

Installation & Configuration

Get the module running end-to-end: install both parts, build your server infrastructure, group your nodes, and create your first product. Start here, then read [Deployment & Segmentation](#) to plan how your fleet scales.

- [Requirements & Installation](#)
- [Add Web / Mail / DNS Servers](#)
- [Server Groups](#)
- [Create a Product](#)

Requirements & Installation

PUQ Web Hosting module **WHMCS**

[Order now](#) | [Download](#) | [Community](#)

Requirements

- A working **WHMCS** installation (see the PHP/WHMCS compatibility table below).
- **ionCube Loader** enabled on the WHMCS host, matching the PHP version it runs.
- One or more **HestiaCP** servers reachable over **SSH** from the WHMCS host. The module drives Hestia entirely over SSH (the panel HTTP API is not required); set up `sudo` NOPASSWD for the management user, or use the Hestia `admin` user with key/password auth.
- (Optional) **PowerDNS** servers if you want PowerDNS as a DNS backend alongside HestiaCP.
- Outbound HTTPS from the Hestia nodes for Let's Encrypt and for the optional file-manager / PHP installers.

PHP & WHMCS compatibility

The module is distributed as **ionCube-encoded builds, one per PHP version: 7.4, 8.1 and 8.2**. Always install the build that matches the **PHP version your WHMCS host runs under** — not the PHP versions you offer to hosting customers (those are independent and managed on each Hestia web server).

WHMCS version	PHP it runs on	Module build to install
WHMCS 8.x	7.4, 8.1 or 8.2	the 7.4 / 8.1 / 8.2 build matching your PHP
WHMCS 9.x	8.2	the 8.2 build
any host on PHP 8.3 / 8.4 or newer	8.3+	the 8.2 build

How to choose, in plain terms:

1. Check the PHP version your WHMCS server runs (WHMCS → *Utilities* → *System* → *PHP Info*, or `|php -v|` on the host).
2. On **WHMCS 8** you may run PHP **7.4, 8.1 or 8.2** — download the build with the same number.
3. On **WHMCS 9** the host runs PHP **8.2** — use the **8.2** build.
4. If your host runs a PHP version **newer than 8.2** (8.3, 8.4, ...), use the **8.2 build** — it is the latest supported and is compatible with newer PHP runtimes.

Download

The module is distributed as **ionCube-encoded ZIP builds — one per supported PHP major version (7.4, 8.1, 8.2)**. Pick the build that matches the PHP version your **WHMCS host** runs (see the compatibility table above), then download it.

All versions and historical builds are in the index:

- https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-WEB-Hosting/

Direct "latest" downloads

PHP 8.2

```
wget https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-WEB-Hosting/php82/PUQ_WHMCS-WEB-Hosting-latest.zip
```

PHP 8.1

```
wget https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-WEB-Hosting/php81/PUQ_WHMCS-WEB-Hosting-latest.zip
```

PHP 7.4

```
wget https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-WEB-Hosting/php74/PUQ_WHMCS-WEB-Hosting-latest.zip
```

“ Not sure which PHP version your WHMCS runs on? Check **Utilities** → **System** → **PHP Info** in the WHMCS admin area. On hosts running PHP **8.3 / 8.4 or newer**,

use the **8.2** build (it is the latest supported and runs on newer PHP).

Unzip the archive

On your WHMCS server (or locally, before uploading):

```
unzip PUQ_WHMCS-WEB-Hosting-latest.zip
```

The archive extracts into a `PUQ_WHMCS-WEB-Hosting/` directory containing the two module folders: `puqWebHosting` (server module) and `puq_web_hosting` (addon module).

“ After any module update, re-run the schema check (below).

Install both modules

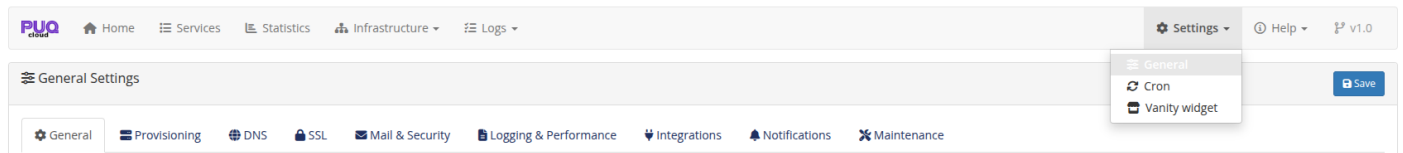
The product is **two** WHMCS modules — install both:

Upload to	Module
<code>modules/servers/puqWebHosting/</code>	the server module
<code>modules/addons/puq_web_hosting/</code>	the addon module

