RadiusForge

Documentation

Install, operate and deploy your self-hosted FreeRADIUS network — subscribers, speed plans, live sessions and hotspot vouchers.

For ISPs & WISPs · Version **2026.10.17** · radiusforge.co.uk

Contents

RadiusForge Documentation · six guides

1 **Overview** 3

What RadiusForge is and where to start

2 **Quick Start** 5

From download to a working console

3 **Install Guide** 7

Download, verify and install, step by step

4 **Getting Started** 10

Concepts, then your first setup

5 **User Guide** 12

Every screen, including hotspot vouchers

6 **Deployment Guide** 16

Server prep, HTTPS, production, upgrades

Overview

What RadiusForge is and where to start

What RadiusForge is

RadiusForge is a friendly web console for running a **FreeRADIUS** network, built for internet service providers (ISPs) and wireless providers (WISPs). It lets you manage subscribers, speed plans, live sessions and Wi-Fi hotspot vouchers from clean web pages — without needing to understand raw RADIUS attributes or edit configuration files by hand.

Underneath, RadiusForge installs and manages the proven, standard open-source stack — **FreeRADIUS**, **MySQL**, and a hardened web application — on a server you own. Your subscriber data never leaves your hardware.

> **New to RADIUS?**
> RADIUS is the system that decides, in an instant, whether someone connecting to your network is allowed in, how fast their connection should be, and how much data they have used. RadiusForge is the control panel for that system. The **Getting Started** guide explains all of this assuming no prior knowledge — start there if any of this is unfamiliar.

Who it's for

- **ISPs and WISPs** authenticating broadband, fixed-wireless or PPPoE subscribers.
- **Hotspot operators** — campsites, hotels, cafés, marinas — who sell or hand out Wi-Fi access with printed voucher codes.
- Anyone replacing an ageing daloRADIUS / RADIUS Manager setup, or moving off hand-edited FreeRADIUS, who wants a modern, supported console.

What you can do with it

- Add **subscribers** and assign them a **speed plan** in plain Mbps — no attribute syntax.
- Register your **routers** (MikroTik, Juniper, Fortinet and any standards-compliant RADIUS gear).
- Watch **who is online right now**, and disconnect or look up any session.
- Issue **hotspot vouchers** in batches, print a tidy voucher sheet, and have them expire by time or data automatically.
- See **reports** on data use and every login attempt, with a plain-English reason when a login is refused.
- Run several networks from one install with **multi-tenant organisations**, each kept separate.
- Secure staff logins with **two-factor authentication**, and check passwords against your own **LDAP / Active Directory** if you prefer.

What's in this pack

Your download includes this documentation set so you can get running without an internet search:

Guide	Read it when
Quick Start	You want the fastest path from download to a working console.
Install Guide	You want the full, careful install with every check explained.
Getting Started	You're new to RADIUS and want the concepts, then your first setup.
User Guide	You want every screen explained, including hotspot vouchers.
Deployment Guide	You're preparing a production server — OS choice, HTTPS, upgrades.

What you need

- A server of its own running **RHEL 8/9, Rocky Linux, AlmaLinux**, or **Ubuntu 22.04 / 24.04 LTS** (a virtual machine is fine).
- Internet access on that server during installation.
- Root (`sudo`) access.

RadiusForge installs its own FreeRADIUS and MySQL, so it needs a clean server — it will not install alongside an existing RADIUS or database server, and it checks for that before making any change.

Licence and trial

A fresh install runs on a **30-day free trial** — no key needed to begin. After the trial, a licence keeps the console editable. Your subscribers keep authenticating no matter what; a lapsed licence only makes the console read-only, it never interrupts anyone's connection.

Getting help

Questions, licences, or anything else: **info@radiusforge.co.uk** — or visit **radiusforge.co.uk**.

Quick Start

From download to a working console

This is the fast path — download, install, log in. If you'd like every step explained in detail, see the **Install Guide** and **Getting Started**.

1. Check you have a suitable server

- A clean server running **RHEL 8/9, Rocky Linux, AlmaLinux**, or **Ubuntu 22.04 / 24.04 LTS** (a VM is fine).
- **Root /** `sudo` access, and **internet access** on the server during install.
- The server reachable on your network from the computer you'll use to log in.

