Changelog

Proxmox KVM module **WHMCS**

Order now | Download | Community

v4.1.0 — 08-10-2026

Release 4.1.0 delivers powerful new capabilities to elevate your hosting experience: native **SSH Key Management** with zero-touch Cloud-Init injection, high-performance **Smart Bulk Provisioning**, seamless **Proxmox VE 9+ noVNC Web Console** integration, automated ISO mounting, dynamic network bridge discovery, robust OS reinstallation, and complete 26-language localization parity.

SSH Keys & Zero-Touch Authentication

- **Client SSH Key Manager**: Clients can now store, label, and manage their OpenSSH public keys (`ssh-ed25519`, `ssh-rsa`, `ecdsa-sha2-*`) directly in the Client Area, complete with cryptographic fingerprint verification.
- **Automated Cloud-Init Injection**: Stored SSH keys can be selected during shopping cart checkout (`cart.php`) and operating system reinstallation, automatically injecting authorized keys into the virtual machine for instant, secure passwordless login.
- **Centralized Admin Oversight**: Administrators can view, filter, search, and manage all client SSH keys across the hosting platform from **Addons → PUQ Proxmox KVM → SSH Keys**.
- **Flexible Navigation Shortcuts**: Configurable menu options in Addon settings allow administrators to place SSH key management shortcuts in the Client Area profile menu, services navigation, or both.

High-Speed Smart Bulk Provisioning

- **Lightning-Fast Cart Checkout**: Eliminated shopping cart timeouts when clients purchase multiple virtual machines in a single order. Bulk orders now confirm and activate in milliseconds.

- **Intelligent Background Provisioning Queue**: Multi-VM orders are automatically queued and provisioned sequentially in the background via the automated cron engine, preventing server overload and execution timeouts.
- **Deferred Welcome Notifications**: Welcome emails with VM access credentials are automatically deferred until each virtual machine is fully cloned, configured, and running, ensuring clients always receive valid server details.
- **Configurable Bulk Threshold**: Hosting providers can easily customize the queue activation threshold (`bulk_provision_threshold`) in Addon Module General Settings to tailor provisioning behavior to their infrastructure.

Proxmox VE 9+ Console & Virtual Media

- **Proxmox VE 9+ noVNC Compatibility**: Full out-of-the-box compatibility with Proxmox VE 9+, delivering fast, reliable, direct HTML5 noVNC console connections without connection drops or proxy timeouts.
- **Automated CD-ROM Auto-Provisioning**: The deployment engine now automatically detects and attaches a virtual CD-ROM drive to newly cloned VMs. Clients can seamlessly mount custom ISOs and boot rescue images, even when deploying modern Cloud-Init templates that lack a default optical drive.

Network & Infrastructure Automation

- **Dynamic Network Bridge Discovery**: Product configuration now features an interactive AJAX bridge selector that automatically discovers active Linux Bridges (`vmbr0`, `vmbr1`), Open vSwitch (OVS) bridges, and Proxmox SDN VNets across cluster nodes.
- **Reliable Lifecycle & Termination Sync**: Enhanced termination automation guarantees that cancelling or terminating an expired service completely frees cluster resources, releases IP pool allocations, and synchronizes WHMCS service status to `Terminated`.

Flawless OS Reinstallation

- **Universal Template Resolution**: Overhauled OS reinstallation and provisioning logic to dynamically discover and validate template availability across cluster nodes, ensuring 100% reliable reinstall operations on existing virtual machines.
- **Enhanced Confirmation & Safety Modals**: Upgraded reinstallation dialogs with clear safety warnings and confirmation inputs to prevent accidental data loss.

Global Localization (26 Languages)

- **Complete 26-Language Parity**: Full translation coverage across all 26 supported WHMCS languages with 100% dictionary key parity.

- **Native User Experience**: Every client-facing prompt, reinstallation dialog, virtual machine deletion notice, and power action notification is fully localized in the client's chosen language.