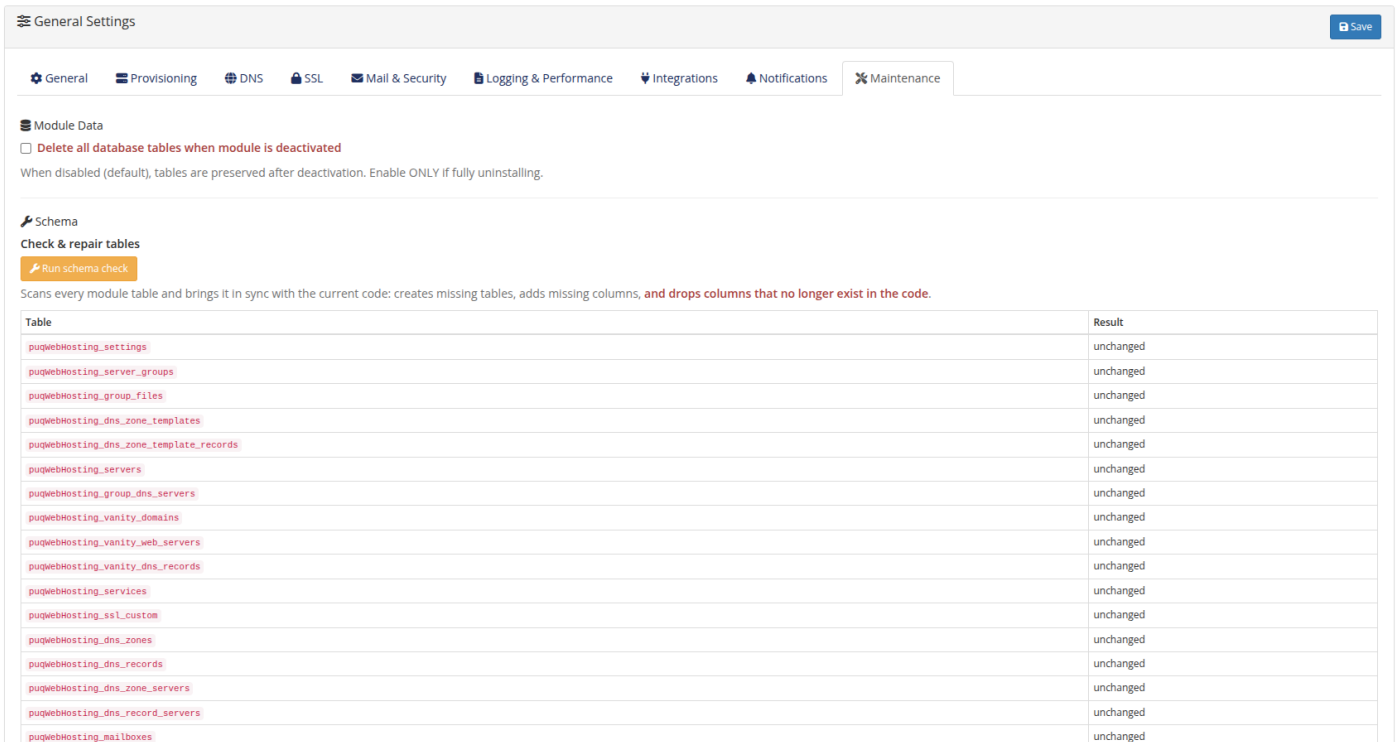
Then in WHMCS go to **Setup → Addon Modules**, find **PUQ Web Hosting**, and **Activate** it (grant access to the admin roles that should manage hosting). Activating the addon creates the database tables the server module needs.

First run: check the schema & find the panels

After activation you have a new **PUQ Web Hosting** admin page (via *Addons*). Its top navigation gives you everything: **Home, Services, Statistics, Infrastructure, Logs, Settings, Help**.



Open **Settings → Maintenance** and click **Run schema check** once. It scans every module table and brings it in sync with the current code — creating missing tables, adding missing columns and dropping columns that no longer exist. You should run this after every module update.



“ The same tab has a “*delete all tables on deactivate*” toggle — leave it **off** unless you are fully uninstalling, so your data survives an accidental deactivation.

What's next

With both modules installed you build your service from the bottom up:

1. **Add your servers** and tag their capabilities — see *Add web / mail / DNS servers*.
2. **Group them** and apply role-targeted configuration — see *Server groups*.
3. **Create a product** pointed at a group — see *Create a product*.

The module **Settings** (timeouts, retry policy, SSL cadence, DNS/SOA defaults, integrations, notifications) are covered in *Addon Module → Settings*; the defaults are sensible, so you can come