> **It needs a server of its own**
>
> RadiusForge installs its own FreeRADIUS and MySQL. Don't install it on a machine already running RADIUS or a database — it checks for this and stops before changing anything.

2. Download and verify

On the server:

```
curl -LO https://radiusforge.co.uk/download/radiusforge-2026.10.17.tar.gz
```

Check the download is intact. The expected SHA-256 is published on **radiusforge.co.uk/#download**:

```
sha256sum radiusforge-2026.10.17.tar.gz
```

The value it prints should match the one on the website exactly.

3. Install (one command)

Extract into `/opt/radiusforge` and run the installer for your operating system:

```
sudo mkdir -p /opt/radiusforge
sudo tar -C /opt/radiusforge --strip-components=1 -xzf radiusforge-2026.10.17.tar.gz
cd /opt/radiusforge

sudo bash scripts/install-rhel.sh       # RHEL / Rocky / AlmaLinux
# or, on Ubuntu / Debian:
sudo bash scripts/install-debian.sh
```

The installer sets up MySQL, FreeRADIUS, the web app and a hardened background service in a single pass. It asks a few questions (database name, an application password, your time zone) and then asks you to **create your admin login**.

> **A one-minute pause is normal**
>
> On its first start, MySQL initialises its data area. The installer waits at "*Waiting for MySQL to accept connections*" — that pause is normal, not a hang.

When it prints **"Hardening complete,"** the console is live at `http://your-server-ip/` — no port number, no extra steps.

4. First login

Open `http://your-server-ip/` in a browser and sign in with the admin account you just created. A **welcome banner** invites you to **Create your first Organisation** — name it after your network, and you're ready to add plans, subscribers and routers.

You'll also see a **"Trial: 30 days left"** banner — that's your free trial running.

5. Secure it (before real use)

To serve the console over HTTPS (required for the hotspot customer portal):

```
sudo bash scripts/https-setup.sh letsencrypt \
    --name radius.example.com --email you@example.com --agree-tos
```

Replace the name and email with your own. See the **Deployment Guide** for self-signed and private-CA options.

6. Add your licence (when you have one)

A new install runs on a **30-day trial**. When you receive your licence file (a `.key` sent to you separately), open **Licence** in the top menu and upload it. The licence only controls whether staff can make changes — your subscribers keep authenticating regardless.

Where to go next

- **Getting Started** — the concepts, then your first organisation, plan and subscriber.
- **User Guide** — every screen, including hotspot vouchers.
- **Deployment Guide** — HTTPS, production hardening and upgrades.

Questions? **info@radiusforge.co.uk**

Install Guide

Download, verify and install, step by step

This guide walks through installing RadiusForge carefully, with every step explained. It works the same way on both supported operating-system families; where they differ, it's called out. If you just want the short version, see the **Quick Start**.

Before you begin

RadiusForge needs **a server of its own**. It installs and configures its own FreeRADIUS and MySQL, so it will not install next to an existing RADIUS server or database server — a safety check stops the install if it finds one, without changing anything.

You will need:

- **A clean Linux server** — physical or virtual — running one of:
  - **RHEL 8/9**, **Rocky Linux**, or **AlmaLinux** (the Red Hat family), or
  - **Ubuntu 22.04 / 24.04 LTS** or **Debian 12** (the Debian family).
- **Root /** `sudo` access on that server.
- **Outbound internet access** during install, so the installer can fetch the packages it depends on.
- The server **reachable on your network** from the computer you'll log in from.

> **RHEL needs a subscription; Rocky and Alma don't**
> Red Hat Enterprise Linux requires you to register it (`subscription-manager register`) before it can install software — that needs your own Red Hat account and can't be automated. **Rocky Linux** and **AlmaLinux** are free, RHEL-compatible alternatives with no subscription step, and everything here applies to them unchanged.

Step 1 — Download

On the server, download the latest release:

```
curl -LO https://radiusforge.co.uk/download/radiusforge-2026.10.17.tar.gz
```

A single download covers every supported operating system — it contains both installers plus all the documentation.

Step 2 — Verify the download

Always check the file arrived intact and unaltered. The expected **SHA-256** checksum is published on **radiusforge.co.uk/#download**. Produce the checksum of your copy:

```
sha256sum radiusforge-2026.10.17.tar.gz
```

Compare the long value it prints against the one on the website — they must match character-for-character. If they don't, delete the file and download it again.

