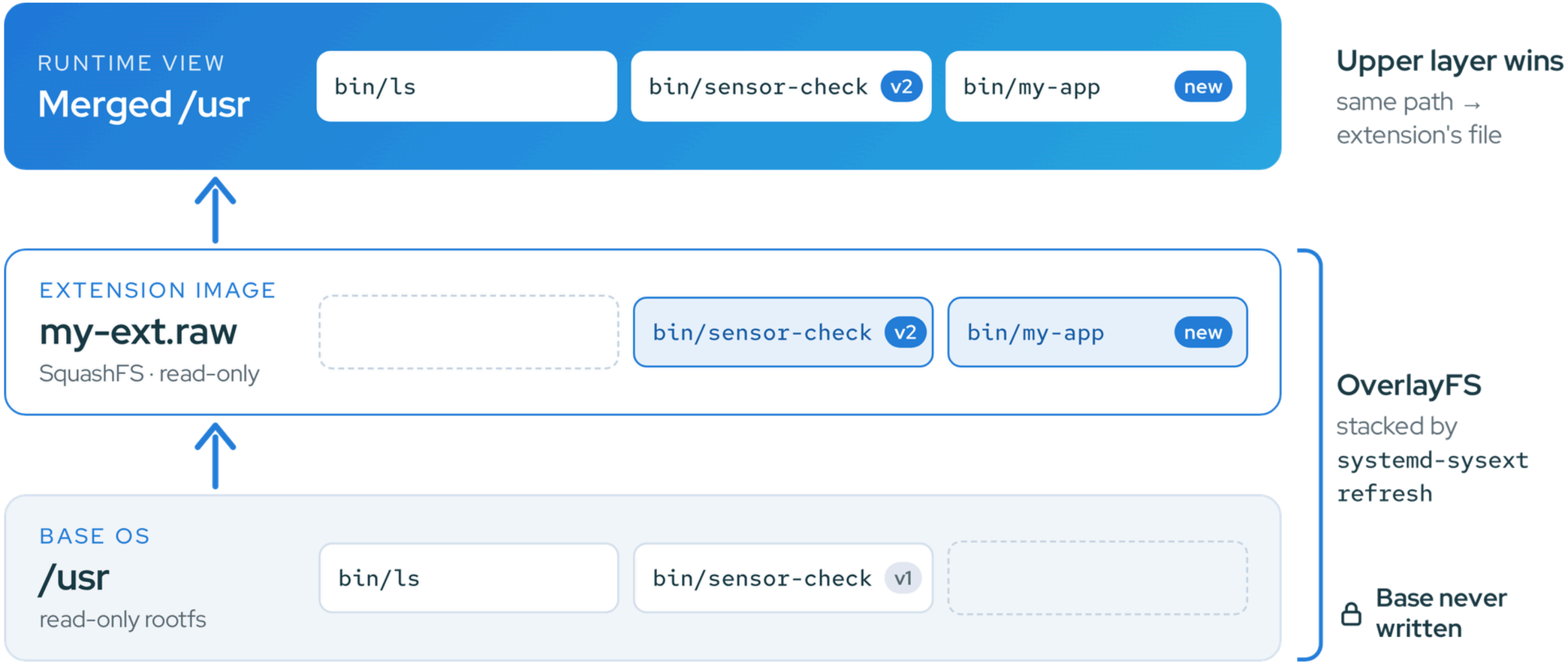
Can we change a few files in /usr
without writing the rootfs – and without rebooting?

Northern.tech

Three ways to ship an update

	Full rootfs A/B partitions	**Containers** Docker / Podman	THIS TALK **Extensions** systemd-sysext
REBOOT	Yes	No	No
TYPICAL PAYLOAD	100s of MB	10s–100s of MB	KB – a few MB
RUNTIME OVERHEAD	None	Runtime + writable storage	None
SCOPE	Everything, incl. kernel	Isolated apps	Files in /usr, /opt

