Client Area

The self-service control panel your customers get inside WHMCS: a live dashboard plus pages for web settings, FTP, databases, DNS, mailboxes, SSL, cron, backups and logs. Every page is permission-gated by the product's Client tab, and every action runs through the same asynchronous task queue (so the customer sees live "queued → creating → active" status).

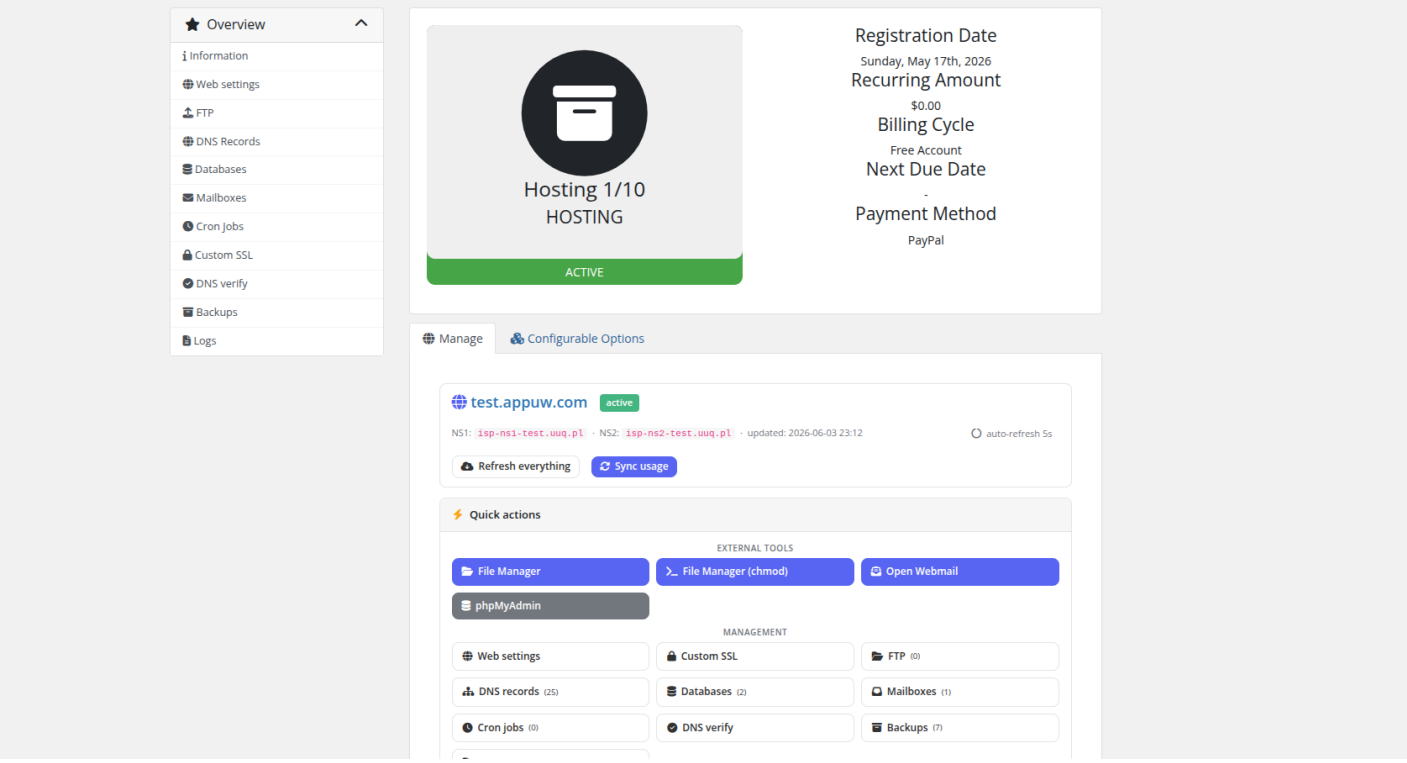
- Client Dashboard
- Web Settings
- FTP Accounts
- Databases
- DNS Records
- Mailboxes
- SSL
- Backups, Cron & Logs

Client Dashboard

PUQ Web Hosting module **WHMCS**

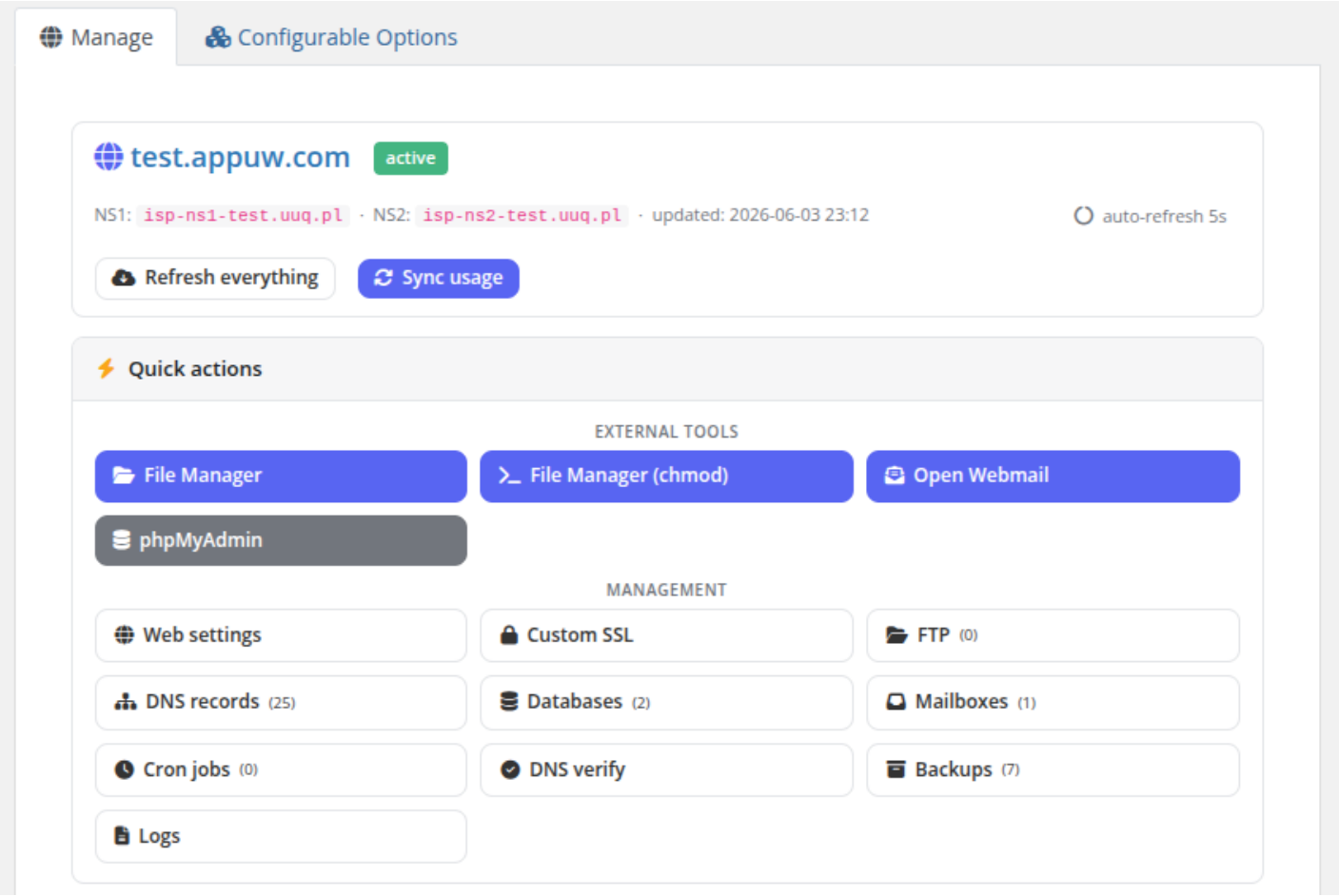
Order now | Download | Community

When a customer opens an active hosting service they land on a two-column **Web** | **Mail** dashboard that summarises everything at a glance.