> **Why verify?**
> The checksum proves two things at once: the download wasn't corrupted in transit, and it's the genuine file we published. It takes two seconds and is good practice for any server software.

Step 3 — Install

Extract the package into `/opt/radiusforge` and run the installer for your operating system:

```
sudo mkdir -p /opt/radiusforge
sudo tar -C /opt/radiusforge --strip-components=1 -xzf radiusforge-2026.10.17.tar.gz
cd /opt/radiusforge
```

Then, on the **Red Hat family** (RHEL / Rocky / AlmaLinux):

```
sudo bash scripts/install-rhel.sh
```

...or on the **Debian family** (Ubuntu / Debian):

```
sudo bash scripts/install-debian.sh
```

> **`/opt/radiusforge` is the right home**
> `/opt` is the standard location for third-party server software on Linux, and the background service is allowed to run from there. If you happen to unpack it under `/root` or your home folder instead, the installer will **automatically move it to `/opt/radiusforge`** and continue — so the one-command install works wherever you extracted it.

What the installer does

In a single pass, with no separate "now start it" step, it:

- installs **MySQL**, **FreeRADIUS** (with the SQL and LDAP modules and disconnect/CoA support), **Python** and **Nginx**;
- creates the application database and FreeRADIUS's database back-end;
- runs the database setup;
- asks you to **create your first admin login**;
- installs the production service — **Gunicorn** behind **Nginx** — running as a locked-down, non-root `radiusforge` system account that starts on boot and restarts itself if it ever crashes;
- opens the firewall for the web console and applies secure defaults.

It first **lists every system-wide change it will make** and asks only a few questions: the database name, the application username and password, the time zone to show times in, and (optionally) a MySQL root password.

> **The MySQL pause is normal**
> The first time MySQL starts, it initialises its data directory, which can take a minute on a slow or busy server. The installer waits at "*Waiting for MySQL to accept connections*" — leave it; it is not stuck.

When it finishes it prints **"Hardening complete,"** and shows the address the console is now live at, e.g. `http://192.168.1.50/`.

> **Safe to re-run**
> If the install is interrupted, you can run the same command again — it's written to pick up safely rather than double-apply.

Step 4 — Confirm it's running

Check the four services are active:

```
# RHEL / Rocky / AlmaLinux:
systemctl is-active radiusforge radiusd nginx mysqld

# Ubuntu / Debian:
systemctl is-active radiusforge freeradius nginx mysql
```

Each should report `active`. Then confirm the web server answers:

```
curl -I http://localhost/
```

You should see `HTTP/1.1 302 Found` redirecting to `/accounts/login/` — that's the console asking you to sign in.

> **RADIUS service name differs by OS**
> On the Red Hat family the RADIUS service is called **`radiusd`**; on the Debian family it's **`freeradius`**. RadiusForge manages it for you — you only need the name if you ever restart FreeRADIUS by hand.

Step 5 — First login

Open `http://your-server-ip/` in a browser and sign in with the admin account you created during install. You'll see:

- a **"Trial: 30 days left"** banner — your free trial, and
- a **welcome banner** inviting you to create your first organisation.

Continue in the **Getting Started** guide to set up your first organisation, plan and subscriber.

Step 6 — Secure the console (HTTPS)

The installer serves plain HTTP so you can log in immediately. Before real use — and required for the hotspot customer portal — put it behind HTTPS:

```
sudo bash scripts/https-setup.sh letsencrypt \
    --name radius.example.com --email you@example.com --agree-tos
```

The **Deployment Guide** covers self-signed (lab) and private-CA options, and how to check certificate status.

Troubleshooting

A service isn't `active`

Look at its log. On the Red Hat family: `journalctl -u <service> -b` (this boot) and `systemctl status <service>`. FreeRADIUS also logs to `/var/log/radius/radius.log` (RHEL) or `/var/log/freeradius/radius.log` (Ubuntu).

The install stopped saying the server is already in use

That's the safety check: RadiusForge found an existing FreeRADIUS, database, or busy ports. Install on a fresh server or VM instead — the message lists what it found.

A “Permission denied” that doesn't make sense (RHEL)

RHEL's SELinux adds extra permission layers. The installer handles the known cases; if you're troubleshooting an unusual setup, `sudo ausearch -m avc -ts recent` shows exactly what SELinux blocked.

