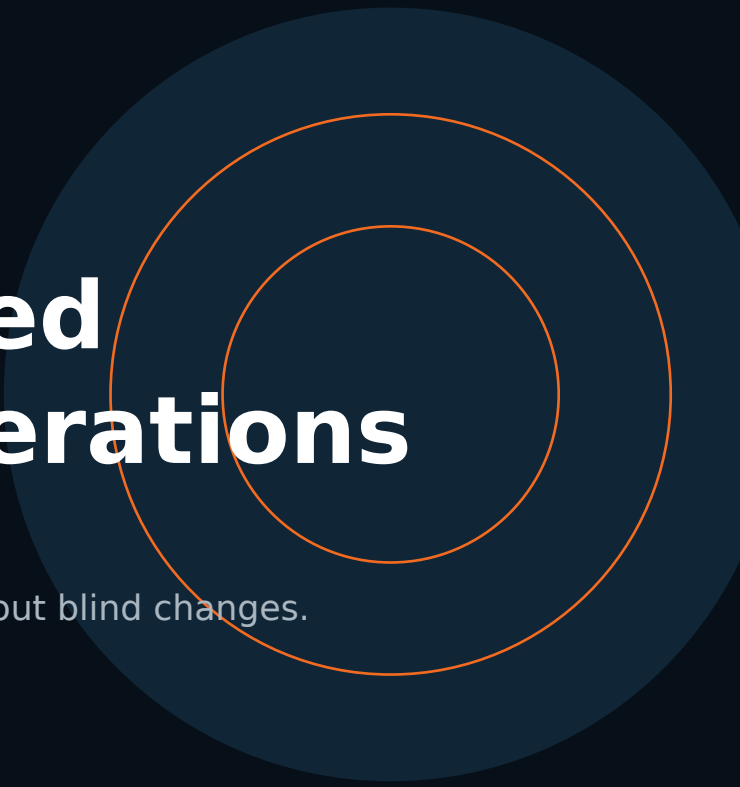


Evidence-based Linux/HPC operations

A decorative graphic consisting of three concentric circles in a dark blue color, centered on the right side of the page. The circles are of increasing size, with the innermost circle being the smallest and the outermost being the largest.

From symptom to verified recovery, without blind changes.

MICKAËL ANGEL

Linux HPC · HexaOS · EN edition

EVIDENCE-BASED OPERATIONS

Before you fix

A useful diagnostic method protects system state, time and the ability to roll back. It strictly separates observations from assumptions.

01 Define user impact

02 Record time in UTC

03 Name the system and scope

04 Avoid premature changes

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

01 · Frame the incident

A precise symptom is more valuable than a fast verdict. Describe what fails, since when, for whom and against which measurable criterion.

01 Expected service and current state

02 Scope: host, node, partition or cluster

03 Last known-good state

04 Operational risk and priority

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

02 · Freeze the timeline

Evidence becomes ambiguous when clocks diverge. Normalise time before correlating logs, metrics and events.

01 `date -u` and `timedatectl`

02 Explicit analysis window

03 NTP/PTP synchronisation

04 Timeline of known changes

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

03 · Collect without changing state

Start with read-only commands. Preserve raw output and the exact command that produced it.

01 `systemctl status and show`

02 `journalctl with UTC bounds`

03 `ss, ip, ps, free and df`

04 `squeue, sinfo and scontrol for Slurm`

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

04 · Form hypotheses

A hypothesis must explain the available evidence and predict an observation that could disprove it.

01 Candidate cause

02 Supporting evidence

03 Missing evidence

04 Discriminating, non-destructive test

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

05 · Choose the least risky test

Test what yields the most information for the least impact. A corrective action is not a discovery tool.

01 Read before writing

02 One change at a time

03 Before/after capture

04 Clear stop condition

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

06 · Prepare rollback

Rollback is designed before the change. It must be actionable, time-bounded and verifiable by someone else.

01 State to restore

02 Exact commands

03 Configuration backup or copy

04 Condition that triggers rollback

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

07 · Validate delivered service

An active process is not always a working service. Validate from the user perspective and across more than one cycle.

01 Functional test

02 Second boot or second job

03 Metrics back within baseline

04 No visible regression

OPERATIONS RULE

No READY without proof.

EVIDENCE-BASED OPERATIONS

A reusable runbook

Turn the incident into a shared asset: concise, reproducible and honest about the limits of the conclusion.

01 Summary and impact

02 UTC timeline

03 Evidence and commands

04 Probable cause and confidence

05 Action, rollback, validation, prevention

OPERATIONS RULE

No READY without proof.