

Backup Manager

Configuration backup, change control and bulk configuration for the switches, routers, firewalls and servers you already run — on your own server.

Version 1.5.1 · September 2026

When a switch is replaced at 2am, the question is never whether someone *can* rebuild it — it is whether last night's configuration still exists and what changed since. Backup Manager collects the configuration of every device on a schedule, tells you line by line what moved, checks it against your own rules, and puts it back when you need it. It runs on one Linux server inside your network. **Nothing leaves your premises, and there is no cloud account.**

One command

installs everything, including Node.js and the service

SSH · Telnet · HTTP

modern kit, legacy ciphers, and appliances with an export API

15-day trial

starts when you activate, not when a key was issued

Air-gapped

supported with a signed offline licence file

Getting devices in

A backup product is only as good as the day you have to fill it with two hundred devices.

Discover a subnet

Scan in CIDR form and see what answered, which ports are open and what the SSH banner says. Devices already in the system are marked so they cannot be added twice.

Add one, or all of them

Tick what you want and add it in a single step, with the platform, backup mode, schedule, retention and tags they should start with. A single device opens the full device form instead.

One login for a rack

A shared credential set can be created as you add. Every device points at it, so rotating that password later is one edit rather than forty.

CSV import and profile detection

Bring a list from wherever it lives, and let the software connect and work out which device profile matches what it finds.

Backing up

Four ways to collect

Copy files and folders over SFTP, run one command and keep its output, drive a device profile over SSH or Telnet, or download from an appliance's HTTP export API.

Schedules that behave

A time per device, staggered so forty switches do not all wake the network at once. A persistent queue retries failures with a widening gap and survives a restart.

Device profiles

Built-in command sequences for Cisco, Juniper, Aruba, MikroTik, D-Link, Sophos, Cambium and Ruckus — or write your own step by step, with enable passwords and pager handling.

Fast on big servers

Large trees are pulled over several SSH sessions at once and big files split into ranges, which took one real 1.6 GB backup from fourteen minutes to under five.

Knowing what changed

Change detection

Every capture is compared with the one before it. When a configuration differs you get a line-by-line diff, when it happened, and a full version history to walk back through.

Compliance rules

Check stored configurations against your own rules — Telnet still enabled, no AAA, weak SNMP strings, missing logging — across the estate or against one device, automatically after every successful backup.

Putting it back

Restore

Push any stored version back to a device, with a dry run first that shows exactly which commands would be sent. Every restore is logged with its full transcript.

Off-site copies

Mirror archives to S3-compatible storage, so a lost backup server is not a lost backup. Grandfather-father-son retention keeps daily, weekly, monthly and yearly copies.

Seeing the state of things

Health and alerts

One page that answers "what is broken right now" — never backed up, stopped, failing repeatedly. Alerts by email or webhook, with a threshold before anyone is told.

PDF reports

Backup summaries, compliance status, change history and device health as charted PDFs. On demand, or emailed on a schedule.

Fits what you run

A token-authenticated REST API, CSV export, Prometheus metrics and an audit log of every action.

Changing the network

Backing a device up tells you what it looks like. This is the other half.

Config Push

Write a set of commands once, pick the devices it should reach, and send it over SSH exactly as you would type it. Device selection is required — nothing is ever sent to "everything" by default.

Backed up before and after

Each device is snapshotted either side of the change, so the Changes page shows the exact diff the push produced and the previous configuration is on disk. A device that cannot be backed up is left alone rather than changed blind.

Try it on one device first

Open a live session on a single device from your selection, run the set there and watch the device answer line by line. Push to the rest only when it looks right.

A record you can act on

Per-device transcripts, how far the set got before it stopped, and the device's own words when it refused. Passwords written into a command reach the device as typed and are masked in the record.

A real terminal, in the browser

Every device has a **Console** button. It opens a session that behaves the way PuTTY does, because every key you press goes straight to the device rather than being collected into a line:

- **Tab** completes commands and paths
- **↑ ↓** the device's own command history
- **Colours and paging** drawn properly, --More-- included
- **? the context help** a switch prints mid-command
- **Ctrl-C** stops what is running, at once
- **Window size** travels, so the device wraps to what you see

Security

Stored credentials	Device passwords are encrypted with a key held separately from the session secret. Shared credential sets keep one login in one place.
Host key checking	Trust on first use, or strict. A fingerprint that changes stops the backup instead of handing credentials to whoever answered.
Archives	Optionally encrypted on disk, checksummed after writing, and verified.
Roles	Administrator, operator and viewer. Changing devices needs operator or above; a viewer cannot open a console.
Accountability	An audit log of every action, login history, and lockout after repeated failed logins.
The service itself	Runs as its own unprivileged account under systemd, with no new privileges and a read-only system tree. Updates are applied by a separate privileged helper; the web service never holds root.

Licensing

A trial that is a full trial

Fifteen days, issued automatically at install. The clock starts when the software is first activated, not when a key was issued.

Licensed per installation

Bound to the machine it runs on. Extending a term is done centrally and the installation picks it up on its next check — nothing to reinstall.

No route to the internet?

Send us the machine fingerprint from the Licence page and import the signed licence file we send back. Nothing else is needed, ever.

Your data is never held hostage

If a licence lapses, new backups pause — but viewing, downloading, comparing and restoring everything already collected keeps working. Your configuration history stays yours.

What it needs

Operating system	Ubuntu 20.04 / 22.04 / 24.04 LTS, Debian 11 / 12, or CentOS / RHEL / Rocky / Alma 8, 9 or 10
CPU and memory	Two cores and 2 GB of RAM is comfortable for a few dozen devices
Disk	Depends on what you back up. Network device configurations are tiny; a server file tree is not
Network	The server reaches your devices on SSH (22), Telnet (23) or HTTPS — whichever each device speaks
Internet	Once, to install dependencies and fetch the licence. Air-gapped installs are supported
Prerequisites	None to prepare. The installer adds Node.js and any build tools that are missing

Installing and updating

```
$ curl -fsSL https://opsec.co.in/opsec/backup-manager/get.sh | sudo bash
# the installer checks the download against a published SHA-256,
# creates a service account, registers the service and starts it
```

- **Updates from the interface.** The Licence page shows what is available and what changed, with an Install button and a progress bar.
- **Reversible.** The database is copied aside, the new version goes in beside the old one, and the previous version is put back automatically if it does not come up.
- **Your data is untouched.** Backups, configuration and history are never replaced by an update.

Try it

One command on a spare Linux server and a fifteen-day trial starts itself.

opsec.co.in/products/backup-manager.html

Talk to us

Pricing, a walkthrough, or a question about your own kit.

info@opsec.co.in