Still stuck? Email **info@radiusforge.co.uk** with the output you're seeing.

Getting Started

Concepts, then your first setup

> **Who this guide is for**
>
> This guide assumes you know little or nothing about RADIUS, MySQL or Linux servers. If you're already comfortable with those, skip ahead to the **User Guide**. If you haven't installed RadiusForge yet, start with the **Install Guide** or **Quick Start**.

What RadiusForge actually does

Every time someone connects to your network — a home broadband customer whose router dials in, a Wi-Fi hotspot user, a VPN user — something has to answer three questions instantly:

1. **Is this person allowed to connect?** (checking a username and password)
2. **If yes, how fast should their connection be, and what settings do they get?**
3. **How long were they connected, and how much data did they use?**

The system that answers those questions is called **RADIUS**. Think of it as a doorman who also sets the rules: every router or access point asks RADIUS "*can this person in?*", RADIUS checks its records and says yes or no, and if yes, tells the router exactly what that person may do — their speed limit, for example.

RadiusForge is the control panel for that doorman. The actual RADIUS software — a program called **FreeRADIUS** — is powerful but was built for network engineers, and configuring it normally means editing text files full of jargon. RadiusForge gives you web pages instead: add a customer, pick their speed plan from a dropdown, done.

The words you'll see everywhere

- **Subscriber** — one of your customers. Has a login name and password that their router or device uses to connect.
- **Plan** — a service tier, like "Home 50/10" or "Business 200/50". Sets the speed limit (and other settings) a subscriber receives.
- **Router (NAS)** — any router, access point or firewall that sends login requests to RadiusForge on your customers' behalf. In RADIUS jargon this is a "NAS" (Network Access Server) — nothing to do with a file-storage box. RadiusForge just calls them routers. Every one your customers connect through must be registered.
- **Session** — one continuous connection, from when a subscriber connects to when they disconnect. RadiusForge keeps a history.
- **Voucher** — a one-off code (for a Wi-Fi hotspot) that grants access for a set time or amount of data, then stops working. Great for campsites, hotels and cafés.
- **Organisation** — a self-contained network inside RadiusForge, with its own subscribers, plans, routers and staff. Run several and they stay completely separate. Most people need just one.
- **Staff** — the people who log into RadiusForge itself to manage things (not the same as subscribers). Staff have roles: **Owner** (full control, including who else has access), **Administrator** (full day-to-day control), and **Support** (can see everything, can't change anything).

Your first login: create your organisation

The first time you sign in with the admin account you made during install, RadiusForge shows a **welcome banner**: "*Create your first Organisation to start adding subscribers, plans and routers.*"

1. Sign in with that admin account.
2. Click **Create your first Organisation** in the welcome banner.
3. Give it a name — your ISP or company name, e.g. `My ISP` — and click **Create Organisation**.
4. You land back on the Dashboard as the **Owner** of your new organisation. Subscribers, Plans, Routers and Staff are all ready to use.

> **Name it for real**
>
> The organisation name appears around the console and on hotspot voucher sheets. Use the name you'd be happy for staff — and voucher users — to see.

A sensible first setup, in order

Set things up in this order and each step has what it needs from the one before.

1. Create a plan

Plans → Add plan. A plan is a speed tier. For a MikroTik network, pick **MikroTik** as the router platform and enter the download and upload speeds in plain Mbps (e.g. 50 and 10). Save. That's a complete plan.

2. Register a router

Routers → Add router. Enter:

- the **IP address** the router sends its requests from,
- a **shared secret** (a password-like value you set identically here and on the router), and
- the **platform** (e.g. MikroTik).

This tells RadiusForge to trust that router.

> **The shared secret must match exactly**
>
> The shared secret has to be typed identically here and in the router's own RADIUS settings. If it doesn't match, the router's requests are silently ignored — this is the single most common cause of "it isn't working."

3. Add a subscriber

Subscribers → Add subscriber. Give them a **username** and **password** (what their router/device will use to connect) and choose the **plan** you just made. Save. The subscriber can now authenticate.

4. Test the login

On the **Subscribers** page, use **Check a login** — type the username and password and RadiusForge tells you whether they'd be accepted, and why, without touching a router. A quick way to confirm everything lines up before real traffic arrives.