sysext: merge an image onto /usr



Northern.tech

An extension is just a directory in an image

BUILD INPUT · A PLAIN DIRECTORY



mksquashfs

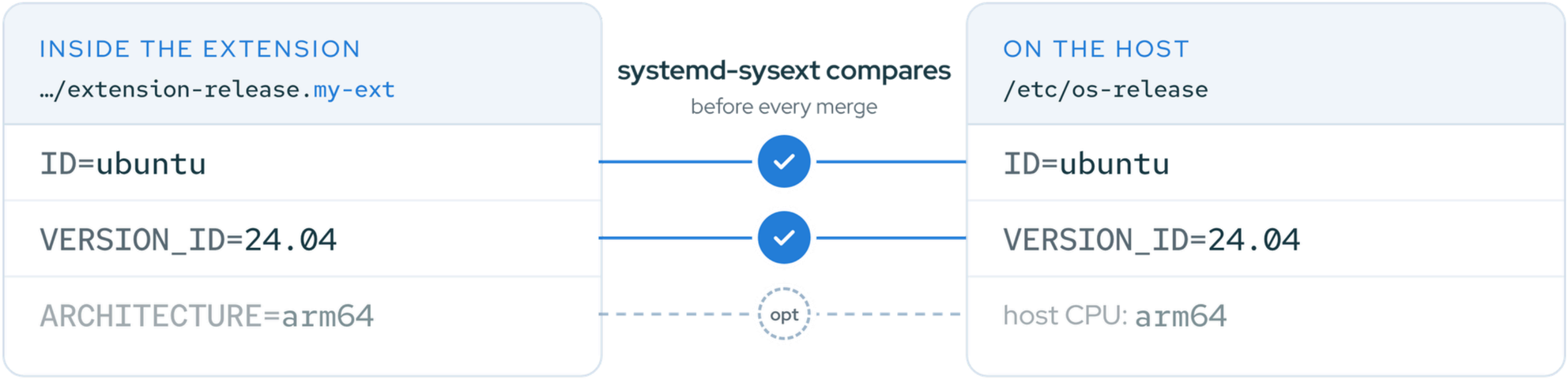
OUTPUT · ONE IMAGE FILE



Rule: paths mirror the host, and `extension-release.my-ext` must match `my-ext.raw`

Northern.tech

extension-release blocks mismatched images



Match → merged
Files appear in `/usr` immediately

Mismatch → refused
e.g. host `26.04` ≠ image `24.04`
— no broken libraries, no crash

Northern.tech

Images live on /var, so verify them

THE EXTENSION IMAGE

⚠ lives on writable `/var/lib/extensions`

Data
the SquashFS / EROFS filesystem

Verity hash tree
hashes every block → a **root hash**

Signature
PKCS#7 over the root hash

NOT IN TODAY'S DEMO

Demo image is plain SquashFS. This is the production path.

Kernel · dm-verity

- signature vs. trusted keyring
- every block re-hashed on read

Valid → merged
Signed by a key the device trusts

Tampered → rejected
Bad signature: never mounted
Bad block later: read fails

sysext solves activation - not delivery

ON ONE DEVICE

sysext does

- **Verify**
  release match, dm-verity signature
- **Merge**
  overlay onto `/usr`, `/opt`
- **Refresh**
  swap images live, no reboot

...on the one device you're logged into.

ACROSS A FLEET

sysext doesn't

- **Deliver**
  get the signed image to every device
- **Check**
  does the new version work as expected after the swap?
- **Roll back**
  restore the previous image on failure
- **Report**
  which devices run which version

Mender, and where our code plugs in

OPEN SOURCE OTA UPDATE MANAGER

Mender

Server + device client. It fills the gaps from the last slide:

- **Deliver**
  signed artifacts, deployments to devices or groups
- **Install, check & roll back**
  same steps every update, rollback built in
- **Report & version control**
  success or failure, and which version runs where