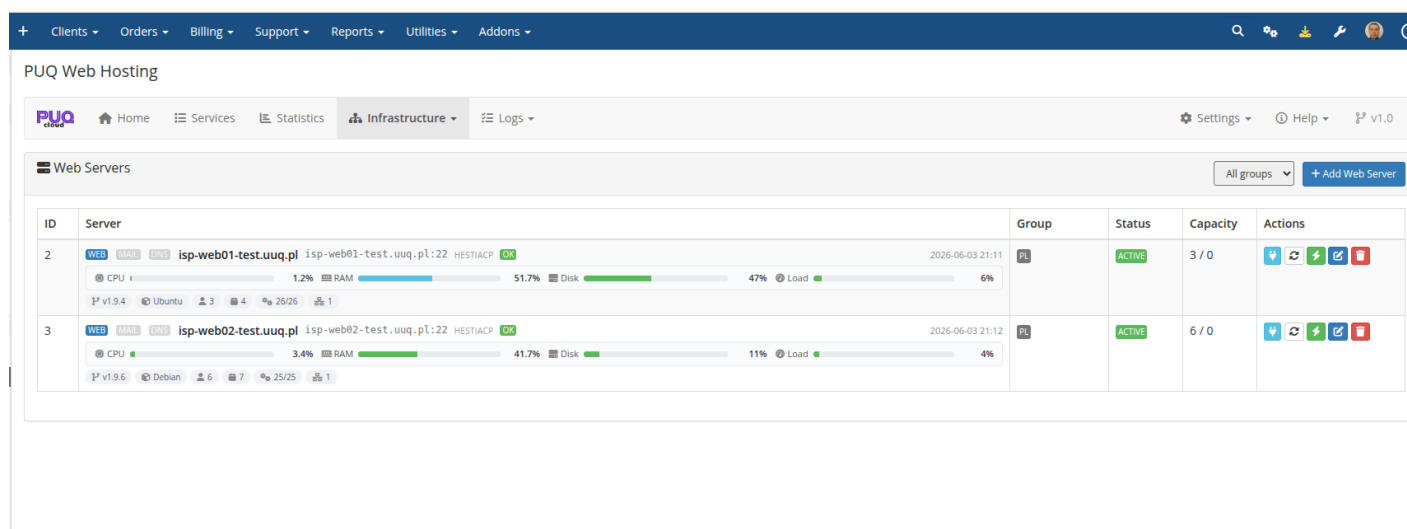
back to them later.

Add Web / Mail / DNS Servers

PUQ Web Hosting module **WHMCS**

[Order now](#) | [Download](#) | [Community](#)

Your fleet is managed under **Infrastructure**. The same physical servers appear under **Web Servers**, **Mail Servers** and **DNS Servers** filtered by the capabilities you give them.



The screenshot shows the PUQ Web Hosting interface. At the top, there's a navigation bar with links for Clients, Orders, Billing, Support, Reports, Utilities, and Addons. Below this is a sub-header for 'PUQ Web Hosting' with a navigation menu including Home, Services, Statistics, Infrastructure (selected), and Logs. The main content area is titled 'Web Servers' and features a '+ Add Web Server' button. A table lists two servers, each with a detailed status bar showing CPU, RAM, Disk, and Load usage, along with a green 'OK' indicator. The table columns are ID, Server, Group, Status, Capacity, and Actions.

ID	Server	Group	Status	Capacity	Actions
2	isp-web01-test.uuq.pl isp-web01-test.uuq.pl:22 HESTIACP OK CPU 1.2% RAM 51.7% Disk 47% Load 6% v1.9.4 Ubuntu 3 4 26/26 1	PL	ACTIVE	3 / 0	+ ↺ ↻ ✉ 🔒
3	isp-web02-test.uuq.pl isp-web02-test.uuq.pl:22 HESTIACP OK CPU 3.4% RAM 41.7% Disk 11% Load 4% v1.9.6 Debian 6 7 25/25 1	PL	ACTIVE	6 / 0	+ ↺ ↻ ✉ 🔒

Each row shows live **CPU / RAM / Disk / Load**, the Hestia version & OS, the panel-OK indicator, the group, the capacity used/max and the row actions. A green **OK** means the SSH/panel probe succeeded.

Add a server

Click **Add Web Server** (or Mail/DNS — they open the same editor) and fill in the connection details:

SSH password

leave blank to keep existing

Requirements on the HestiaCP server:

1. SSH access enabled on port **22** (or custom port above).
2. SSH user can log in via password or private key.
3. **FULL passwordless sudo is required.** The module runs both Hestia **v-*** commands and system probes (**df**, **/proc/***, custom scripts) — partial NOPASSWD scoped only to **/usr/local/hestia/bin/*** is not enough.

Two ways to satisfy this:

- (a) SSH as **root** — easiest, no sudoers config needed. The module auto-detects **uid=0** and skips **sudo** entirely.
- (b) Use a sudoer user — create **/etc/sudoers.d/puq-webhosting** on the Hestia server:

```
sudo tee /etc/sudoers.d/puq-webhosting <<'EOF'
# Full passwordless sudo for PUQ Web Hosting integration.
# Replace "admin" with the SSH username configured above.
admin ALL=(ALL) NOPASSWD: ALL
EOF
sudo chmod 440 /etc/sudoers.d/puq-webhosting
sudo visudo -c # syntax check - must say "parsed OK"
```

Verify it works: **sudo -n -l** on the server must list a **NOPASSWD: ALL** rule.

4. For **private-key** auth — public key in **~/.ssh/authorized_keys** of the SSH user, perms **600**.
5. Recommended: firewall SSH so only the WHMCS server IP can connect.

The Test connection button below runs **sudo -n -l** as a precheck — connection only succeeds if a **NOPASSWD: ALL** rule is in effect (or SSH is as root).

DNS zone template:

Capacity max:

Driver-specific config

SSH command timeout (seconds):

How long a single v-* invocation may run before SSH times out.

Default Hestia package for new users:

Must already exist on the Hestia panel (v-list-user-packages).

- **DNS zone template** — the template used for zones created on/for this node.
- **Capacity max** — soft capacity used for least-loaded placement.
- **SSH command timeout** — override for slow nodes.
- **Default Hestia package** for new users.