Your licence and trial

A new install starts on a **30-day trial** — you'll see a banner counting down. Your subscribers are authenticated the whole time regardless. The licence only controls whether staff can make **changes** in the console. When you have a licence file, upload it under **Licence** in the top menu. If a licence ever lapses, the console becomes read-only until you add a valid one — **logins are never interrupted**.

Where to go next

- **User Guide** — every screen explained in full, including **hotspot vouchers**.
- **Deployment Guide** — HTTPS, production hardening and safe upgrades.

Questions? **info@radiusforge.co.uk**

User Guide

Every screen, including hotspot vouchers

Before you start

This guide walks through every screen in RadiusForge, written for someone who has never used a RADIUS management tool. If you haven't installed RadiusForge or created your first organisation yet, read **Getting Started** first.

Logging in

Go to your server's address in a web browser and enter your username and password. If two-factor authentication is switched on for your account (see **Security** below), you'll also enter a 6-digit code from an authenticator app after your password.

A **trial banner** across the top shows how many days remain on your free trial; a **welcome banner** appears until you've created your first organisation.

The Dashboard

The first page after logging in. It's a live summary of the selected organisation:

- **Status cards** — whether passwords are checked locally or against your directory, whether your last change is in use, whether your routers are sending sessions, and the state of your secure (HTTPS) address.
- **Headline numbers** — total subscribers, how many are online now, logins today, and data used this week.
- **Charts** — logins (accepted vs refused) and data used over the last seven days.
- **Recent login attempts** and **Top users this week**, with links to the full reports.

There's nothing to configure here — it's your at-a-glance health check.

Plans — your service tiers

Plans → Add plan. A plan is a service tier — "Home 50/10" or "Business 200/50". Every subscriber is on exactly one plan, and the plan controls their speed and other settings.

- **Name** — whatever you want to call it, e.g. `Home 50/10`.
- **Router platform (vendor)** — the most important choice here; it changes which fields appear:
  - **MikroTik** — the simplest case. Enter download and upload speeds in plain Mbps. Done.
  - **Cisco** — instead of a speed, you name a QoS policy that already exists **on the router**. RadiusForge references it; it can't create it remotely.
  - **Juniper** — same idea: you name a traffic-control profile already configured on the router.
  - **Fortinet (FortiGate)** — you name a FortiGate **user group**; the speed limit is a traffic-shaping rule tied to that group on the FortiGate itself.
  - **Generic / other** — for any router not listed. Works like MikroTik (plain speed fields) using standard RADIUS attributes.
- **Simultaneous use** — how many devices can be connected at once on this plan. Residential plans usually use `1`.
- **Session / idle timeout** — optional; leave blank unless you want to force a disconnect after a fixed time.

Only using MikroTik? Ignore the vendor complexity

If all your gear is MikroTik, pick MikroTik and fill in two speed numbers — you never need to think about the other vendors.

Plan addressing uses a pool, not a single IP

A single fixed IP belongs to one subscriber, so you set that on the subscriber (below). On a plan you can assign an address **pool** instead (via Advanced Attributes). Size it for the plan: a `/24` pool holds **254** usable addresses, so at most 254 subscribers on that plan can be online with an address at once.

Advanced Attributes (shown after you save a plan) attach extra settings to *every* subscriber on the plan — an address pool, a DNS server, a firewall rule reference and more. Pick from the dropdown, which explains what each option expects.

Most people never need this.

Subscribers — your customers

Subscribers → Add subscriber.

- **Username / Password** — what the subscriber's router or device uses to connect. (Not their RadiusForge login — subscribers don't log into RadiusForge.)
- **Plan** — the service tier they're on.
- **Static IP address** (optional) — a fixed IPv4 address for this subscriber, sent as `Framed-IP-Address`. Leave blank to let the router assign one. Use it for business customers who need a known address.
- **Status** — **Active**, **Suspended**, or **Cancelled**.

> **"Suspended" is the button for non-payment**
> Suspend instantly blocks a customer from connecting without deleting their account or settings. Switch them back to Active the moment they pay and everything returns exactly as it was.

Check a login (on the Subscribers page) tells you whether a given username and password would be accepted — and why or why not — without contacting a router. Use it to confirm a setup or diagnose a complaint.