ON THE DEVICE · THE IDEA

THE GENERIC PART · SAME FOR EVERY UPDATE

Mender Client

download · state machine · rollback · reporting

+

THE SPECIFIC PART · ONE SMALL SCRIPT

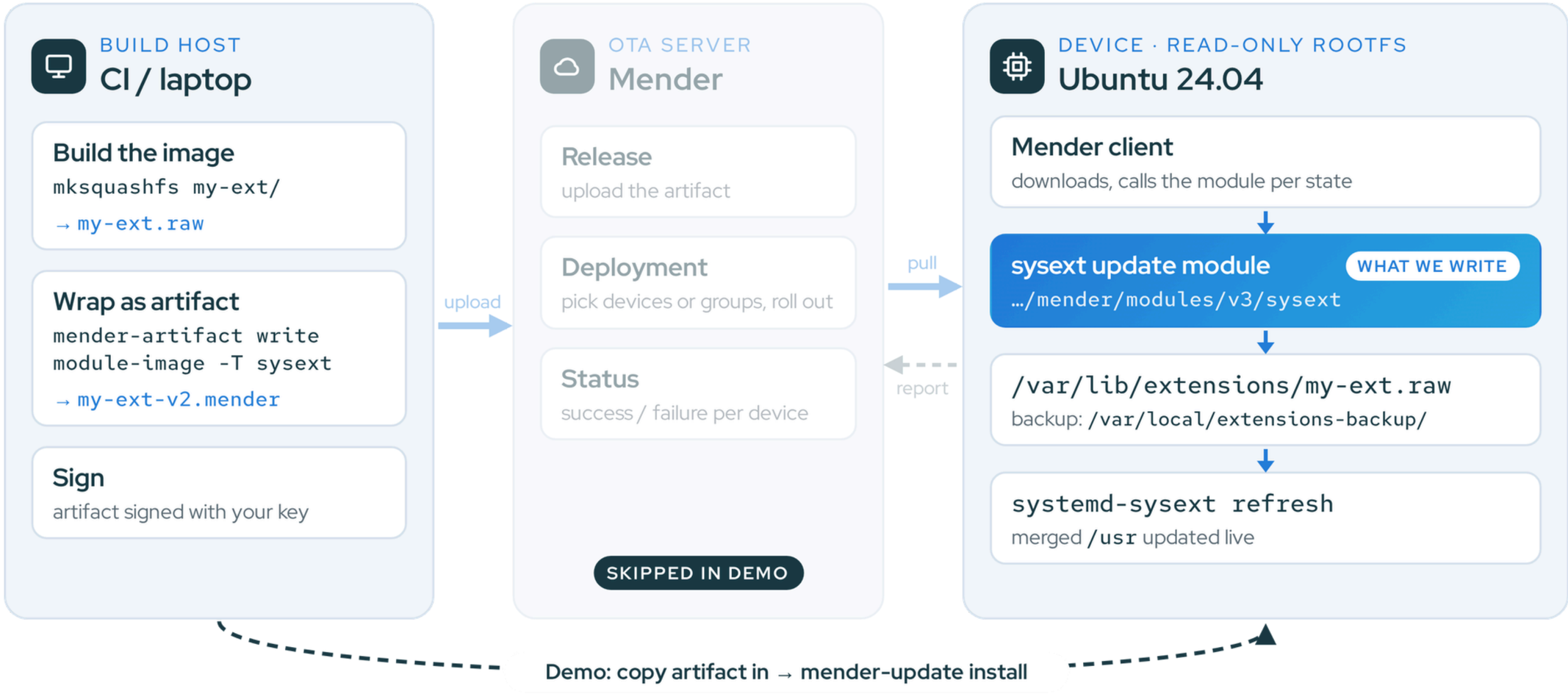
Update Module

knows how to install **your** payload
ours: `sysext` → place the image, `refresh`

You write the **what**. Mender handles the **when** and the **what-if**.

Northern.tech

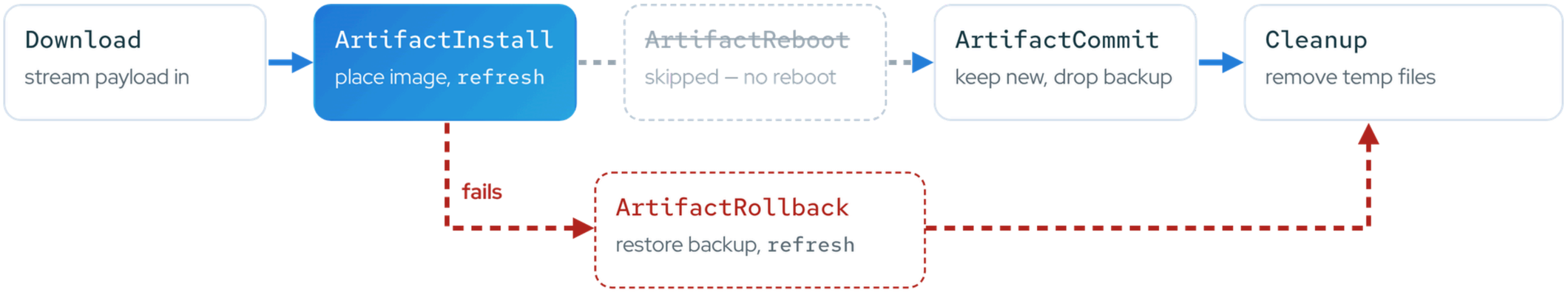
The whole pipeline



Northern.tech

Update Modules: a script per state

Mender runs `sysext <STATE> <files>` at every step



MENDER ASKS

NeedsArtifactReboot → No

the update is applied live

MENDER ASKS

SupportsRollback → Yes

keeps the previous version to fall back to

Northern.tech

Demo: update and roll back, no reboot

Mac → QEMU → Ubuntu 24.04, read-only `/usr`

Mender standalone mode: artifacts installed locally with `mender-update`. Same update module as OTA — just no server, no network, no dm-verity.