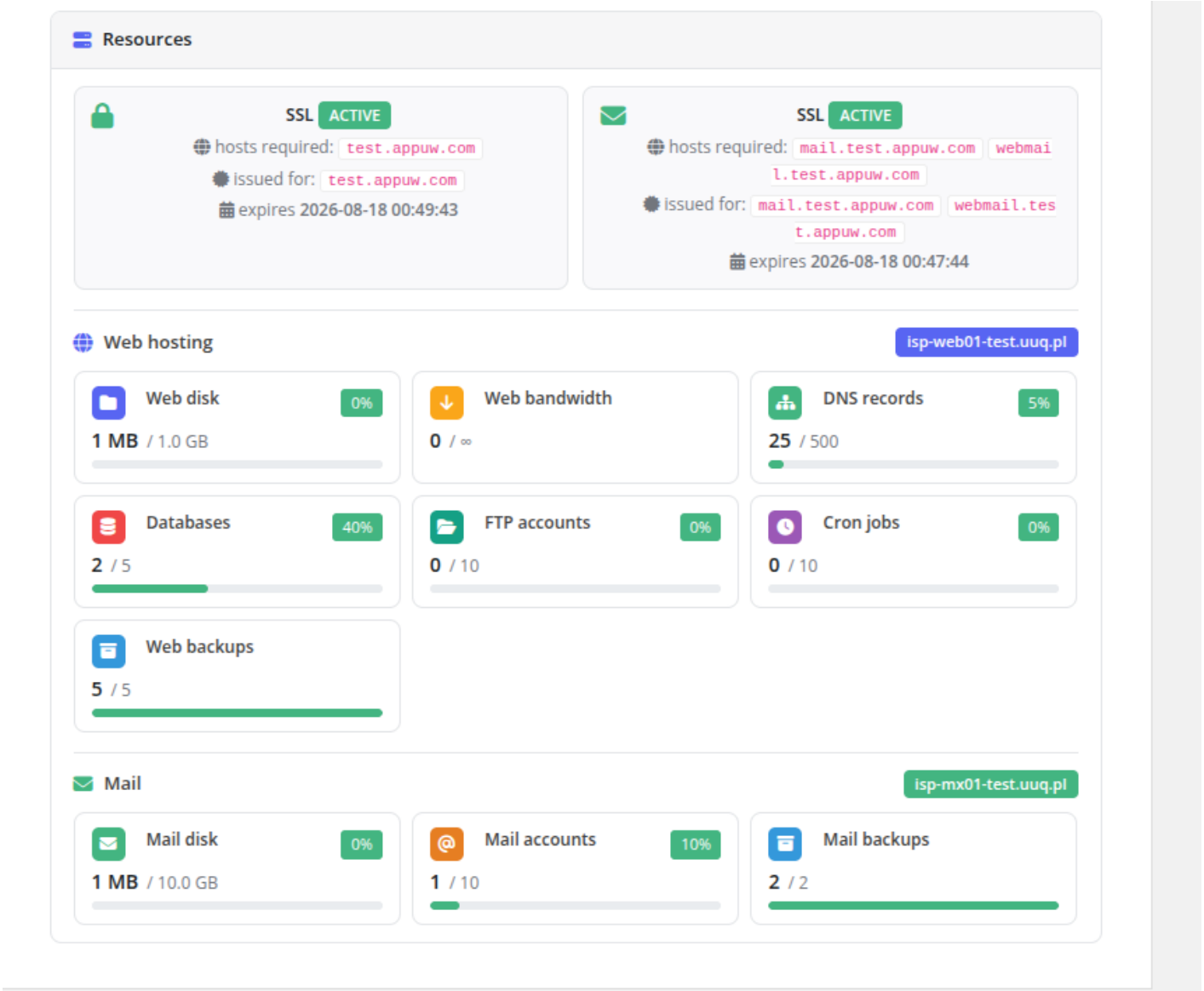


Layout

- **Web card** — website URL, web server, PHP version, SSL state, and live usage bars (disk, bandwidth, databases, FTP accounts…).
- **Mail card** — webmail link, mail server, SSL state, and mailbox/quota usage bars.
- **Quick actions** — one-click shortcuts to the panel, webmail, and the most-used resource pages.

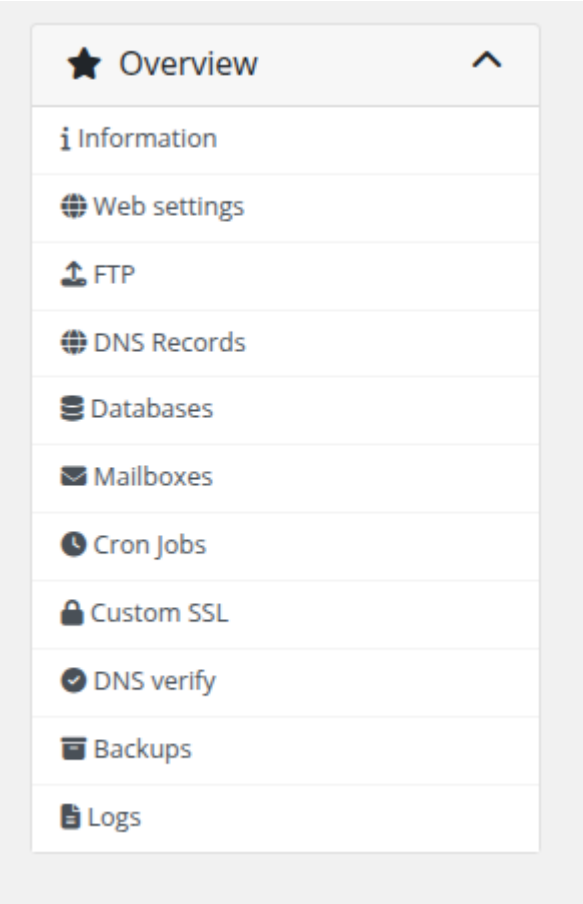


The usage bars are read from the local database (kept fresh by the usage-sync cron); **Sync now** enqueues an on-demand recompute if the customer wants the latest numbers immediately.



Navigation

A sidebar lists every enabled feature — Web settings, FTP, Databases, DNS, Mailboxes, SSL, Backups, Cron, Logs. Which entries appear depends on the **product's Client permissions** (a Mail-only product hides web pages, a Vanity product shows the simplified Website/Email set, etc.).



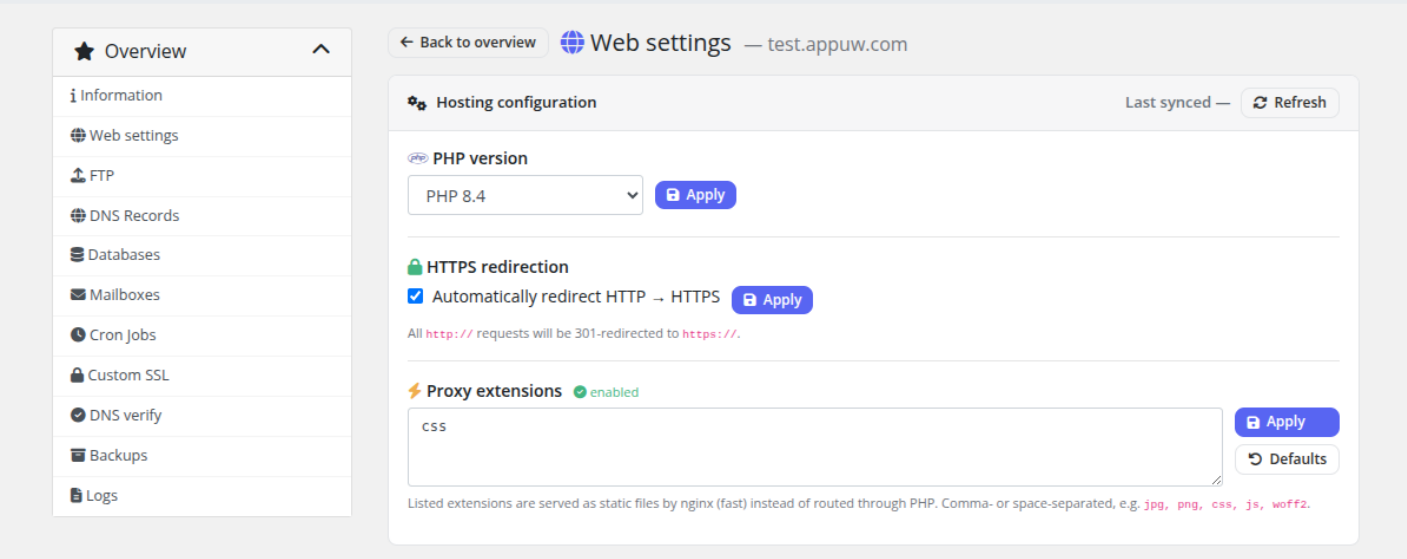
> During deployment the customer sees a live progress splash instead of the dashboard; on failure, an error screen with a **Try again** button (if the product allows redeploy). See **Deployment & Segmentation → How deployment works**.

Web Settings

PUQ Web Hosting module **WHMCS**

Order now | Download | Community

The **Web settings** page lets the customer self-serve the three most common website tweaks without a ticket.