Advanced Attributes (on a saved subscriber) add per-subscriber extras, most usefully a **routed subnet** to the customer's own router (`Framed-Route`, e.g. a `/29`).

Routers — your network equipment

Routers → Add router. Every router, access point or firewall your subscribers connect through must be registered so RadiusForge trusts it.

- **IP address** — the address the router sends its requests *from*. For a router with several connections or behind NAT, this may not be the address you expect. Each address can be registered only once.
- **Shared secret** — a password-like value set identically here and in the router's RADIUS settings.
- **Platform** — the same vendor list as plans.

> **Mismatched shared secret = silent failure**
> If the shared secret doesn't match exactly on both sides, the router's requests are ignored with no error. It's the first thing to check when a router "isn't working."

Changes take effect within a few seconds. **Deleting** a router removes it *and* stops it being trusted, so its customers can no longer log in through it — you'll be asked to confirm.

Sessions — who's online, and history

Sessions. The top section, **"Online right now,"** lists everyone connected. Each row has two buttons:

- **Disconnect** — actually drops the connection now (RadiusForge tells the router to disconnect). The customer must log in again.
- **Mark as ended** — only tidies RadiusForge's own records; it does **not** drop a live connection. Use it when a session is stuck showing online because a router lost power without telling RadiusForge.

The lower section, **"Recently ended sessions,"** is a searchable history — filter by username, router IP, or date range.

Hotspot vouchers

Vouchers let you sell or hand out Wi-Fi access as one-off codes that expire by time or data — ideal for campsites, hotels, cafés and marinas. RadiusForge issues and enforces the codes; your **MikroTik hotspot hosts the login page**, so there's no portal to build.

> **Turn vouchers on first**
> Vouchers are off until an Owner enables them. Go to **Organisation** settings and switch on **vouchers**. A **Vouchers** item then appears in the top menu. (An organisation that never enables vouchers behaves exactly as before.)

Create a voucher batch

Vouchers → create a batch. You choose:

- **Plan** — the speed the vouchers get (reuses a normal plan, e.g. "Hotspot 10/5").
- **Quantity** — how many codes to generate.
- **Code prefix** — a short label in front of each code, e.g. `CAFE-`, so a glance tells you which batch a code came from.
- **Validity** — either **from first use** ("valid 24 hours / 7 days from first login" — the model most hotspots actually sell) or a **fixed window** (between two dates).
- **Data cap** (optional) — total data the voucher allows before it stops.
- **Simultaneous use** — devices per code (usually 1).
- **Price label** (optional) — a printed label only, e.g. `£5`. No money is processed.

RadiusForge mints the codes using an unambiguous alphabet (no look-alike `0/O` or `1/I/L`), each globally unique.

Print or export the sheet

Open the batch and use **Print** for a tidy, self-contained voucher sheet — a stack of cards, each showing your network name, the code, the plan and the price label — ready to cut up at reception. You can also **export to CSV** for your records or a mail-merge.

How a voucher behaves

- It starts as **unused**. On the customer's **first successful login**, RadiusForge stamps it active and (for "from first use" vouchers) computes its expiry from that moment.
- While active, the customer gets the plan's speed. RadiusForge tracks data used against the cap.
- When it **expires** or **hits its data cap**, RadiusForge stops it working and disconnects any live session on it. A used-up or expired code can't be reused.

The batch view shows live counts — unused, active, expired, used-up — who's currently online on a voucher, and data used.

> **Why “from first use” matters**
>
> A campsite guest who buys a 7-day voucher on Monday but first logs in on Wednesday should get until the following Wednesday — not have two days silently eaten. "From first use" does exactly that, which cheaper tools often get wrong.

Connect your MikroTik hotspot

On the MikroTik, point its hotspot at RadiusForge as the RADIUS server (matching the **shared secret** you set when registering the router), and set the hotspot profile to use RADIUS. Guests then see the MikroTik login page and enter their voucher code; RadiusForge authenticates it and applies the plan. (A one-paste setup snippet is provided with the hotspot materials.)

> **Hotspot specifics**
>
> Hotspot logins use CHAP, so voucher passwords are stored in a form FreeRADIUS can check — handled automatically. Vouchers deliberately don't assign a fixed IP per code; hotspot guests take an address from the router's own pool.