Enterprise Security & Architecture

- **CSRF Protection Across All Operations**: Integrated robust anti-CSRF token verification across all client and administrator actions, protecting snapshots, backups, firewall rules, and OS reinstalls.
- **Resilient Usage Metrics**: Isolated resource usage metrics collection ensuring that statistical queries for individual VMs run independently without affecting cluster-wide billing data.
- **Performance & Environment Guidance**: Integrated built-in web server and PHP performance recommendations directly into the module settings, helping administrators optimize their hosting environment for peak efficiency.

---

v4.0.0 — 03-09-2026

Major release: Dual-mode snapshot architecture, unified client UX matching backups, modernized usage charts, ionCube 15 support, and comprehensive multi-language synchronization.

Dual-Mode Snapshot System

- **Snapshot Schedule Mode**: New product setting allowing automated weekly scheduled snapshots with automatic FIFO rotation, functioning identically to scheduled backups.
- **Automated Rotation**: When the snapshot quota (e.g. 3/3) is reached, the oldest snapshot is automatically purged before creating the new scheduled snapshot.
- **Dedicated Cron Task**: Registered `scheduleSnapshot` cron task running every 5 minutes to execute scheduled snapshots labeled `Scheduler: YYYY-MM-DD`.
- **Self-Healing Schema Migration**: Automatic database column verification and dynamic migration (`vm_snapshot_schedule`, `vm_snapshot_last`).
- **Dynamic Product Settings**: Enabling `Snapshot schedule mode` dynamically locks and disables `Snapshot lifetime`, preventing conflicting cleanup policies.

Client Area UI & UX Modernization

- **Snapshots Page**: Rebuilt with simultaneous server-side rendering of the schedule card and loader spinners matching the backups page (no jumping elements or layout shifts).
- **Usage Charts (`charts.tpl`)**: Completely refactored with dedicated cards for CPU & RAM,

Disk I/O, and Network, integrated `kvmAjax` client, full localization, and responsive auto-scaling on window resize.

Localization & Compatibility

- **25 Languages Synchronized**: Comprehensive audit and translation across all supported WHMCS languages.
- **Ukrainian Language Compatibility**: Added support for both `ukranian` and `ukrainian` WHMCS naming standards.
- **Two-Tier Fallback Loader**: Enhanced `puqProxmoxKVM_LoadLang()` to load base English defaults before applying user translations, eliminating blank strings.
- **ionCube 15 Architecture**: Standardized packaging on ionCube 15 with universal out-of-the-box PHP support.

---

v3.3.1 — 15-05-2026

Compatibility patch for **Proxmox VE 9.1.1** and templates that don't use VirtIO.

Proxmox 9.x strict schema — fixed

Proxmox VE 9.x hardened its API schema validation. The «Set system disk I/O» step started failing with `HTTP 400: {"_root":"property is not defined in schema"}` whenever the module couldn't pin down the system disk from the VM config. Now bulletproof — see below.

Robust system-disk detection

Any combination of the following templates is now handled correctly:

- **SCSI / SATA / IDE system disks**.
- **Legacy** `bootdisk: scsi0` parameter from older Proxmox versions.
- **Boot order with CD-ROM first** (`boot: order=ide2;scsi0`) — cloudinit / CD-ROM / network entries are skipped, the first real data disk wins.
- **Templates without a** `boot:` **line at all** — the first detected data disk is used as a fallback.

Additional disk creation — works on any

storage

Format detection for the new additional disk used to fail when the system disk volid contained extra dots (storage names with `.`, path-style volids on certain plugins). The parser now extracts only the trailing extension and validates against the Proxmox enum (`raw`, `qcow2`, `vmdk`, …), falling back to `raw` for block storage and volids without an extension. No more `format error: value 'qcow2/101/vm-101-disk-0' does not have a value in the enumeration` on creation.

Bandwidth step — no more useless API calls

`Set system / additional disk I/O` now skips the API call entirely when all four bandwidth/IOPS values are `0`. Faster deploys, clean cron logs, and one less surface for schema-strict Proxmox versions to fail on.