Section	What it does
PHP version	Pick any PHP version installed on the web server (e.g. 7.4 → 8.3). Applies the matching Hestia web template.
HTTPS redirect	Force `http → https`. Only selectable once SSL is active for the domain.
Proxy extensions	Choose which static file extensions the front proxy serves directly (with **Defaults** and **Disable** shortcuts).

Each **Apply** is asynchronous: the request is queued, a spinner badge shows while the worker runs, and an error tooltip surfaces if Hestia rejects the change. The page mirrors the live state back from the server, so what you see is what's actually applied.

> “

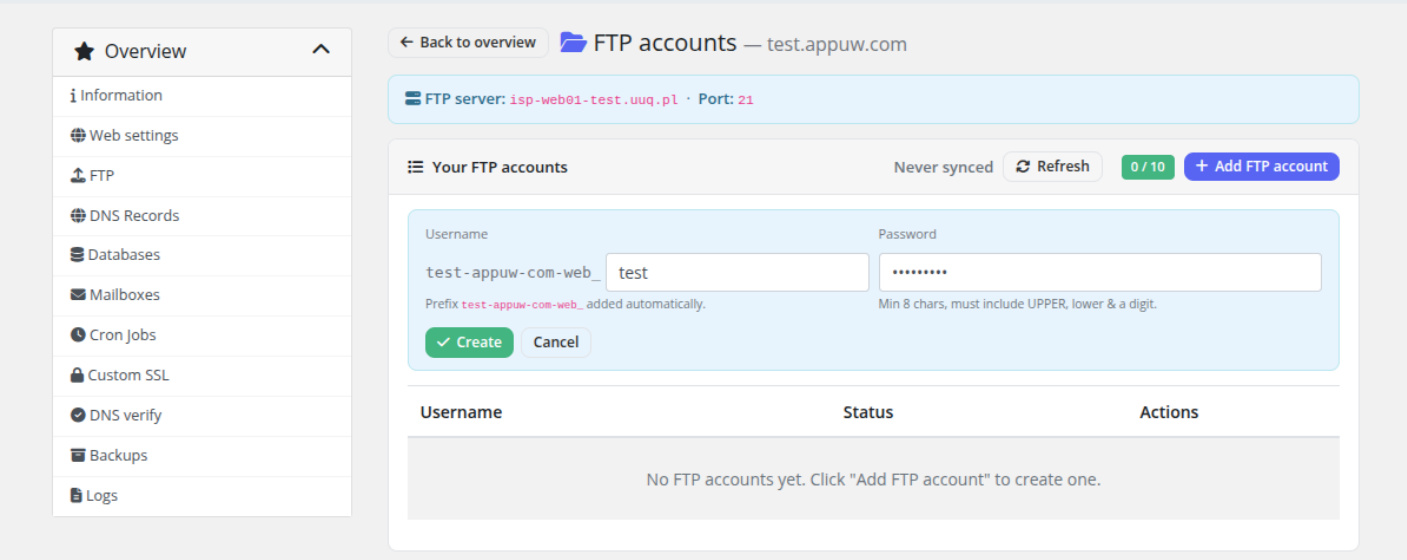
> These controls appear only if the product's Client permissions enable hosting settings. On a Vanity service this page is simplified to the relevant subset.

FTP Accounts

PUQ Web Hosting module **WHMCS**

Order now | Download | Community

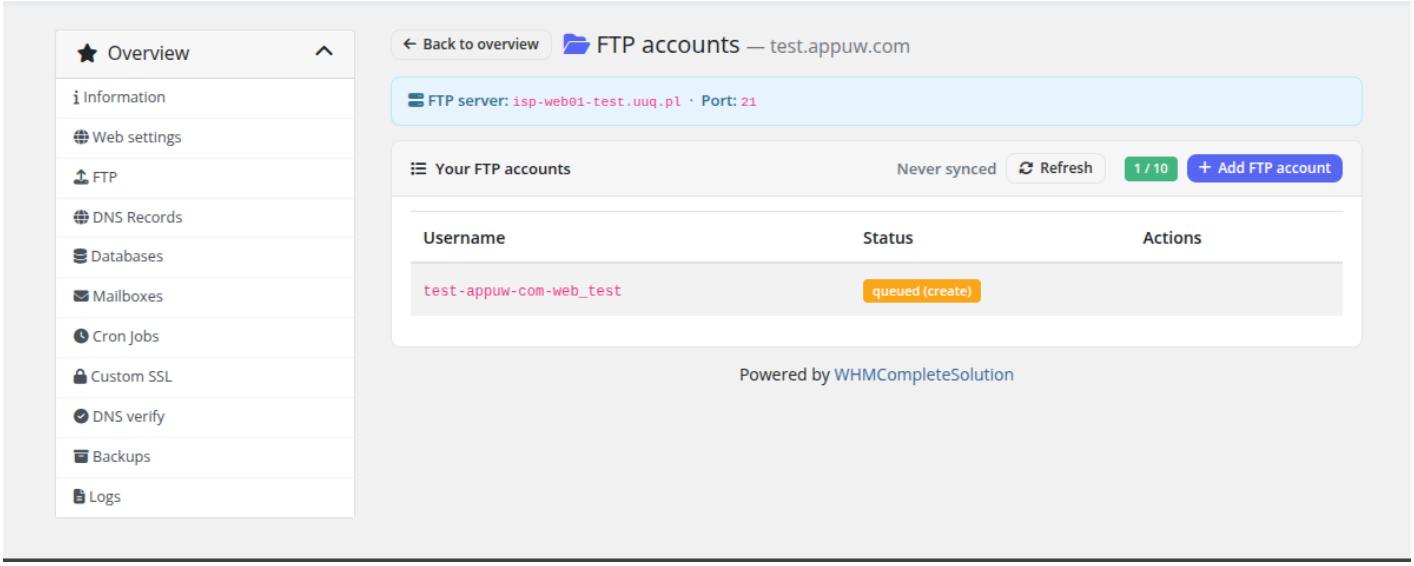
Customers manage their own FTP accounts here, up to the product's FTP-account limit.



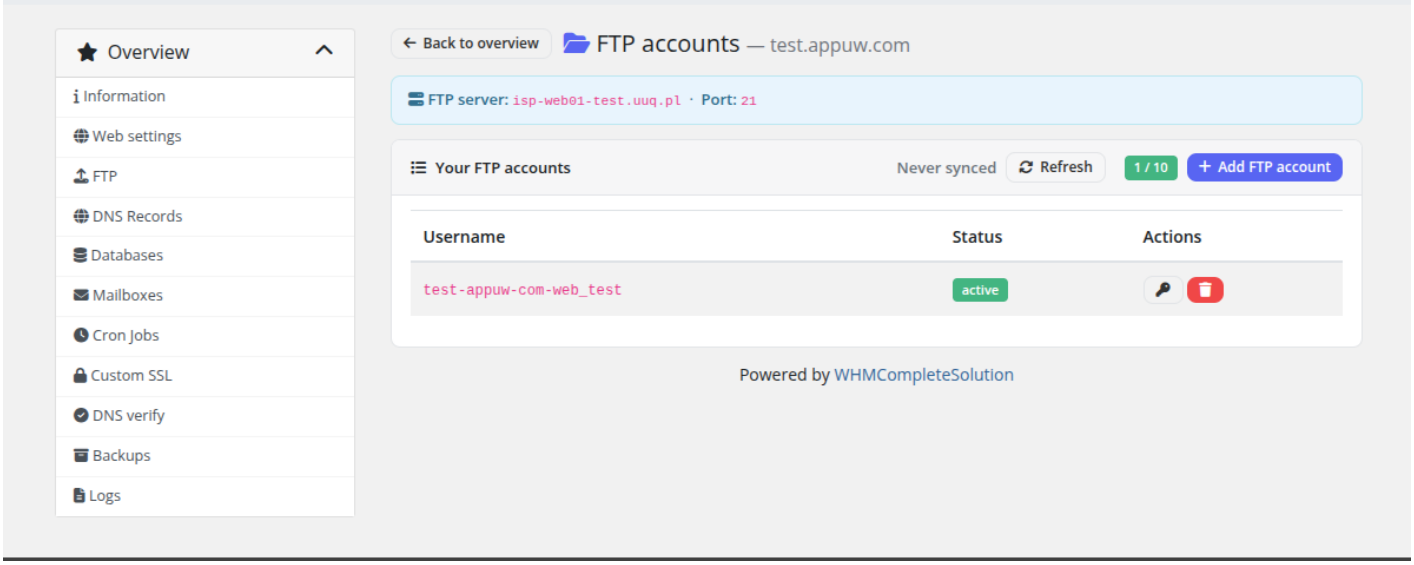
Creating an account

Enter a suffix and a password (or generate one) and submit. The real Hestia username is the main web user plus your suffix (`<webuser>_<suffix>`) — the page shows the **full** username so the customer knows exactly what to type into their FTP client. The home directory and shell are set by the **Provisioning** defaults (typically `nologin`, so FTP/SFTP only).