Use **Test connection** before saving; the row will then show **OK** and start reporting live stats.

Capabilities = your topology

How you tick capabilities **is** your segmentation plan:

- **One node, all roles** → tick Web + Mail + DNS on a single server (great for starting out).
- **Web/mail split** → some nodes tick **Web** only, dedicated nodes tick **Mail** only.
- **Three tiers** → separate Web, Mail and **DNS** (nameserver) pools.

+
Clients
Orders
Billing
Support
Reports
Utilities
Addons

Search
Settings
Help
v1.0

PUQ Web Hosting

PUQ
Home
Services
Statistics
Infrastructure
Logs

Settings
Help
v1.0

Mail Servers

All groups
+ Add Mail Server

ID	Server	Group	Status	Capacity	Actions
8	MAIL isp-mx01-test.uuq.pl isp-mx01-test.uuq.pl:22 HESTIACP 2026-06-03 21:12 CPU 3.4% RAM 29.2% Disk 22% Load 2% v1.9.6 Debian 4 5 16/16 1	PE	ACTIVE	4 / 0	Down Arrow Refresh Lightning Bolt Envelope Stop
9	MAIL isp-mx02-test.uuq.pl isp-mx02-test.uuq.pl:22 HESTIACP 2026-06-03 21:12 CPU 2.3% RAM 49.8% Disk 10% Load 5% v1.9.6 Debian 2 2 16/16 1	PE	ACTIVE	2 / 0	Down Arrow Refresh Lightning Bolt Envelope Stop

+
Clients
Orders
Billing
Support
Reports
Utilities
Addons

v1.0

PUQ Web Hosting

PUQ
Home
Services
Statistics
Infrastructure
Logs

Settings
Help
v1.0

DNS Servers

All drivers
+ Add DNS Server

DNS servers are independent — attach them to groups from Group ... DNS servers tab. One DNS server can serve many groups.

ID	Server	Status	Actions
10	WEB MAIL DNS HEALTHY isp-ns1-test.uuq.pl isp-ns1-test.uuq.pl:22 OK 2026-06-03 21:12 CPU 6.7% RAM 37.3% Disk 10% Load 8% P v1.9.6	ACTIVE	Refresh Restart Stop Start Delete
11	WEB MAIL DNS HEALTHY isp-ns2-test.uuq.pl isp-ns2-test.uuq.pl:22 OK 2026-06-03 21:12 CPU 6.8% RAM 36.7% Disk 10% Load 12% P v1.9.6	ACTIVE	Refresh Restart Stop Start Delete

Server Groups

PUQ Web Hosting module WHMCS

[Order now](#) | [Download](#) | [Community](#)

A **server group** bundles a set of nodes together. A WHMCS product points at one group, and the module places each service's roles on the group's capable nodes. Groups are managed under **Infrastructure → Server Groups**.

+

Clients

Orders

Billing

Support

Reports

Utilities

Addons

Search

Settings

Help

v1.0

PUQ Web Hosting

Home

Services

Statistics

Infrastructure

Logs

Server Groups

+ Add Server Group

Click a row's Open button to manage the group: actions / Hestia config / standard files. = managed Hestia config keys.

ID	Name	Description	Managed	Actions
1	pl		Ht	OpenApply
2	Vanity		Ht---	OpenApply

Click **Open** on a row to manage the group. The **H:** badge shows how many `hestia.conf` keys the group manages.

Group purpose: Standard vs Vanity

The first decision, on the group's **General** tab, is **Group purpose**:

The screenshot shows the PUQ Web Hosting interface. At the top, there's a navigation bar with links for Home, Services, Statistics, Infrastructure, and Logs. A 'Settings' dropdown and 'Help' link are on the right, along with a version indicator 'v1.0'. Below this, a breadcrumb trail shows 'Back' and 'pl' with a status '4 server(s) — 4 online'. A 'Save group' button is in the top right. The main content area has tabs for Actions, General (selected), All-servers config, Web config, Mail config, DNS config, Variables, Standard files, and DNS servers. The 'General' tab contains a 'Name' field with 'pl', a 'Description' text area, and a 'Group purpose' section. The 'Group purpose' section has two radio buttons: 'Standard hosting' (selected) and 'Vanity'. Below these, a note explains the difference between the two modes.

← Back pl 4 server(s) — 4 online Save group

⚡ Actions ⓘ General ⚙️ All-servers config 🌐 Web config ✉️ Mail config 🏗️ DNS config 📁 Variables 📄 Standard files 🖨️ DNS servers

Name *

pl

Description

Group purpose

☒ Standard hosting — web / mail / dns servers + DNS cluster ☐ Vanity — sell name.<domain> / name@<domain> on domains you own

Choosing Vanity hides the hosting-config tabs and shows the *Vanity domains* tab; web servers are picked per vanity domain there. Switching purpose takes effect after you click **Save group**.

- **Standard hosting** — web / mail / DNS servers + a DNS cluster. Used by Split and Unified products.
- **Vanity** — sell `name.<domain>` / `name@<domain>` on domains you own. Choosing this hides the hosting-config tabs and shows the **Vanity domains** tab. (See the dedicated **Vanity Mode** chapter.)