---

v3.3 — 14-05-2026

Configurable Options release: full coverage with plain-English names, Module Settings defaults for every resource, disk shrink protection.

Configurable Options

- **18 supported options** with plain-English names: `CPU Cores`, `RAM`, `System Disk`, `Additional Disk`, `System Disk Read/Write Bandwidth`, `System Disk Read/Write IOPS`, `Additional Disk Read/Write Bandwidth`, `Additional Disk Read/Write IOPS`, `Network Bandwidth`, `IPv4 Addresses`, `IPv6 Addresses`, `Backups`, `Snapshots`, `Operating System`.
- **11 are new in v3.3** — all disk size / bandwidth / IOPS parameters and Network Bandwidth.
- Legacy prefix names (`B|...`, `S|...`, `CPU|...`, `RAM|...`, `ipv4|...`, `ipv6|...`, `OS|...`) still work. When both forms exist on the same product, the plain name wins.

Module Settings defaults

A product now works without any Configurable Options at all — every resource has a default in Module Settings, overridden only when a matching option is assigned to the service. New default fields: `Backups` and `Snapshots` count in VM Configuration; `IPv4 count` and `IPv6 count` in Network.

Disk shrink protection

Proxmox cannot shrink disks. v3.3 blocks downgrades at three layers:

1. **Upgrade page** — smaller sub-options are visually disabled with `(downgrade not allowed)`, warning banner above the form, client-side submit guard.
2. **Change-package state machine** — backend skips the resize step with `skip — shrink not allowed by Proxmox`, VM is not stopped, snapshots are not removed.
3. **Post-backup-restore** — re-applying a smaller package size after restore is treated the same way.

Additional Disk = 0 deletes the disk

Selecting `0` for Additional Disk now detaches the disk and purges the file from storage. Upgrade form labels the sub-option `(removes the existing disk — data will be lost)` and requires a JavaScript `confirm()` before submit. To disallow this for clients, omit the `0|...` sub-option from the Additional Disk dropdown.

Faster change-package

The Start VM step at the end of a change-package now polls for up to 60 seconds in the same cron pass. Slower-starting VMs (cloud-init, large memory) no longer force a one-minute wait for the next cron tick.

---

v3.2 — 18-04-2026

A DNS, lifecycle and admin-UX release. Key goal: long-running operations (provisioning many DNS records, tearing down a service with large backups) must never time out the WHMCS request. Both **Set DNS records** and **Terminate** now run asynchronously in cron with live progress streamed to the cron output. Under the hood — full null-safety hardening across both modules for PHP 8.1/8.2 stability.

PowerDNS provider

Native support for the **PowerDNS Authoritative Server REST API** as a third DNS provider (alongside Cloudflare and HestiaCP). Works out of the box with standard PowerDNS installations — configure `server` URL and `api_key`, the module takes care of the rest. Fully integrated with forward and reverse zones, automatic `ensureTrailingDot` / FQDN normalization, and PowerDNS-strict content formatting for PTR / CNAME / NS records.

Asynchronous Set DNS records

The **Set DNS records** admin button used to call the Proxmox and DNS APIs synchronously — on a service with many reverse-DNS records it would exceed the PHP execution limit and fail with a blank error page. The button now queues the job by setting the VM status to `set_dns_records` and returns `success` instantly. The cron task picks it up on the next tick, runs `DeleteDNSRecords` + `SetDNSRecords`, and writes a full step-by-step log to the VM record.

Asynchronous Terminate

Same treatment for service termination. When an admin clicks Terminate, the module sends a fire-and-forget "stop" request to Proxmox, sets `vm_status = 'terminate'`, returns `'success'` — and WHMCS marks the service Terminated immediately. The actual heavy work (graceful stop with polling, backups removal, DNS deletion, VM DELETE API call, DB cleanup) is done by cron.