The action is queued, so the new row appears with a **Queued** badge first:



...and flips to **Active** once the worker has created it on the server:



> The suffix length is budget-aware: the module reserves enough characters so the combined username stays within Hestia's limit. If a suffix would overflow, it's rejected before submission.

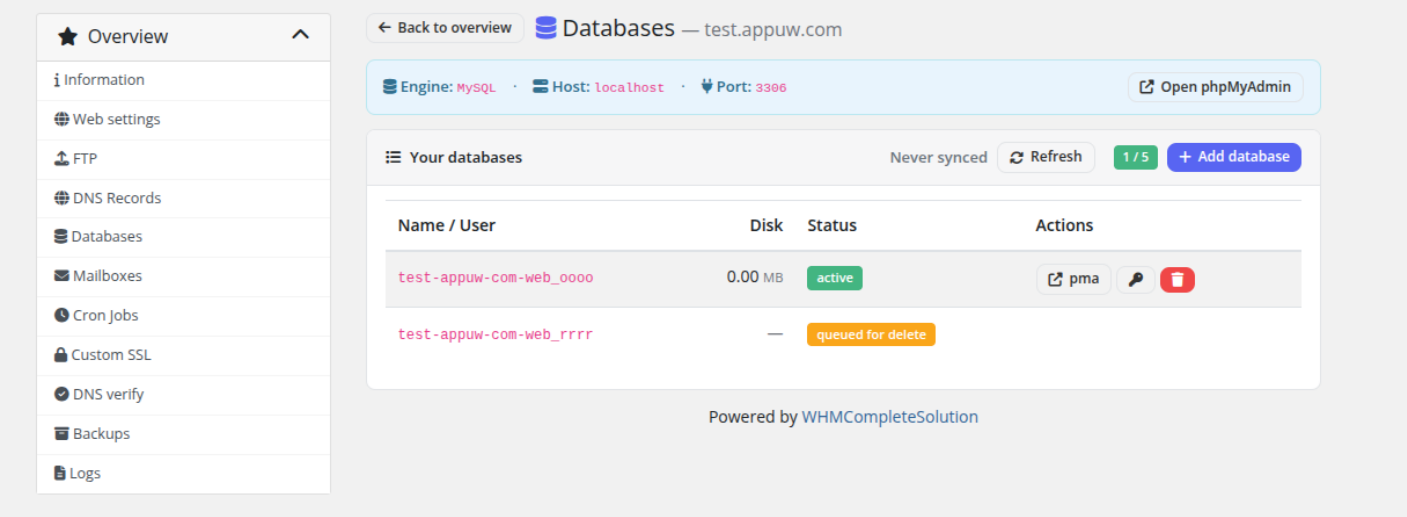
Open the file manager (FileGator / net2ftp) from the dashboard quick actions for in-browser file access without a desktop FTP client.

Databases

PUQ Web Hosting module **WHMCS**

Order now | Download | Community

The **Databases** page lets customers create and manage MySQL/MariaDB databases up to the product's database limit.



Each database is created with its own dedicated user. As with FTP, the database and user names are prefixed with the main web user (`<webuser>_<suffix>`), and the page shows the **full** names plus the live size of each database.

- **Create** — pick a suffix and password; the database and matching user are provisioned together (queued → active).
- **Delete** — removes the database and its user.
- **phpMyAdmin** — opens the configured phpMyAdmin URL for in-browser administration.

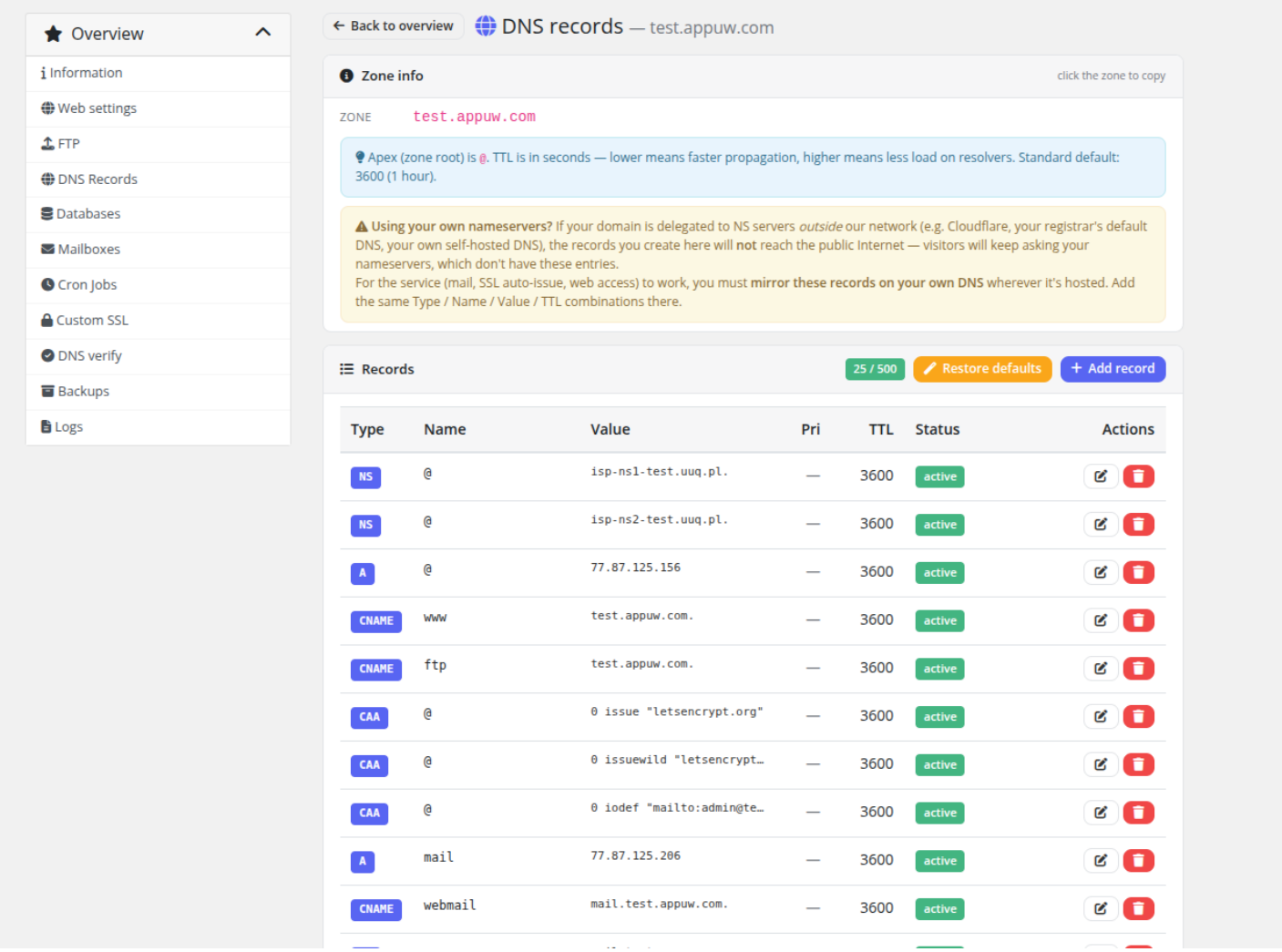
> Database size is read by the usage-sync cron and counts toward nothing by itself, but the **number** of databases is capped by the product limit / Configurable Option.