Attach DNS servers (the cluster)

On the group's **DNS servers** tab, tick the nameservers this group replicates zones to. DNS servers are independent and shared — one server can back many groups. Every zone for the group is pushed to **every** attached DNS node (active-active).

Home
Services
Statistics
Infrastructure
Logs
Settings
Help
v1.0

Back
pl
4 server(s) — 4 online
Save group

Actions
General
All-servers config
Web config
Mail config
DNS config
Variables
Standard files
DNS servers

Default DNS zone template — used for new services in this group unless the web server has its own override

Resolution chain: `web_server.dns_template_id` → `group.dns_template_id` (this) → `global_is_default`. Leave empty to fall through to the global default.

Default (global default)
Save default template

SOA defaults for new zones — substituted into `puqwebhosting.dns_zones` on first deploy

These values run through the same placeholder substitution as zone template records. Available variables: `{domain}` `{server_ip}` `{mail_server_ip}` `{mail_server_hostname}` `{ns1}` `{ns2}` `{client_id}` `{service_id}` `{year}` `{unixtime}` + any custom key you define in the Variables tab.

SOA email (use `.` instead of `@`)
TTL (s)
Refresh (s)
Retry (s)

admin.{domain}
3600
3600
3600

Save SOA defaults

Expire (s)
Minimum (s)

3600
3600

Attach DNS servers to this group. DNS servers are independent — one server can serve many groups. Zones and records of services in this group will be replicated to every attached DNS server (active-active cluster).

Available DNS servers — 2 of 2 selected
Save DNS selection

No server in this group has Web AND Mail capabilities ticked — unified-deployment products will fail to provision here. Edit a server under Web servers and tick both caps, or stick with Split-deployment products.

Name	Driver	Caps	Hostname	Status	Capacity
☒ isp-ns1-test.uuq.pl	HESTIACP	WEB MAIL DNS	isp-ns1-test.uuq.pl	ACTIVE	4
☒ isp-ns2-test.uuq.pl	HESTIACP	WEB MAIL DNS	isp-ns2-test.uuq.pl	ACTIVE	4

Don't see your DNS server? Add it on the DNS Servers page first.

This tab is also where you set the group's **default DNS zone template** and **SOA defaults** for new zones.

Apply configuration

The group's **Actions** tab lets you push the managed Hestia configuration and standard files to the members, per role (all / web-only / mail-only / DNS-only). The detail of role-targeted config (`hestia.conf` management, drift, standard files) is covered in **Addon Module → Server-group editor** and **Deployment & Segmentation → Server segmentation**.

“ A minimal setup is **one group** containing your nodes. As you grow, use multiple groups to separate locations/tiers or to offer different SLAs, and a separate **Vanity** group for your name-selling offers.

Create a Product

PUQ Web Hosting module **WHMCS**

[Order now](#) | [Download](#) | [Community](#)

Create a product in WHMCS (*Setup* → *Products/Services*), then on **Module Settings** choose **PUQ Web Hosting** and point **Server Group** at the group you built.

DetailsPricingModule SettingsCustom FieldsConfigurable OptionsUpgradesFree DomainCross-sellsOtherLinks

Module NamePUQ Web HostingServer GroupNone

License key

success: License check disabled (DEV)

GeneralWeb limitsMail limitsDNS limitsClientEmailConfig options

Infrastructure

Server Group (addon side) *

pl

Picks which addon server group (web + mail + dns) this product deploys into. Nameservers and DNS variables are configured per-group under *Addon* → *Server Groups* → *Edit* → *DNS variables*.

Package contents

What this package includes

☒ Web (domain, files, databases, FTP, cron) ☒ Mail (mailboxes, webmail, mail backups) ☐ DNS (always active — locally managed)

Deployment mode

☒ Split (one Hestia user per role; web / mail / dns can live on different servers) ☐ Unified (single Hestia user holds web + mail; DNS joins the same account when the server is in the DNS cluster)

☐ Vanity (sell name.<your-domain> sites + name@<your-domain> mailboxes on a domain YOU own)

Split is the classic mode and supports any group composition. Unified requires the chosen server group to contain at least one server with both *Web* and *Mail* capabilities ticked (*Addon* → *Servers* → *Edit* → *Capabilities*). Customer-facing change: a single *Account backups* quota replaces the separate *Web* / *Mail* backup counts. The mode is locked at *CreateAccount* time — switching the product of an existing service between split and unified is refused.

Vanity sells slots on a domain you own (configured under *Addon* → *Server Groups* → *Edit* → *Vanity domains*). The buyer picks the parent domain via a Configurable Option named *vanity_domain* and enters their name in a Custom Field named *vanity_name*; the service then gets a normal per-service web account for *name.<domain>* and ONE mailbox *name@<domain>* on your shared mail user. The module never creates or deletes the parent domain / zone / your provider user. Set the per-order site and mailbox sizes on the *Vanity limits* tab.