Benefits:

- Client loses access to the service instantly; no waiting 30+ seconds for the admin action to complete.
- Large backup cleanup and bulk DNS deletion can't time out the HTTP request anymore.
- The VM starts shutting down in the background while the cron queue is still processing other VMs — by the time cron picks it up, it's often already stopped.

Robust VM stop polling

The terminate flow previously used a fixed 15-second stop window which was insufficient for VMs with large memory footprints or QEMU guest-agent filesystem freeze. It now issues a single stop request and polls the remote status every 5 seconds for up to 120 seconds (graceful), then a 60-second force-stop window. Live progress is emitted every 15 seconds so admins see what's happening.

New `error_terminate` status + Reset / Delete Record actions

When termination fails (for example, the Proxmox API DELETE call returns an error), the VM no longer silently falls back to `remove`. It's now marked `error_terminate`:

- Cron **never automatically retries** — the admin has to act.
- The VM record is **not deleted** from the database and the IPs stay allocated in the pool, so they cannot be accidentally reassigned to another client while the failing VM is still present on Proxmox.
- A clear red error banner in the VM Log modal shows the failure reason.

The **Reset VM Status** modal has been expanded with `terminate` (retry) and `remove` (force-mark) options, plus an embedded reference table explaining when to use each status. A new **Delete Record** button (trash icon) appears for rows in `error_terminate` / `remove` status — it removes the row from `puqProxmoxKVM_vm_info` only, with an explicit confirmation dialog warning that Proxmox state is not touched.

Live cron output

The standalone cron (`php cron.php`) now streams every individual step in real time with timestamps. During a deploy you can watch DNS records being created zone by zone, IP by IP, instead of waiting 60 seconds and seeing only the summary. During a terminate you see `stop request sent`, periodic `still running, waited Xs / 120s` heartbeats, each DNS deletion, the final `VM deleted`. Output is flushed after every line — nothing is buffered.

DNS zones UX + credentials never leave the server

The DNS Zones page now shows three provider types (Cloudflare, HestiaCP, PowerDNS) with a single unified CRUD interface. Secret fields (API tokens, admin passwords, API keys) are **no longer returned to the browser** — the edit form shows `(unchanged — enter new to replace)` placeholders, and the save flow preserves the stored value if the field is left empty.

IP Pools — automatic reverse-DNS zone hint

When configuring an IP Pool, the required reverse-DNS zone name for the prefix is now computed automatically and shown as a hint both in the add/edit modal and as a second line in the Addresses column of the pool list. For example, a `2001:db8::/120` pool shows:

```
2001:db8::2 - 2001:db8::50
```

```
rDNS zone:  0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa
```

Admins no longer have to compute nibble reversals by hand — copy the value straight into the DNS Zones form. Both IPv4 (`/8`, `/16`, `/24`) and IPv6 (any nibble-aligned prefix) are supported; non-aligned prefixes show a "classless delegation required" hint.

DNS record creation — reliability

A collection of DNS bugs fixed in one pass:

- **IPv6 PTR names** — previously computed via `str_replace(':', '')` which produced a garbage PTR for any compressed IPv6 address (e.g. `2001:db8::1` became `1.8.b.d.1.0.0.2.ip6.arpa` instead of the correct 32-nibble form). Now uses `inet_pton` + `bin2hex`, correct for every IPv6 form.
- **PowerDNS PTR/CNAME/NS content** — automatically wrapped with a trailing dot so PowerDNS strict mode does not reject records with "Record content malformed".
- **HestiaCP server URL** — normalized on save so it always ends with `/`, regardless of how the admin types it.
- **Zone suffix matching** — zones saved with a trailing dot (`example.com.`) now match correctly against record names.
- **IPv6 DNS1-only / DNS2-only** — an IP pool with only `dns1` or only `dns2` now correctly sets the VM's DNS server; previously only the `dns1+dns2` combination worked reliably.