DNS Records

PUQ Web Hosting module **WHMCS**

Order now | Download | Community

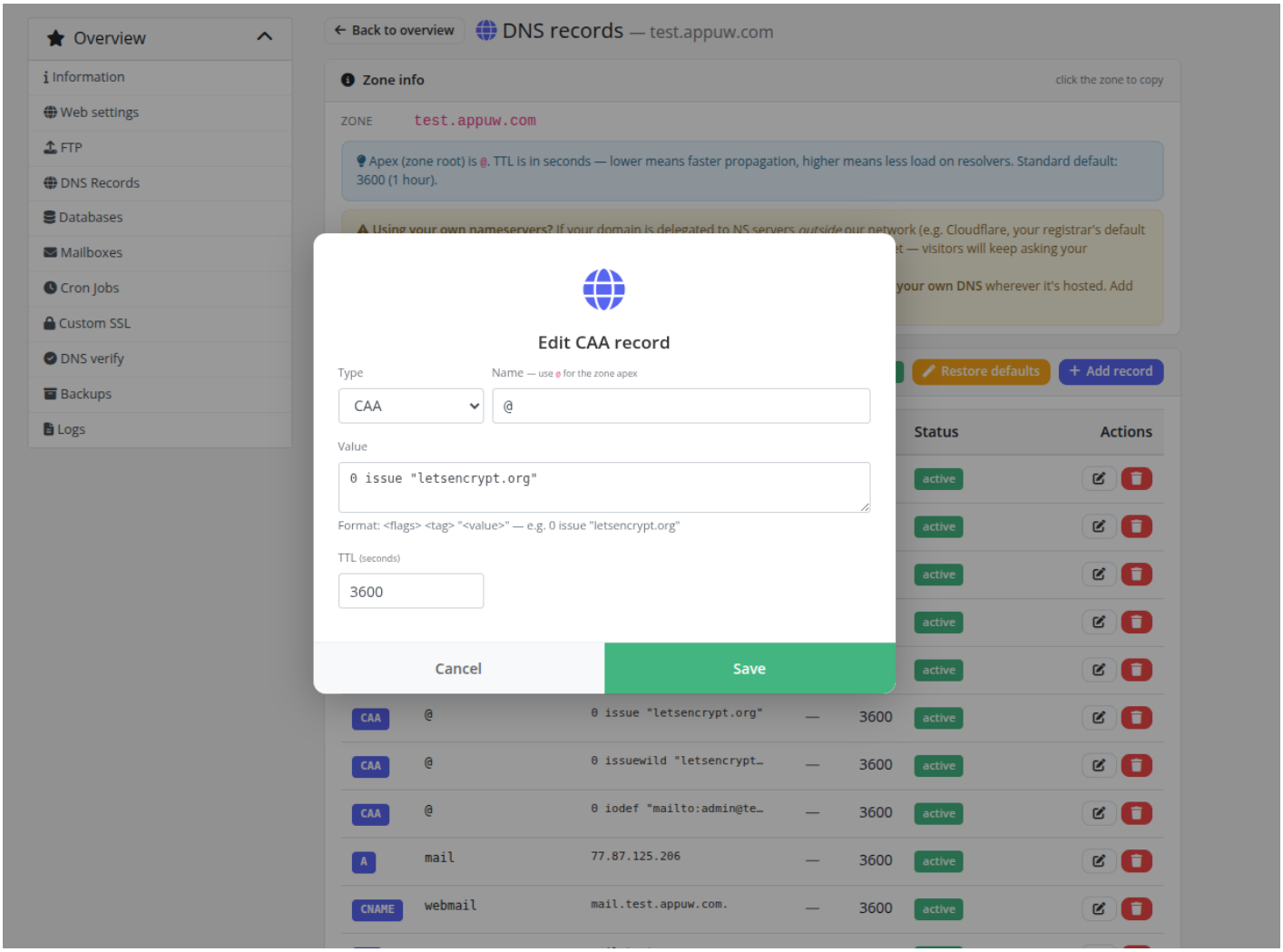
If the product includes the DNS role, customers manage their zone's records here, up to the product's record limit.



Editing records

Add, edit or delete A / AAAA / CNAME / MX / TXT / SRV records. The editor validates the type/value

and shows the TTL within the configured min/max bounds.



Every change is fanned out to **all** DNS servers attached to the service's group (the active-active cluster), so records stay identical on every nameserver. See **Deployment & Segmentation → Server segmentation** for how the DNS pool works.

External DNS

If the service was set up with **external DNS** (the customer keeps DNS elsewhere), this page instead shows the records they need to publish at their own provider, plus a **Verify** check that probes the live nameservers and confirms the records resolve:

← Back to overview

DNS verify — test.appuw.com

DNS records to publish

This is the canonical record set for `test.appuw.com`. Publish these on your registrar's DNS panel, then click **Verify all** to confirm propagation.

Our DNS — point your domain to the nameservers below; we publish the records.

Nameservers (only required if we manage DNS)

- `isp-ns1-test.uuq.pl`
- `isp-ns2-test.uuq.pl`

Verify all records

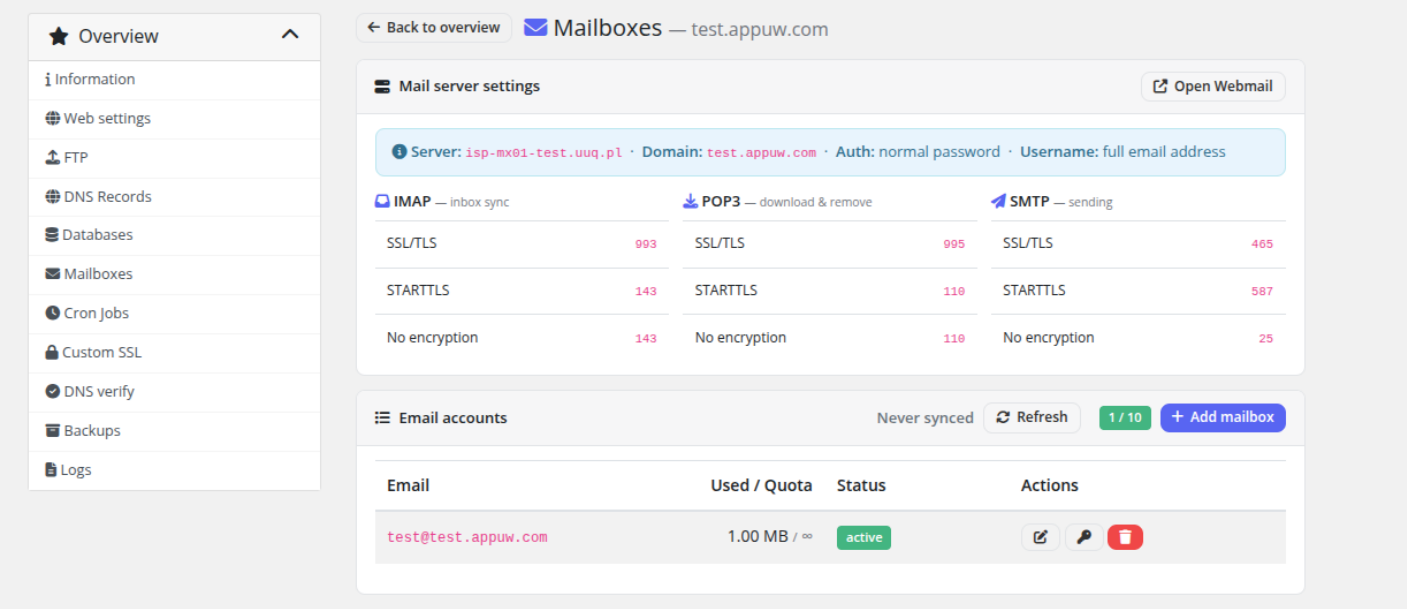
Type	Name	Value (expected)	Priority / TTL	Live
NS	@	isp-ns1-test.uuq.pl.	TTL:3600	(not checked)
NS	@	isp-ns2-test.uuq.pl.	TTL:3600	(not checked)
A	@	77.87.125.156	TTL:3600	(not checked)
CNAME	www	test.appuw.com.	TTL:3600	(not checked)
CNAME	ftp	test.appuw.com.	TTL:3600	(not checked)
CAA	@	0 issue "letsencrypt.org"	TTL:3600	(not checked)
CAA	@	0 issuewild "letsencrypt.org"	TTL:3600	(not checked)
CAA	@	0 iodef "mailto:admin@test.appuw.com"	TTL:3600	(not