1 Truly read-only

```
$ touch /usr/bin/x
```

WATCH FOR

Read-only file system

2 Baseline

```
$ uptime
$ systemd-sysext status
$ sensor-check
```

WATCH FOR

`my-ext` merged · v1

3 Install v2

```
$ mender-update install
    my-ext-v2.mender
$ sensor-check
```

WATCH FOR

v2 live · no reboot

4 Roll it back

KEY MOMENT

```
$ mender-update rollback
$ sensor-check
```

WATCH FOR

back to v1 in seconds

5 Commit & prove it

```
$ mender-update install
    my-ext-v2.mender
$ mender-update commit
$ uptime
```

WATCH FOR

v2 kept · same boot

One boot, start to finish — zero reboots

Northern.tech

Gotcha #1 — overlayroot changes don't last

v2 installed & committed
on an overlayroot system, new version running

→ REBOOT →

v2 is gone
no error, no warning

WHY

The disk is read-only — the system isn't

RAM layer · writable
every write lands here, incl. extensions and Mender state

+

Real disk · read-only
what you started with

reboot → RAM wiped → back to what's on disk

DO THIS INSTEAD

- **Truly read-only rootfs**
  read-only mount or dm-verity
- **Persistent `/var`**
  extensions and Mender state on a real partition
- **Test after a reboot**
  not just after the update

overlayroot doesn't break sysext — it quietly throws your update away on reboot.

Northern.tech

Gotcha #2 — sysext stops at /usr and /opt

INSIDE MY-EXT.RAW

sysext only merges two places

- ✓ `usr/bin/sensor-check` — merged
- ✓ `opt/…` — merged
- ✗ `etc/sensor.conf` — silently ignored

FOR /ETC, USE ITS SIBLING

`systemd-confext`

Same idea, same image formats, same refresh — for configuration

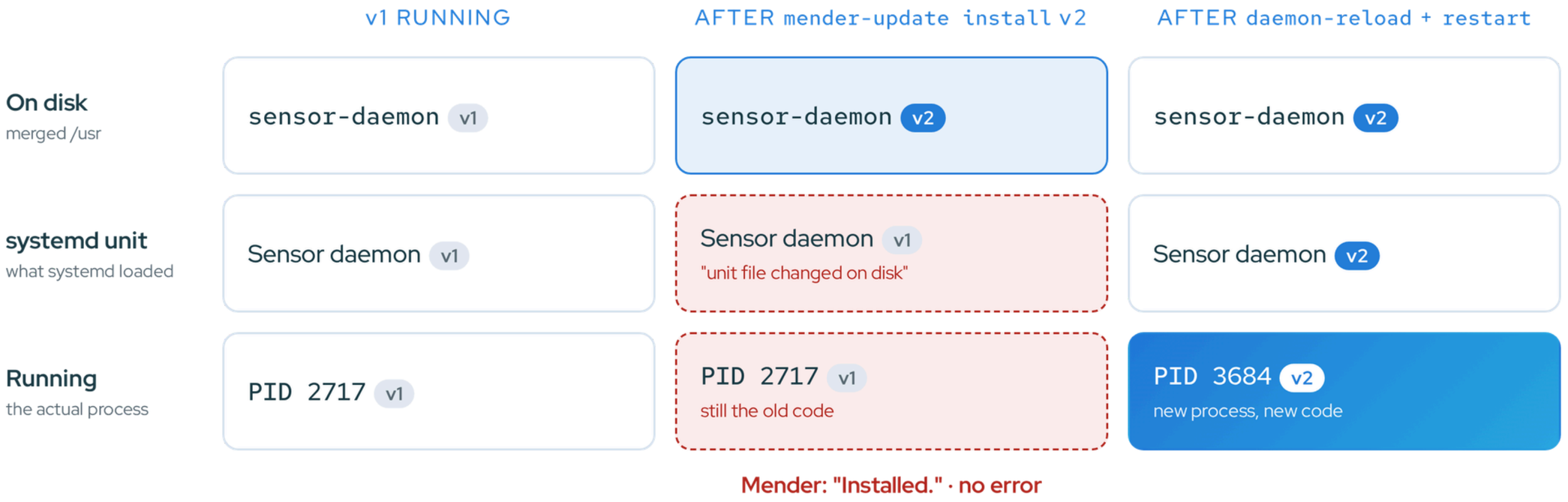
Merges onto	`/etc`
Images live in	`/var/lib/confexts/`
Metadata	`etc/extension-release.d/`
Apply live	`systemd-confext refresh`