Non-blocking DNS errors

DNS API failures (zone missing, provider down, auth error) **never block** deploy, change-package, or terminate. Each zone and each record is wrapped in individual try-catch and logged as a non-blocking event. The operation proceeds with the rest. A summary (`forward_ok/err`, `rev_ok/err`, per-zone messages) is written to the VM log and, when errors occurred, to the WHMCS module log as well.

VM Management page polish

- **IPs column** — each IP is now shown together with its rDNS on the line below (smaller, muted) as one visual block. Easy to scan.
- **Actions column** — all buttons stay on a single row.
- **Filters** — service-status and vm-status filters remember the admin's last choice in `localStorage` and restore it on the next visit. Default is now "All" on first visit (used to be "Active").

Admin UX — clearer error surfacing

- VM Log modal shows a prominent red alert banner at the top when `last_error_action` / `last_error_message` are present in the last action log.
- Client Activity Log gets exactly one entry per terminate attempt — "terminated successfully" on success, "termination FAILED — admin attention required: " on error. No spam.

Under the hood

- **Full null-safety audit** across both modules: all typed-property assignments, `$params['...']`, `$_GET['...']`, and `explode()[n]` reads now use `?? default` or bounds checks. Prevents `TypeError: cannot access offset on null` warnings on PHP 8.1/8.2 when Proxmox API responses or DB rows omit optional fields.
- Removed dead code: legacy pre-ProxmoxApi ticket parsing, unused `isOk()` helper, commented-out stubs (~40 lines total).
- Cloudflare / HestiaCP DNS clients harden `json_decode` results against non-JSON / empty responses.
- `cleanupFirewall` now distinguishes benign "IPSet does not exist" from real API errors (auth, 500) and logs the latter.
- Live-cron logger is shared across deploy, change-package, and terminate flows — consistent output format everywhere.

---

v3.1 — 16-04-2026

A stability and admin UX release on top of v3.0. Focused on making product configuration self-explanatory and hardening the cron against bad data.

Actionable errors on the Product Configuration page

The custom Module Settings UI no longer fails silently when something is wrong with the product's Server Group. Instead of a generic "No server found" message, the page now shows a contextual banner with an exact fix-it hint and highlights the affected fields (Node, OS Template, Storages):

- **"Server Group is not selected"** — when the product hasn't been saved or no group is

assigned.
- **"Server Group no longer exists"** — when the referenced group was deleted.
- **"Server Group has no servers assigned"** — with a direct path to `Setup → Products/Services → Servers → Edit group`.
- **"Server Group references a missing server"** — when the group still exists but points to a deleted server.

Cron stability — safe handling of incomplete network data

Fixed a regression where one service with a missing IP-pool entry or server address field could crash the entire `processVirtualMachines` cron run on PHP 8.0+. All assignments from `server_address_list` and IP pool data (netmask, gateway, DNS, bridge, VLAN) are now null-safe, so the cron continues processing the rest of the queue even if a single service has stale or incomplete network configuration.

Statistics collection fix

`GetStatistics()` now resolves the VM's current Proxmox node before collecting RRD data and safely skips services whose remote node is not yet known (for example, services still in the deployment queue). Prevents spurious errors in the statistics cron.

---

v3.0 (April 2026) — Major Release

Version 3.0 is a complete rewrite with a new architecture, dedicated addon module, and dozens of new features. This is the biggest update since the initial release.

New Dedicated Addon Module

The PUQ Customization addon module is **no longer required**. Version 3.0 includes its own dedicated addon module (`puq_proxmox_kvm`) with:

- **Dashboard** — centralized overview of all resources: IP pools, DNS zones, KVM services
- **IP Pool Management** — redesigned with per-server pools, usage visualization, improved validation
- **DNS Zone Management** — Cloudflare and HestiaCP integration for automatic forward/reverse DNS

- **VM Management** — centralized view of all VMs across all servers with deploy logs, status monitoring, retry/reset actions, and database record inspection
- **Settings** — multi-page settings (General + Cron) with API timeouts, migration configuration, and per-task cron intervals
- **Auto-migration** — seamless migration from old `puq_customization` tables on first activation
- **Access control** — configurable admin role groups for addon access