Mailboxes

PUQ Web Hosting module **WHMCS**

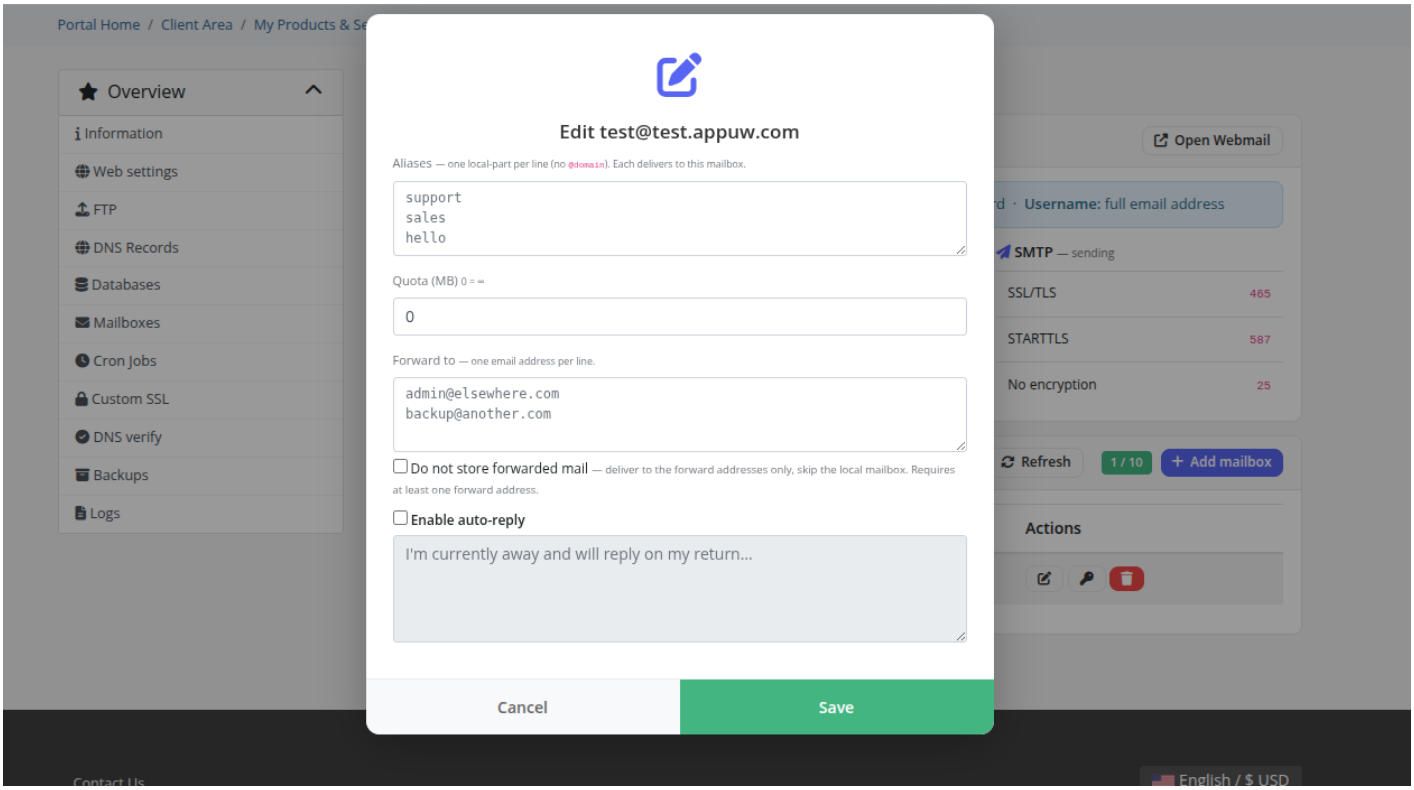
Order now | Download | Community

On a service with the Mail role, customers manage their mailboxes here, up to the product's mailbox limit.



Per-mailbox settings

Create a mailbox with an address and password, then open it to manage:



- **Password** — reset the mailbox password.
- **Quota** — per-mailbox storage limit (or unlimited), within the service's mail quota.
- **Forwards** — forward incoming mail to one or more addresses.
- **Auto-reply** — vacation / out-of-office responder.

Webmail opens the configured webmail URL for the mail server.

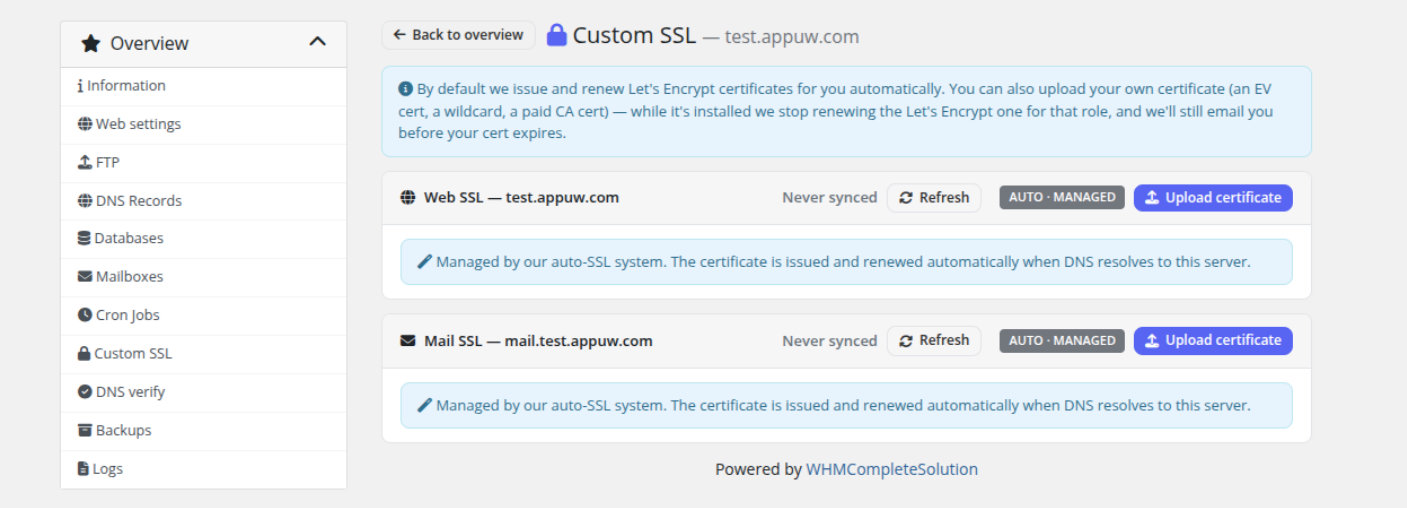
> On a **Vanity** service this is the single `name@<your-domain>` mailbox, presented as a simplified Email page — see **Vanity Mode → Order & client experience**.

SSL

PUQ Web Hosting module **WHMCS**

Order now | Download | Community

The **SSL** page shows certificate status and lets customers upload a custom certificate.

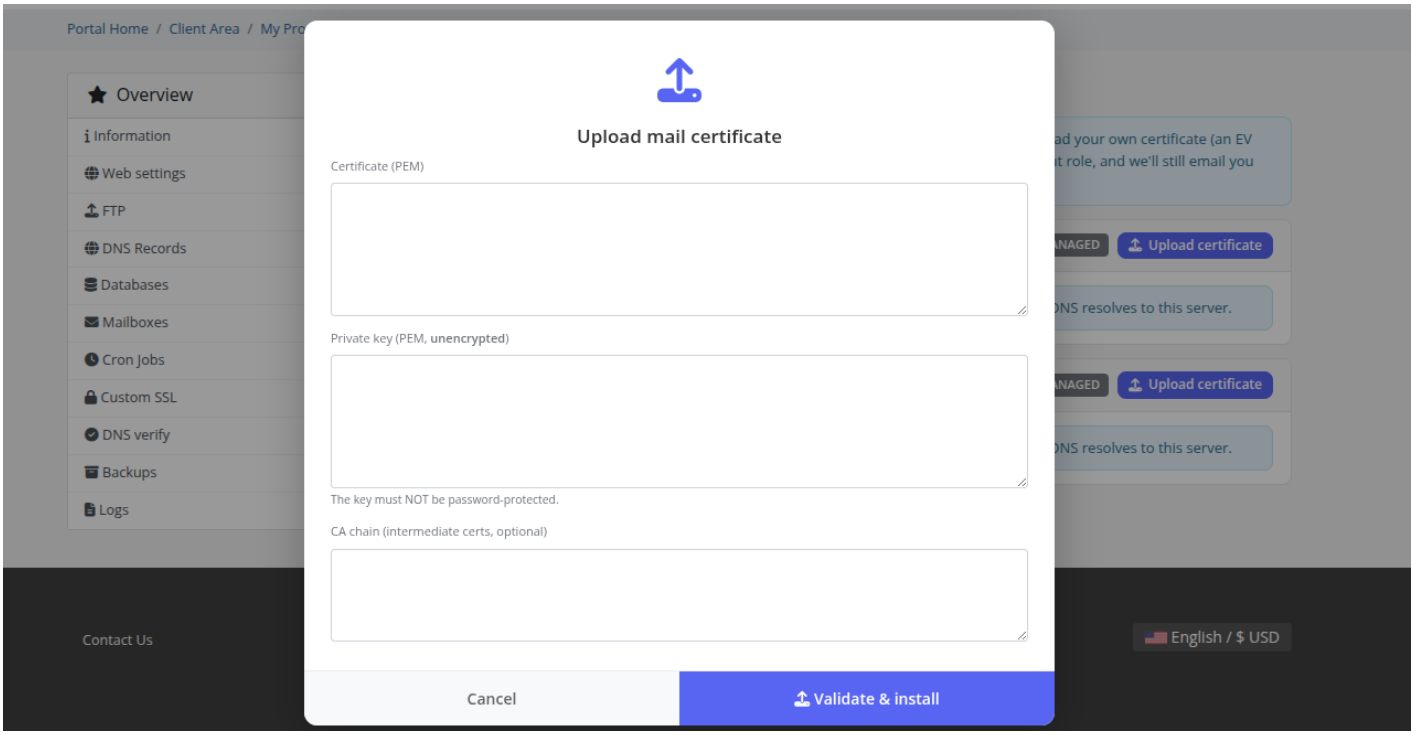


Automatic SSL

In the common case there's nothing to do: the module issues and renews **Let's Encrypt** certificates automatically for the website (and for `mail.` / `webmail.` on the mail role). The page shows the current state and expiry; the cadence and rate-limit guards are described in **Cron & Automation → SSL automation**.