⚠ **While merged, `/etc` is read-only** — runtime config writes will fail

Code → `sysext` · **Config** → `confext`

Northern.tech

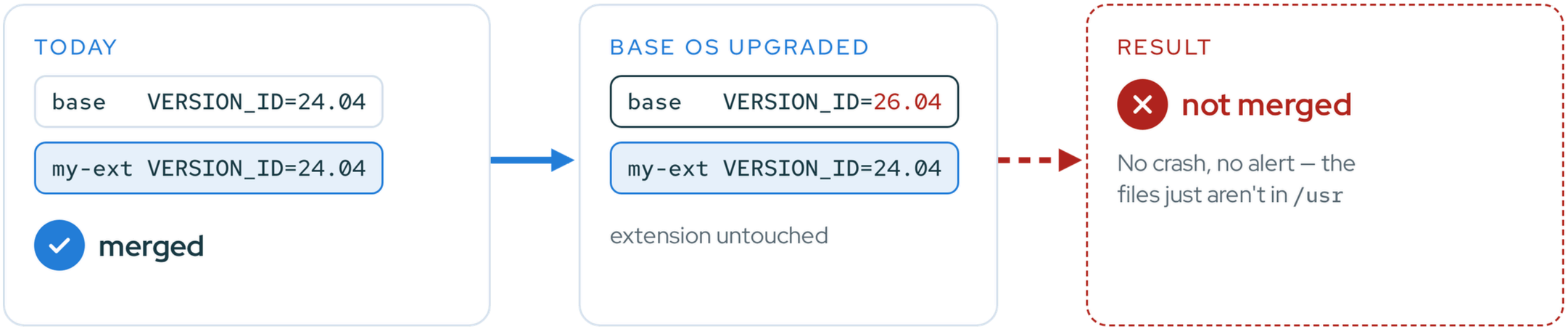
Gotcha #3 – Refresh isn't restart



After refresh: `systemctl daemon-reload` then `systemctl restart <svc>`

Northern.tech

Gotcha #4 – Base OS updates break extensions



PICK A POLICY BEFORE YOU SHIP

Rebuild & ship with the base

build each extension for every base release, deliver it in the same update

Match on SYSEXT_LEVEL

a compatibility number you bump only when needed

ID=_any

⚠ UNSAFE

disables the compatibility check. Scripts only, never binaries

Northern.tech

From proof of concept to production

DONE IN THIS POC

- **Live install, no reboot**
  sysext update module for Mender
- **Backup & rollback**
  previous version kept until commit
- **Merge check**
  catches images sysext silently skips
- **Tested the pitfalls**
  overlayroot and service restarts, reproduced on the VM

NEXT STEPS

- [ ] **dm-verity signing**
  build signed images with `systemd-repart`
- [ ] **Automatic health check**
  the hook is already in the module
- [ ] **Restart services after refresh**
  `daemon-reload` + `restart` in the module
- [ ] **Config via confext**
  a sibling module for `/etc`
- [ ] **Multiple extensions**
  one module, many images

The module is open – feedback, contributions and ports to other OTA frameworks welcome

Northern.tech

What this doesn't replace

IF YOU'RE CHANGING…	…USE
Kernel, bootloader, core libraries anything the base OS itself depends on	→ **A/B rootfs** full image, reboot, proven rollback
Isolated or third-party apps own dependencies, own lifecycle	→ **Containers** sandboxed, needs writable storage
Native tools, scripts, services in /usr, /opt built against the host's libraries	→ **sysext** small, live, no reboot

A well-designed fleet uses each where it fits. Not one tool for everything.

Northern.tech

Immutable doesn't have to mean inflexible

01

sysext is activation

You still need delivery, health checks and rollback.

02

Works with any OTA framework

Install → check → commit or roll back. Mender is just one way.

03

Next to A/B, not instead

Big changes: full image. Small native changes: extension, live.

Resources

WRITE-UP + MODULE

Mender Hub post

hub.mender.io · systemd-sysext update module

REFERENCE

systemd-sysext docs

freedesktop.org · man systemd-sysext

GET IN TOUCH

Jana Perić

jana.peric@northern.tech

Northern.tech · Mender

Northern.tech