Deploy State Machine

The VM deployment process has been completely rewritten as a **step-by-step state machine**:

- Each deployment step is executed individually with status tracking
- If any step fails, deployment pauses and **resumes automatically** on the next cron run
- Full deploy log with per-step timing, status transitions, and error messages
- Visible in both CLI output and admin UI

Deploy steps: Allocate IP → DNS + Clone → Migrate to target node → Set CPU & RAM → Resize system disk → Set disk I/O → Create additional disk → Resize additional disk → Set additional disk I/O → Configure network → Configure firewall → Configure cloud-init → Start VM → Verify running + Email

Post-Clone VM Migration

New intelligent migration system for cross-node deployment:

- After cloning to the template node, VMs are **automatically migrated** to the target node with the correct storage
- Supports offline migration with storage mapping (`targetstorage` parameter)
- Finds suitable target nodes based on storage availability and free RAM
- Configurable migration timeout with cron-based retry
- If migration fails or no suitable node is found, VM stays on the template node and deployment continues

Change Package State Machine

Package upgrades/downgrades have been rewritten with the same state machine approach:

- **12 individual steps**: Update IP/DNS → Stop VM → Set CPU/RAM → Resize disks → Configure I/O → Configure network → Configure firewall → Start VM → Verify running
- Each step checks if a change is actually needed and **skips if no change** detected
- Full logging to `vm_last_action_log` with step-by-step detail

- Resilient to failures — continues from last successful step on next cron run

Firewall Management

Complete firewall feature — both for deployment and client self-service:

- **Deploy configuration** — firewall options (enable, DHCP, NDP, MAC filter, IP filter, log levels), policies (input/output), and anti-spoofing IPSet are configured during deployment
- **Client area** — full firewall rules management page: add rules, delete rules, drag-and-drop reorder, change input/output policies
- **Admin product settings** — new "Firewall" panel with all options configurable per product
- **Rule validation** — server-side validation of action, direction, protocol, IP/CIDR, port ranges

Cron System

Flexible cron with two modes:

- **WHMCS Hook mode** (default) — runs automatically with WHMCS cron
- **Standalone mode** — independent cron file with CLI tools (`--task`, `--force`, `--list`, `--help`)
- **Per-task intervals** — each cron task has its own configurable interval (set to 0 to disable)
- **Lock management** — flock-based locking with stale PID detection and configurable timeout
- **Structured results** — each cron function returns `{processed, errors, details}` for monitoring

Client Area Redesign

All client area pages have been fully redesigned:

- **AJAX-based** — no Proxmox API calls on page load, all data loaded asynchronously
- **Modern UI** — consistent card-based design with PUQ styling
- **Session cache** — 30-second cache for VM status to reduce API load
- **Fast poll** — 1-second status polling after start/stop for instant feedback
- **New Firewall page** — rules management with drag reorder, policies, add/delete
- **Network info message** — notification when additional IPs need manual configuration
- **Translation support** — all UI text through `L()` helper, 25 languages

Admin Area Improvements

- **Custom product settings UI** — Bootstrap-based panels replacing default WHMCS configoptions for VM Configuration, Storage, Network, Firewall, Integrations, Email Templates, Client Permissions
- **Real-time VM information** — JSON AJAX panel with status, CPU, RAM, disk, network details
- **Deploy log viewer** — expandable step-by-step deploy history with timing
- **Change package log** — step-by-step change history with skip indicators
- **noVNC + Redeploy buttons** — quick actions in admin service view
- **Metric billing** — bandwidth usage in/out with WHMCS Metric Billing integration

Security & Stability

- **Path traversal fix** — whitelist validation for addon page routing
- **Admin session check** — explicit `$_SESSION['adminid']` verification in addon AJAX
- **Input validation** — firewall rules, IP/CIDR, port ranges validated server-side
- **Error handling** — safe `explode()` operations, operator precedence fixes, try-catch on database operations
- **DNS log filtering** — sensitive tokens/passwords removed from log output
- **PHP 7.4 compatibility** — `str_contains()` replaced with `strpos()`, no PHP 8.0+ features required