Custom certificate

Customers who already have a certificate (e.g. a wildcard or EV cert) can upload it here — certificate, private key and optional CA bundle:



Uploading a custom certificate installs it on the domain and **suspends auto-SSL** for that role so it won't be overwritten. A daily cron checks custom-cert expiry and notifies before it lapses.

Backups, Cron & Logs

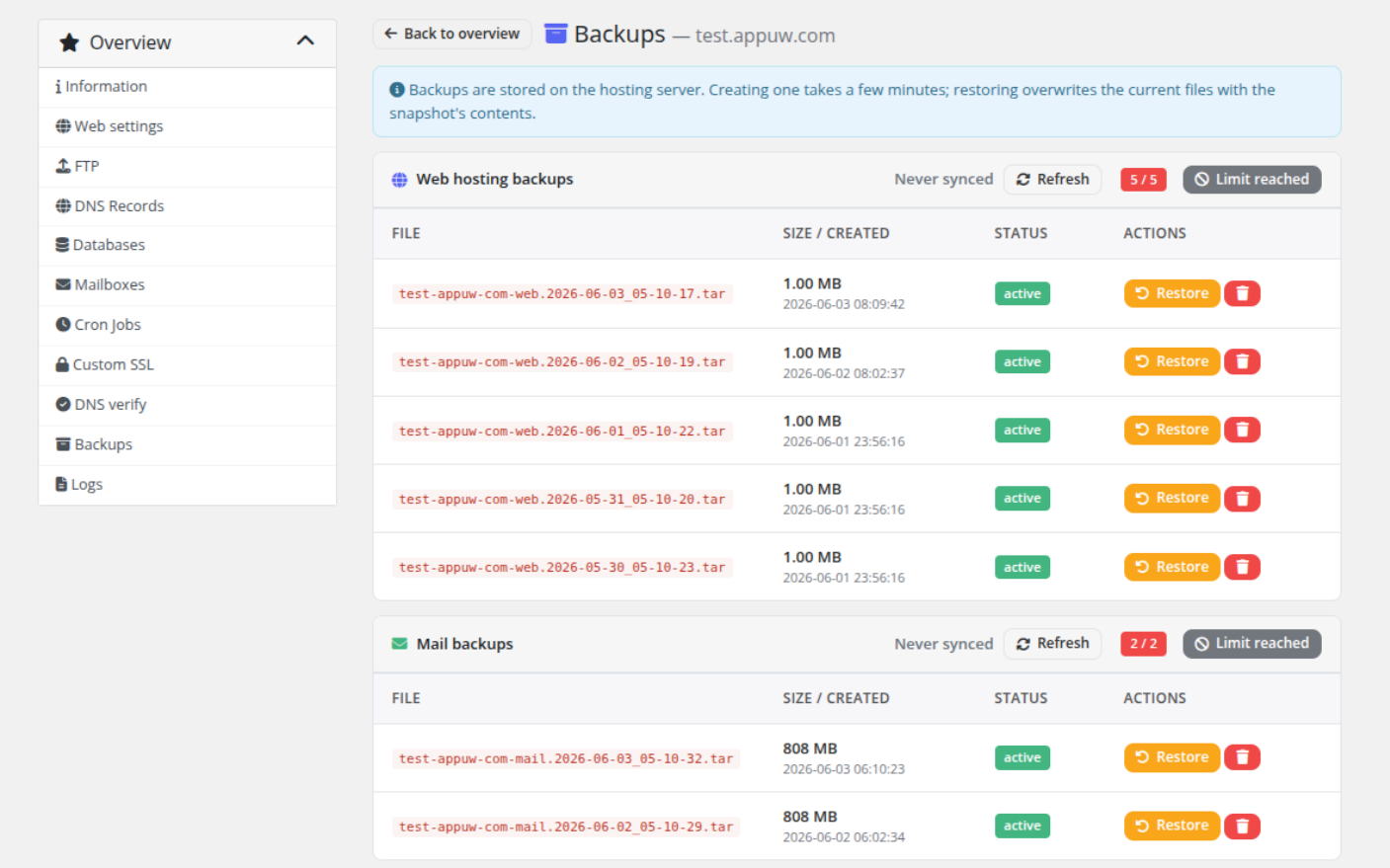
PUQ Web Hosting module **WHMCS**

Order now | Download | Community

Three more self-service pages round out the client area.

Backups

Customers can take and restore their own snapshots, up to the product's web/mail backup quota.



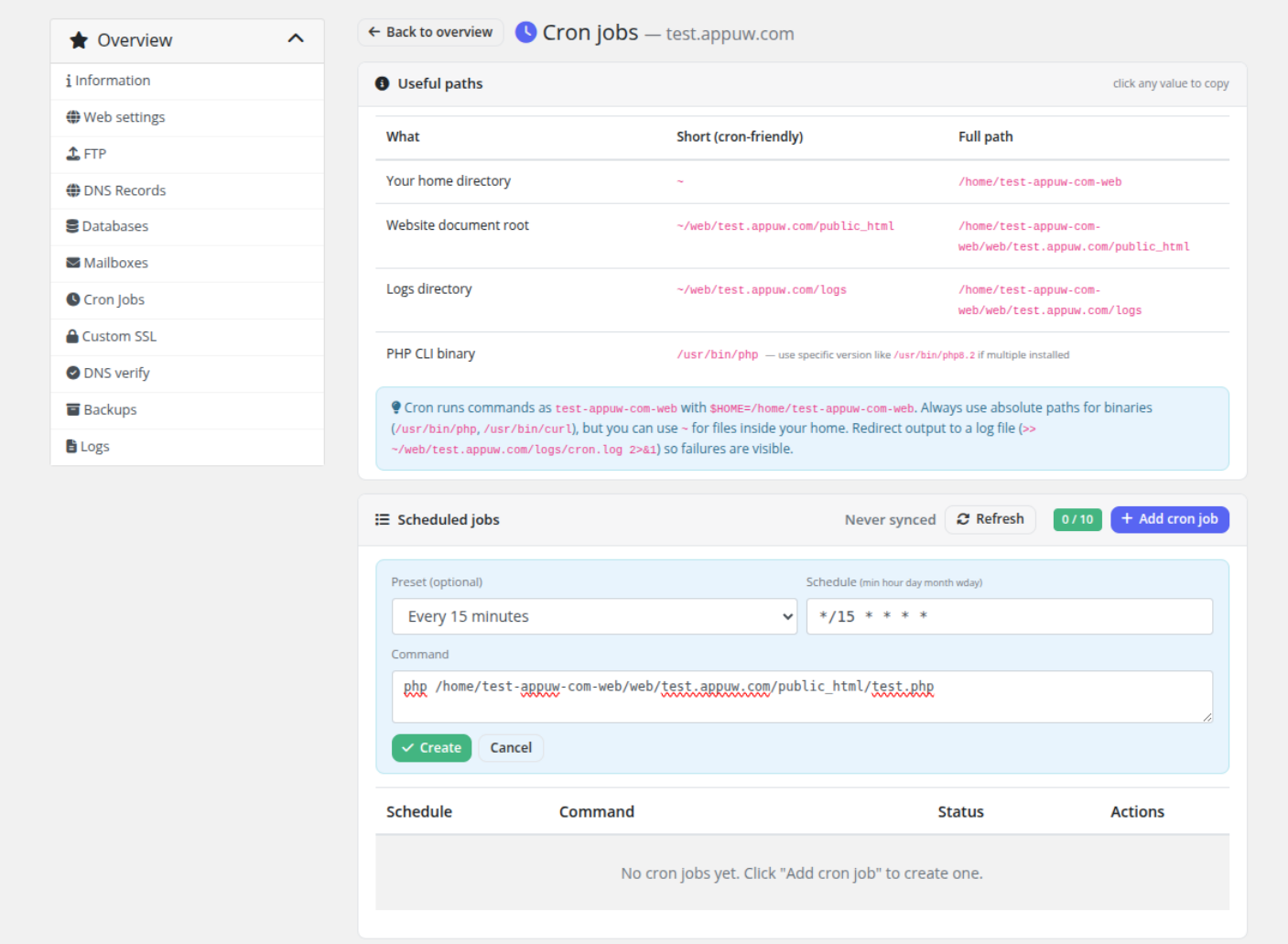
- **Create** — schedules a snapshot on the server (asynchronous; the row shows a *creating* badge until the file appears).
- **Restore** — restores a snapshot; a typed **RESTORE** confirmation is required, and write actions on the service are locked while the restore runs.