Reports

Reports.

- **Top Users** — who used the most data or time across recorded history.
- **Connection Log** — every login attempt, successful or failed, newest first. For failures, RadiusForge gives its best-guess reason (wrong password, unknown username, suspended account) — a genuinely useful hint, not a certainty, since RADIUS itself doesn't record the exact reason.

Staff — who can log into RadiusForge

Staff (visible to Owners). Add teammates with a username and an initial password (share it securely — there's no password-reset email yet), and a role:

- **Owner** — full access, including this Staff page.
- **Administrator** — full day-to-day access, but can't manage staff or roles.
- **Support** — can view every page, but every Add / Edit / Save / Remove button is blocked. Ideal for a support contractor.

Owner-only pages

- **Authentication** — choose where subscriber passwords are checked: stored locally in RadiusForge, or against your own **LDAP / Active Directory** server.

- **Organisation** — your organisation's settings: its name, the hotspot **vouchers** switch, the customer-portal switch, how emails are sent, and API keys.
- **Activity** — a read-only log of who changed what, and when. Downloadable as CSV.
- **Licence** — your licence status and days remaining, and where you upload a new licence file.

Security — two-factor authentication

Security (next to your username, top right). Two-factor is **per person** — each staff member turns it on for their own login.

Click **Enable Two-Factor Authentication**, add the shown key to an authenticator app (**Google Authenticator, Microsoft Authenticator, Authy** or most password managers — use "enter a setup key manually" if you can't scan the QR code), type back the 6-digit code, and you're done. From then on, login needs that code as well as your password.

> **Keep a backup of your authenticator**
>
> If you lose access to your authenticator app, an Owner currently can't reset two-factor for you from the console. Back up your authenticator, or keep its recovery option set up.

Moving from an existing RADIUS server

RadiusForge can read an existing FreeRADIUS server to help you move across. The assessment is **read-only** — it leaves the old machine exactly as it found it. Your old server keeps serving customers until you decide to cut over, router by router, at your own pace. There's no data lock-in either: underneath it's a standard FreeRADIUS + MySQL back-end.

Questions, or help planning a migration? **info@radiusforge.co.uk**

Deployment Guide

Server prep, HTTPS, production, upgrades

This guide covers running RadiusForge in production: choosing and preparing a server, networking, HTTPS, the post-install checks worth doing, and safe upgrades. For the basic install itself, see the **Install Guide**.

What you end up with

A single server running **MySQL**, **FreeRADIUS** and the **RadiusForge** web console, reachable at `http://your-server-ip/` (then HTTPS once you set it up). No separate application server is needed — it's the "install it, browse to it" model, done properly with a hardened service.

Choosing the operating system

RadiusForge supports two OS families, and behaves identically on both — only the underlying plumbing differs, which the installer handles.

	Red Hat family	Debian family
Versions	RHEL 8/9, Rocky Linux, AlmaLinux	Ubuntu 22.04 / 24.04 LTS, Debian 12
Installer	`scripts/install-rhel.sh`	`scripts/install-debian.sh`
RADIUS service	`radiusd`	`freeradius`
FreeRADIUS config	`/etc/raddb/`	`/etc/freeradius/3.0/`
Firewall	firewalld	ufw
Subscription needed	RHEL yes; Rocky/Alma no	No

No preference? Ubuntu 24.04 LTS or Rocky Linux are easy starts

Both are free, widely supported, need no subscription, and are long-term-support releases — a sensible default if your organisation has no existing standard.

Preparing the server

- **Give it its own machine.** RadiusForge installs its own FreeRADIUS and MySQL and will not share a server with an existing RADIUS or database server. A dedicated VM is perfect.
- **Size it modestly.** A small server handles a lot of RADIUS traffic. 2 vCPU and 2–4 GB RAM is a comfortable starting point for most ISPs/WISPs; scale up with subscriber count and accounting volume.
- **Networking.** The server needs:
  - **outbound internet** during install (to fetch packages), and
  - to be **reachable** on the address your routers will send RADIUS to, and from the computer you'll manage it from.
- **Fixed address.** Give the server a **static IP** (or a permanent DHCP reservation). Your routers point at this address, so it must not change.

Two network interfaces? Make sure both come back after a reboot

If the server has more than one interface, confirm each is set to connect automatically on boot. The installer enables autoconnect for the interfaces that are up while it runs — verify with the reboot test below.