How role gating works: A role is *enabled* for a service when BOTH the package-level checkbox is ticked AND the corresponding *disk_quota* resolves to a non-zero value. Setting *web_disk_quota=0* (or *mail_disk_quota=0*) via the Web/Mail tab or via a WHMCS Configurable Option behaves exactly like un-ticking the box here — the role is skipped entirely:

- Provisioner skips *createPackage* / *createuser* / *addDomain* for the disabled role.
- Client area hides the matching tabs (Mailboxes / Databases / FTP / Cron / Web settings / Custom SSL / Backups) and dashboard cards.
- Admin inline panel skips role-specific pills.
- DNS is never gated — DNS records are managed locally regardless of which Hestia roles are active. If the server group has no DNS servers attached, DNS provisioning is silently skipped.

At least one of *Web* / *Mail* must be ticked — provisioning errors otherwise.

The configuration lives on a row of sub-tabs: **General** • **Web limits** • **Mail limits** • **DNS limits** • **Client** • **Email** • **Config options** (in Vanity mode the limits tabs collapse to a single **Vanity limits** tab).

General — roles & deployment mode

On **General** you pick which **roles** the package includes (tick **Web** / **Mail** / **DNS**) and the **Deployment mode**

(Split / Unified / Vanity — see Deployment & Segmentation → Deployment models).

DetailsPricingModule SettingsCustom FieldsConfigurable OptionsUpgradesFree DomainCross-sellsOtherLinks

Module NamePUQ Web Hosting

Server GroupNone

License key

success: License check disabled (DEV)

GeneralVanity limitsClientEmailConfig options

Infrastructure

Server Group (addon side) *

Vanity

Picks which addon server group (web + mail + dns) this product deploys into. Nameservers and DNS variables are configured per-group under Addon → Server Groups → Edit → DNS variables.

Package contents

What this package includes

☒ Web (domain, files, databases, FTP, cron)

☒ Mail (mailboxes, webmail, mail backups)

☒ DNS (always active — locally managed)

Deployment mode

☐ Split (one Hestia user per role; web / mail / dns can live on different servers)

☐ Unified (single Hestia user holds web + mail; DNS joins the same account when the server is in the DNS cluster)

☒ Vanity (sell name-<your-domain> sites + name@<your-domain> mailboxes on a domain YOU own)

Split is the classic mode and supports any group composition. Unified requires the chosen server group to contain at least one server with both Web and Mail capabilities ticked (Addon → Servers → Edit → Capabilities). Customer-facing change: a single Account backups quota replaces the separate Web / Mail backup counts. The mode is locked at `_CreateAccount` time — switching the product of an existing service between split and unified is refused.

Vanity sells slots on a domain you own (configured under Addon → Server Groups → Edit → Vanity domains). The buyer picks the parent domain via a Configurable Option named `vanity_domain` and enters their name in a Custom Field named `vanity_name`; the service then gets a normal per-service web account for `name.<domain>` and ONE mailbox `name@<domain>` on your shared mail user. The module never creates or deletes the parent domain / zone / your provider user. Set the per-order site and mailbox sizes on the Vanity limits tab.

How role gating works: A role is *enabled* for a service when BOTH the package-level checkbox is ticked AND the corresponding `disk_quota` resolves to a non-zero value. Setting `web_disk_quota=0` (or `mail_disk_quota=0`) via the Web/Mail tab or via a WHMCS Configurable Option behaves exactly like un-ticking the box here — the role is skipped entirely:

- Provisioner skips `createPackage` / `createuser` / `addomain` for the disabled role.
- Client area hides the matching tabs (Mailboxes / Databases / FTP / Cron / Web settings / Custom SSL / Backups) and dashboard cards.
- Admin inline panel skips role-specific pills.
- DNS is never gated — DNS records are managed locally regardless of which Hestia roles are active. If the server group has no DNS servers attached, DNS provisioning is silently skipped.

At least one of Web / Mail must be ticked — provisioning errors otherwise.

Limits — what the customer gets

The **Web / Mail / DNS limits** tabs define the Hestia package quotas and the module-local caps for each ticked role. Every field maps to a **Configurable Option override key** (e.g. `web_disk_quota`) so you can offer upgradeable tiers.

- Web limits** — disk, bandwidth, cron jobs, backups kept, max subdomains (`WEB_ALIASES`), max databases, max FTP accounts, default PHP version, auto-install Let's Encrypt.

Details
Pricing
Module Settings
Custom Fields
Configurable Options
Upgrades
Free Domain
Cross-sells
Other
Links

Module Name
PUQ Web Hosting
Server Group
None

License key
success: License check disabled (DEV)

General
Web limits
Mail limits
DNS limits
Client
Email
Config options

Hestia package quotas

Disk quota (MB)
DISK_QUOTA
1024
0 = role disabled (acts like un-ticking *Web* in General). Configurable Option override: `web_disk_quota`.

Bandwidth (MB / month)
BANDWIDTH
0
Outgoing traffic. 0 = unlimited. Configurable Option override: `web_bandwidth`.

Cron jobs
CRON_JOBS
10
0 = unlimited. Configurable Option override: `web_cron_jobs`.

Backups kept
BACKUPS
5
0 = backups disabled. Configurable Option override: `web_backups`.

☐ Incremental backups
BACKUPS_INCREMENTAL

Max subdomains
WEB_ALIASES
1
Aliases / subdomains under the main web domain. Sets Hestia `WEB_ALIASES`. 0 = none.