- **Delete** — removes a snapshot.

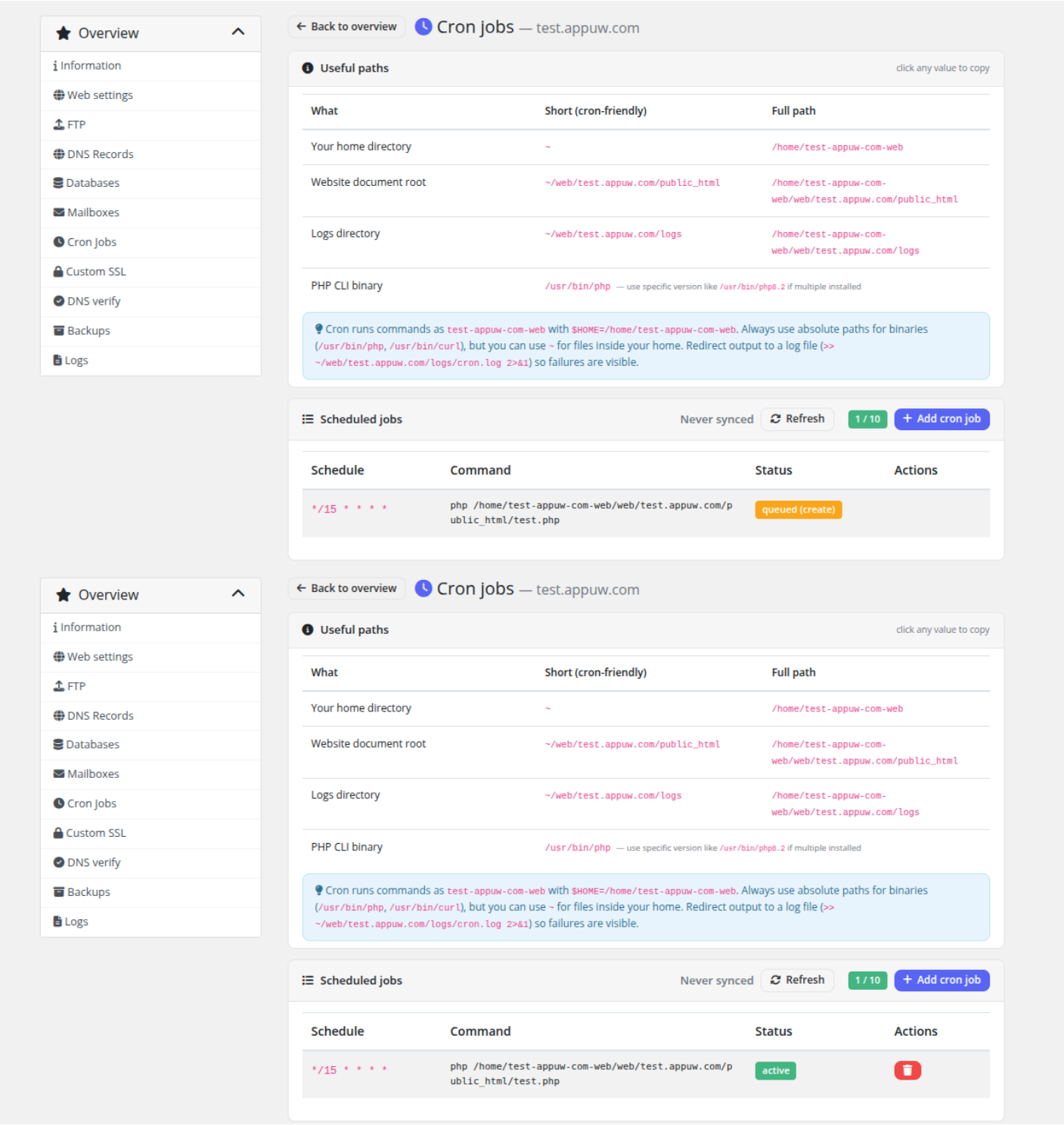
Web and mail snapshots are listed and quota-counted separately.

Cron

Customers manage scheduled jobs for their website, up to the cron-job limit.

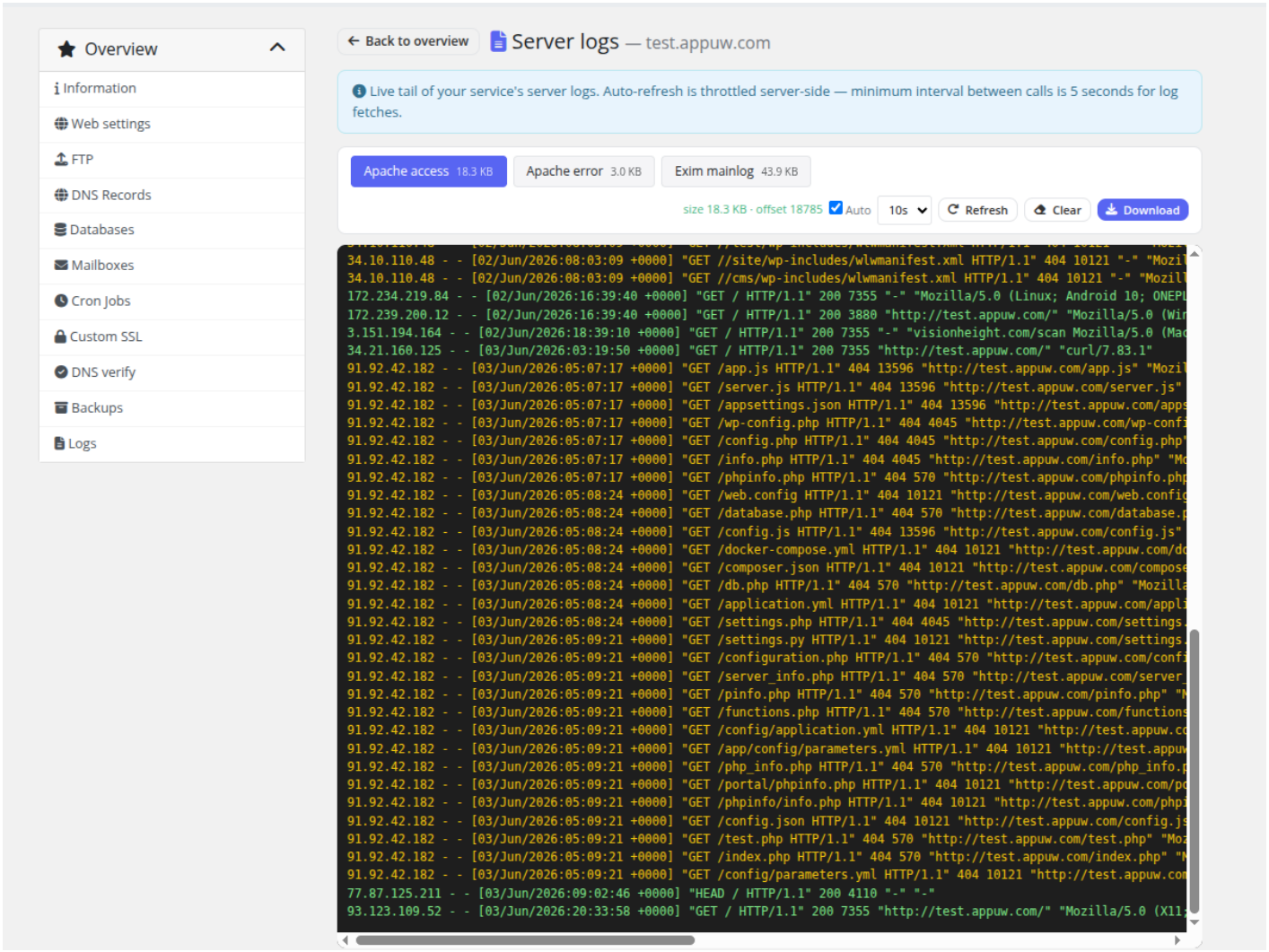


Add a job with a schedule and command; it's queued (**Queued** badge) and goes **Active** once installed on the server:



Logs

The **Logs** page tails the website's real server logs (Apache/Exim/Dovecot as applicable), grep-filtered to the customer's own domain, with a size-capped download.



> Every one of these pages is gated by the product's **Client permissions**, so you can expose exactly the feature set each plan should have.