Compatibility

Component	Supported Versions
WHMCS	8.x, 9.x
PHP	7.4, 8.1, 8.2
Proxmox VE	8.x, 9.x
ionCube Loader	v13, v14, v15

---

v2.4 — 31-08-2025

- Accounted for a custom path to the WHMCS admin panel.
- Direct links now take the `WHMCS System URL` parameter into consideration.

v2.3 — 09-08-2025

- **Breaking change:** authentication switched from login/password to **Proxmox API token** . Users who update **must** create a token and enter the new credentials in the Proxmox server settings. Username must be in the format `root@pam! whmcs-dev` (token ID), password — the token value itself.
- Renamed the anti-spoofing rule filter from `wm-VMID` to `ipfilter-net0`.
- Various performance improvements that increased the module's response speed.

> **Warning:** before updating to v2.3+, create a Proxmox API token and enter its details in the server settings.

v2.2 — 14-07-2025

- Backup restoration mechanism improved.
- Security fixes implemented.
- Client web interface updated: button-related bugs fixed, loaders added.
- Adapted for compatibility with Proxmox v8.4.

v2.0 — 23-09-2024

- Module is now coded with **ionCube v13**.
- Supported PHP versions:
  - PHP 7.4 — WHMCS ≤ 8.11.0
  - PHP 8.1 — WHMCS ≥ 8.11.0
  - PHP 8.2 — WHMCS ≥ 8.11.0
- Added an active check for PUQ Customization + extension `Module PuqProxmoxKVM`.

v1.5 — 04-03-2024

- Fixed an `No IPv6 addresses available` bug that occurred during IPv6 assignment in some

cases.
- Fixes in client-zone templates.
- Changed the display of the product in the admin area.
- Added metrics for incoming and outgoing traffic (usage-based billing now possible).

v1.4.5 — 11-10-2023

- Added support for WHMCS 8.8.0.
- Translations added/updated for 25 languages: Arabic, Azerbaijani, Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, Farsi, French, German, Hebrew, Hungarian, Italian, Macedonian, Norwegian, Polish, Romanian, Russian, Spanish, Swedish, Turkish, Ukrainian.

v1.4 — 24-07-2023

- Added synchronization of forward and reverse DNS zones (required PUQ Customization): **Cloudflare**, **HestiaCP**.
- The "Change Package" function moved to cron.
- Fixed a bug related to the default operating system template.
- Added Virtual Machine Templates (CentOS 9).

v1.3 — 11-07-2023

- Integration with **PUQ Customization** (FREE addon).
- **IPv6** support (required PUQ Customization).
- Ability to create VMs with IPv6 only.
- Added **pools of IP addresses** (required PUQ Customization).
- Added ability to define multiple IPv4 and IPv6 addresses.
- Added configurable options for RAM, CPU, IPv4, IPv6.
- Added a check that re-runs cloning if the previous cloning attempt failed.
- Redesigned the main screen of the client area (dropdown list with VM network settings).
- Changed the display of VM graphs in the admin area (3 graphs per row).
- Removed the "name servers" fields from the order form.
- Added VM templates (Debian 12, Ubuntu 22.04).

v1.2.1 — 04-03-2023

- Support for PHP 8.1 and PHP 7.4.
- Changes made to templates.

v1.2 — 06-01-2023

- Support for WHMCS 8.6.
- Support for ionCube PHP Loader v12.
- Support for PHP 8.1.
- Changes made to templates.

v1.1 — 12-10-2022

- Modified security.
- Remote debug logging in the admin panel.
- Corrections in translations.
- Fixed a bug that incorrectly checked whether a service belongs to the logged-in client in the client area.

v1.0 — 19-09-2022

First public release.

Revision #38
Created 22 October 2022 06:37:11 by Ruslan
Updated 8 October 2026 19:20:47 by Ruslan