Max databases
DATABASES
5
Configurable Option override: `web_databases`.

Module-local limits
Enforced by client-area CRUD (our DB tables) — not pushed to Hestia.

Max FTP accounts
10
Configurable Option override: `web_ftp_accounts`.

Default PHP version
8.2

☐ Auto-install Let's Encrypt on account create

- **Mail limits** — mail disk, mail accounts, outbound rate limit/hour, mail backups (independent of web), max forwarders, max aliases, autoresponders/spam-filter.

Details
Pricing
Module Settings
Custom Fields
Configurable Options
Upgrades
Free Domain
Cross-sells
Other
Links

Module Name
PUQ Web Hosting
Server Group
None

License key
success: License check disabled (DEV)

General
Web limits
Mail limits
DNS limits
Client
Email
Config options

Hestia package quotas

Disk quota (MB)
DISK_QUOTA
10240
0 = role disabled (acts like un-ticking *Mail* in General). Configurable Option override: `mail_disk_quota`.

Mail accounts
MAIL_ACCOUNTS
10
Configurable Option override: `mail_accounts`.

Rate limit (email / hour)
RATE_LIMIT
100
Outgoing emails per mailbox per hour. 0 = unlimited.

Backups kept
BACKUPS
2
0 = backups disabled. Independent from web backups. Configurable Option override: `mail_backups`.

☐ Incremental backups
BACKUPS_INCREMENTAL

Module-local limits
Enforced in our DB tables — not part of Hestia's package.

Max forwarders
10

Max aliases
10

☐ Autoresponders enabled
☐ Spam filter enabled

- **DNS limits** — max DNS records, DNS disk quota, default TTL, and the **allowed record types** the customer may use (A, AAAA, CNAME, MX, TXT, NS, SRV, CAA, PTR, DNSKEY, DS, NAPTR, TLSA — on both HestiaCP BIND and PowerDNS).

Details

Pricing

Module Settings

Custom Fields

Configurable Options

Upgrades

Free Domain

Cross-sells

Other

Links

Module Name

PUQ Web Hosting

Server Group

None

License key

success: License check disabled (DEV)

General

Web limits

Mail limits

DNS limits

Client

Email

Config options

Max DNS records

DNS_RECORDS

500

0 = unlimited. Configurable Option override: dns_records.

Disk quota (MB)

DISK_QUOTA

256

DNS user disk allowance for zone files.

Default TTL (seconds)

3600

Allowed record types (CSV)

A,AAAA,CNAME,MX,TXT,NS,SRV,CAA,PTR,DNSKEY,DS,NAPTR,TLSA

Use all 13

Comma-separated subset of supported types. Customers see ONLY the types listed here in the client-area record editor.

Copy-paste full list: A,AAAA,CNAME,MX,TXT,NS,SRV,CAA,PTR,DNSKEY,DS,NAPTR,TLSA

Supported by both DNS backends (HestiaCP BIND + PowerDNS): A (IPv4) AAAA (IPv6) CNAME alias NS nameserver MX mail server TXT free-form text (SPF/DKIM/DMARC/verification) SRV service location CAA certificate-authority authorization PTR reverse-DNS pointer DNSKEY DNSSEC public key DS DNSSEC delegation signer NAPTR naming-authority pointer (ENUM/SIP) TLSA TLS certificate association (DANE).

“Setting a role's disk quota to `0` disables that role for the product (the same as un-ticking it on General).

Client — what the customer can do

The **Client** tab is a set of toggles controlling which client-area pages/actions are visible — create/delete mailbox, change password, create database/FTP, edit DNS, install SSL, manage cron, web settings, backups, view logs, client-side redeploy, file-manager & webmail SSO. Untick to hide.

Details

Pricing

Module Settings

Custom Fields

Configurable Options

Upgrades

Free Domain

Cross-sells

Other

Links

Module Name

PUQ Web Hosting

Server Group

None

License key

success: License check disabled (DEV)

General

Web limits

Mail limits

DNS limits

Client

Email

Config options

Toggles controlling which client-area pages/actions are visible. Unchecked = hidden.

Mail

☒ Create mailbox
 ☒ Delete mailbox
 ☒ Change mailbox password
 ☐ Mailbox settings — forwards / auto-responder VANITY
☒ Webmail SSO link

Storage

☒ Create database
 ☒ Delete database
 ☒ Create FTP account
 ☒ File manager SSO link

DNS / SSL / Misc

☒ Edit DNS records
 ☒ Allow external-DNS mode (skip our nameservers)
 ☒ Install SSL certificates
 ☒ Manage cron jobs
 ☒ Web settings (PHP backend template, HTTPS redirect)
 ☒ Backups (list / create / delete / restore)
 ☒ View server logs (apache access/error, exim main)
 ☒ Client-side redeploy

For Vanity products the client gets a single fixed mailbox; "Create / Delete mailbox" don't apply — use "Mailbox settings" + "Change password".

Email — lifecycle emails

The **Email** tab maps a WHMCS email template to each module event (Account Ready, Deploy Failed, Mailbox/Database/FTP/Backup Created, SSL Installed, SSL Expiring 14/3, Quota Warning 80/Exceeded, DNS Zone Deployed, Backup Restored/Failed). Leave an event blank to disable it.