HTTPS — required before real use

The installer serves plain HTTP so you can log in immediately, but the hotspot **customer portal refuses plain HTTP**, and you should never send staff logins in the clear. Set up a certificate with the bundled script, which covers three cases:

Public server with a real domain — Let's Encrypt (recommended):

```
sudo bash scripts/https-setup.sh letsencrypt \
    --name radius.example.com --email you@example.com --agree-tos
```

This obtains and installs a free, trusted certificate and switches the site to HTTPS. It renews automatically.

A lab or internal test box — self-signed:

```
sudo bash scripts/https-setup.sh self-signed --name radius.lab
```

Browsers will warn (expected for self-signed) — fine for a lab, not for customers.

Your own certificate authority — install a cert you were issued:

```
sudo bash scripts/https-setup.sh install --cert server.crt --chain ca.crt
```

Check what's in use and until when at any time:

```
sudo bash scripts/https-setup.sh status
```

> **Point a DNS name at the server before Let's Encrypt**
> Let's Encrypt validates a real domain name, so create a DNS record (e.g. `radius.example.com`) pointing at the server's public address first. The dashboard's secure-address card warns you well before a certificate is due to expire.

Post-install checks worth doing

Confirm all services are active

```
# Red Hat family:
systemctl is-active radiusforge radiusd nginx mysqld
# Debian family:
systemctl is-active radiusforge freeradius nginx mysql
```

The reboot test (do not skip)

Every service is set to start on boot — but that's only worth anything once you've watched it happen. Reboot and check nothing needed a manual start:

```
sudo reboot
# after a minute, log back in and run:
ip -br addr
systemctl is-active radiusforge nginx    # plus radiusd/mysqld or freeradius/mysql
```

Both interfaces should have their addresses and every service should say `active` with no intervention.

> **Why this matters**
> Two classes of problem only appear on the first reboot: FreeRADIUS starting before the database is ready, and network interfaces not reconnecting. The installer guards against both — the reboot test is how you confirm it on *your* server.

Everyday commands

```
systemctl status radiusforge     # the web application
systemctl restart radiusforge    # after a change
journalctl -u radiusforge -f     # live application logs

# RADIUS service: radiusd (Red Hat) or freeradius (Debian)
systemctl status freeradius
```

FreeRADIUS keeps its own log at `/var/log/radius/radius.log` (Red Hat) or `/var/log/freeradius/radius.log` (Debian).

SELinux note (Red Hat family only)

RHEL, Rocky and Alma ship SELinux in enforcing mode, which adds permission layers beyond normal file permissions. The installer handles the cases RadiusForge needs. If you're troubleshooting an unusual setup and see a "Permission denied" that makes no sense from file permissions alone, run:

```
sudo ausearch -m avc -ts recent
```

It shows exactly what SELinux blocked and why. (Installing under `/opt/radiusforge`, as the installer ensures, avoids the most common case.)

Backups

Your important state is the **MySQL database** (subscribers, plans, sessions, vouchers, staff) and your **configuration**. Take a regular database dump — for example a nightly `mysqldump` of the RadiusForge database to off-server storage — so you can restore after hardware loss. Keep your **licence file** safe too.

> **Upgrades back themselves up**
>
> The upgrade process (below) takes its own backup of the database and configuration before it changes anything — but that's for upgrade safety, not a substitute for your own regular off-server backups.

Upgrading

When a newer release is available, upgrading is one safe command. Download and verify the new release as in the Install Guide, then:

```
sha256sum radiusforge-<new-version>.tar.gz   # compare to the website
tar xzf radiusforge-<new-version>.tar.gz
sudo bash radiusforge-<new-version>/scripts/upgrade.sh
```

The upgrade:

- checks every file against the release manifest before touching anything,
- **backs up** your database and configuration,
- carries your settings across and applies any database changes,
- verifies the result, and **rolls back by itself** if anything isn't right.

Your subscribers keep authenticating throughout.

> **Upgrade during a quiet window the first time**
>
> Although the upgrade is designed to be safe and self-rolling-back, run your first one during a low-traffic window and confirm the reboot test afterwards — standard caution for any production change.

Need a hand planning a production rollout or migration? **info@radiusforge.co.uk**