

Uplivra is in beta and online purchasing is paused. **Apply to test it:** testers get a license on request and 10% off when purchasing opens.

INSTALL GUIDE · AUTOMATION & REMEDIATION

Automation runbooks

Write a runbook, try it with a dry run, run it by hand, on a schedule or when an alert opens, and see every run with its version, approvals, rollback and linked tickets.

Uplivra Technologies LLC · Guide for Uplivra 26.10 · Updated 28 September 2026 · Latest version:
<https://uplivra.com/guides/automation.html>

What you need

- The **Automation & Remediation** module on your license. Without it, the Automation page only explains what runbooks do.
- An administrator, or someone whose access profile has **Write and change runbooks**, to write runbooks. Running and approving runs needs **Run and approve runbooks**. Seeing them needs **See runbooks and runs**.
- For SSH steps: an SSH login saved under **Settings > Device logins**. Runbooks name the login; they never hold the password or key.
- Optional: the **Native Service Desk** module to open Service Desk tickets for runs, and **ITSM Integration** to open tickets in ServiceNow, Jira Service Management, Freshservice, Zendesk or a webhook.

Automation adds **no new ports**. SSH steps run on the collector at the device's site, as other SSH work does. Web steps leave from the Uplivra server.

Write a runbook

Go to **Tools > Automation > New runbook**.

1. Give it a name, and say what it's for.
2. Add steps. Each step does one thing:

Step	What it does
Run commands over SSH	Runs commands on the device. Tick <i>Network device</i> for switches and routers that need an interactive shell.
Call a web address	Sends a GET, POST, PUT, PATCH or DELETE to another system, with your headers and body. It may reach internet addresses and the run's own device, nothing else inside your network.
Wait	Waits 1 to 900 seconds.
Re-run the device's checks	Runs the device's checks again, to confirm the fix worked.
Acknowledge the alert	Acknowledges the alert that started the run.

In commands, addresses and bodies you can use `{{device.name}}`, `{{device.address}}`, `{{alert.title}}` and `{{alert.id}}`.

A failed step stops the run, unless you tick **Keep going if it fails**.

Tick **Read-only** on a step that only looks, such as `show interface status` or a GET. A dry run really does read-only steps (see below). Only tick it when the step changes nothing.

Start a run

There are three ways:

- **By hand.** On the Automation page or the runbook's page, pick a device and press **Run**.
- **When an alert opens.** Choose *Also when an alert opens that matches*, and say which device, tag, title or severity. A **cooldown** stops it running again for the same device too soon. Alert-started runs never start while a maintenance window covers the device.
- **On a schedule.** See the next section.

Approval. Tick *Every run waits for approval* and each run, however it started, waits until another administrator approves it. The person who started a run can't approve it. While a run waits, Uplivra keeps an alert open saying so.

Schedules

Under **Schedule** pick one of:

Runs	Settings
Every so many minutes or hours	5 to 1440 minutes, or 1 to 168 hours
Every day	at a time
On days of the week	tick the days, and a time
Once a month	a day of the month (1-31) and a time. Days past the month's end run on its last day.

Time zone. Leave it empty to use the server's time zone, or type a name such as `Europe/London` or `America/Chicago`. When the clocks go forward and a time doesn't exist that day, the run happens an hour later.

Device. If the runbook has SSH or check steps, pick the device scheduled runs act on.

Maintenance windows. If an approved maintenance window covers the device (or everything) when a run is due, the runbook can:

- **Skip that run.** It shows in the run list as *skipped*, with the window's name. The next one runs as usual.
- **Run it when the window ends.** The Automation page shows *Waiting for a maintenance window to end*.
- **Run anyway.**

The Automation page and the runbook page show when it runs next. Scheduled runs follow the same approval rule as every other run, and show *its schedule* as who started them. A runbook that is switched off doesn't run on its schedule.

Dry runs

Press **Dry run** to see what a runbook would do, without changing anything. A dry run doesn't need approval. For each step it shows:

- the **target**: the device, address, port and login, or the web address and the IP it resolves to;
- **what it would send or run**: the commands, or the method, headers and body, with placeholders filled in. Header values are hidden, because they often hold keys;
- a **problem**, if the real run would fail: a missing SSH login, no device, or a web address Uplivra would refuse.

Waits are skipped. Re-running checks really runs, because it only looks. Steps marked read-only really run too, and say *read-only: really ran*, but only when the runbook doesn't need approval and you may run runbooks. Otherwise they are only shown, so a dry run is never a way round approval. Dry runs are kept in the run history, marked *dry run*.

Versions

Every save makes a new numbered version, with who saved it, when, and the note in **What changed**. Each run records the version it ran.

Open **Version ... · history** on a runbook to:

- **View** any version;
- **Compare** two versions, line by line (steps and settings);
- **Restore** an old version. It is saved again as a new version, so nothing is lost.

A run's page links to what changed if the runbook has changed since.

Rollback steps

Under **Rollback** add steps that put things back, such as starting a service a step stopped. Choose the order:

- **Undo the completed steps, last first**. Each rollback step says which step it undoes. Uplivra goes back through the steps that finished, last first, and runs their rollback steps. Rollback steps that undo no particular step run at the end.
- **Run the rollback steps as listed**.

Tick **Roll back by itself when a step fails** to roll back straight away. Otherwise, or if a rollback failed, press **Roll back** on the failed run. Every rollback step is tried, even if one fails. The run shows each rollback step's result and who started the rollback.

How many runs at once

Each runbook allows **one run at a time** unless you raise it (up to 50). When that many are going, a new run either **waits in the queue** or is **refused** with a message saying why.

The whole installation also has a limit, **5 at once** unless you change it on the Automation page. In a multi-company (MSP) installation, only the service provider's administrators can change it. Queued runs start by themselves, oldest first, when a slot is free. The Automation page shows how many are running and queued, and a queued run's page says how many are ahead of it. Dry runs don't count toward either limit.

Tickets and change records

A run can be linked to:

- **A Service Desk ticket** (Native Service Desk). Set *Each run opens a Service Desk ticket* on the runbook, or press **Open a Service Desk ticket** on a run, or link an existing ticket by number. The ticket's page lists its runs. When the run finishes, Uplivra adds a note to the ticket.
- **A ticket in another system** (ITSM Integration). Set *Each run opens a ticket in ...*, or press **Open a ticket there** on a run. Uplivra adds a note to that ticket when the run finishes. You can also type in any ticket's reference and web address.
- **A change record**. Link a maintenance window to the run. A run on a device that is in a maintenance window links to it by itself.
- **The alert**. Runs started by an alert link to it, and to the alert's own ticket, by themselves.

Audit

Every change is in **Settings > Audit log**: saving, restoring and deleting runbooks, starting, approving, turning down, cancelling, queueing, refusing and skipping runs, dry runs, rollbacks, links and the installation's limit.