Module NamePUQ Web Hosting

Server GroupNone

License key

success: License check disabled (DEV)

General

Web limits

Mail limits

DNS limits

Client

Email

Config options

Pick which WHMCS email template is sent for each event. Leave blank to disable.

Account Ready

Deploy Failed

Mailbox Created

Database Created

Ssl Installed

Ssl Expiring 14

Ssl Expiring 3

Quota Warning 80

Quota Exceeded

Dns Zone Deployed

Ftp Created

Backup Created

Backup Restored

Backup Failed

Module NamePUQ Web Hosting

Server GroupNone

License key

success: License check disabled (DEV)

General

Web limits

Mail limits

DNS limits

Client

Email

Config options

Configurable Options

The module reads its per-order limits (and, in Vanity, the parent domain) from WHMCS Configurable Options, and the customer's name (Vanity) from a product Custom Field `vanity_name`. This creates the ones this product needs — found by name, missing ones added. Safe to run any number of times; existing options and values are never changed. Prices start at 0 — tune them under *Configurable Options* afterwards. The set shown depends on the Deployment mode + ticked roles above; for Vanity, the `vanity_domain` values come from the group's Vanity domains.

Create / sync missing

Refresh

Option	Key	Source	Status	Default values
Web disk	<code>web_disk_quota</code>	static	MISSING	1024 1 GB, 2048 2 GB, 5120 5 GB, 10240 10 GB, 20480 20 GB
Web bandwidth	<code>web_bandwidth</code>	static	MISSING	0 Unlimited, 51200 50 GB, 102400 100 GB, 512000 500 GB
Databases	<code>web_databases</code>	static	MISSING	1 1, 5 5, 10 10, 25 25
FTP accounts	<code>web_ftp_accounts</code>	static	MISSING	1 1, 5 5, 10 10
Cron jobs	<code>web_cron_jobs</code>	static	MISSING	5 5, 10 10, 25 25
Web backups	<code>web_backups</code>	static	MISSING	0 None, 3 3, 7 7
Mail disk	<code>mail_disk_quota</code>	static	MISSING	1024 1 GB, 2048 2 GB, 5120 5 GB, 10240 10 GB
Mailboxes	<code>mail_accounts</code>	static	MISSING	1 1, 5 5, 10 10, 25 25
Mail backups	<code>mail_backups</code>	static	MISSING	0 None, 3 3, 7 7
DNS records	<code>dns_records</code>	static	MISSING	25 25, 50 50, 100 100, 250 250

After syncing they show **COMPLETE**:

DetailsPricingModule SettingsCustom FieldsConfigurable OptionsUpgradesFree DomainCross-sellsOtherLinks

Module NamePUQ Web Hosting

Server GroupNone

License key

success: License check disabled (DEV)

GeneralWeb limitsMail limitsDNS limitsClientEmailConfig options

⚙️ Configurable Options

The module reads its per-order limits (and, in Vanity, the parent domain) from WHMCS Configurable Options, and the customer's name (Vanity) from a product Custom Field `vanity_name`. This creates the ones this product needs — found by name, missing ones added. Safe to run any number of times; existing options and values are never changed. Prices start at 0 — tune them under *Configurable Options* afterwards. The set shown depends on the Deployment mode + ticked roles above; for Vanity, the `vanity_domain` values come from the group's Vanity domains.

✔ Create / sync missing

🔄 Refresh

Created 10 option(s) + 37 value(s) + 0 custom field(s).

Option	Key	Source	Status	Default values
Web disk	<code>web_disk_quota</code>	static	COMPLETE 5/5	1024 1 GB, 2048 2 GB, 5120 5 GB, 10240 10 GB, 20480 20 GB
Web bandwidth	<code>web_bandwidth</code>	static	COMPLETE 4/4	0 Unlimited, 51200 50 GB, 102400 100 GB, 512000 500 GB
Databases	<code>web_databases</code>	static	COMPLETE 4/4	1 1, 5 5, 10 10, 25 25
FTP accounts	<code>web_ftp_accounts</code>	static	COMPLETE 3/3	1 1, 5 5, 10 10
Cron jobs	<code>web_cron_jobs</code>	static	COMPLETE 3/3	5 5, 10 10, 25 25
Web backups	<code>web_backups</code>	static	COMPLETE 3/3	0 None, 3 3, 7 7
Mail disk	<code>mail_disk_quota</code>	static	COMPLETE 4/4	1024 1 GB, 2048 2 GB, 5120 5 GB, 10240 10 GB
Mailboxes	<code>mail_accounts</code>	static	COMPLETE 4/4	1 1, 5 5, 10 10, 25 25
Mail backups	<code>mail_backups</code>	static	COMPLETE 3/3	0 None, 3 3, 7 7
DNS records	<code>dns_records</code>	static	COMPLETE 4/4	25 25, 50 50, 100 100, 250 250

Product is already installed.

Tune the **prices** afterwards under WHMCS *Configurable Options*. The product is now ready to order.

“ For a **Vanity** product the limits and config-options are different (Website/Mailbox quota + the `vanity_domain` option + `vanity_name` custom field). See **Vanity Mode → The vanity product**.