00. Changelog

v2.4.0 — October 2026

Major release — IPAM server IP conflict resolution, backup archive ZIP integrity fix, native Telegraf metrics endpoint, comprehensive Debian 12 & 13 kernel support, WHMCS integration documentation, and automated system testing framework.

IPAM & OpenVPN IPv6 Collision Resolution

Resolved an IP address allocation conflict where server interface addresses were erroneously assigned to clients:

- **Server Interface Exclusion** — the IP allocation engine (`IPAM`) now strictly reserves and excludes the server's local interface IP addresses (e.g. `...::1` on IPv6 and gateway addresses on IPv4) from being automatically assigned to new clients.
- **CCD Configuration Partitioning** — prevents duplicate OpenVPN CCD entries sharing the server IP in dual-stack IPv4/IPv6 networks, guaranteeing collision-free routing.
- **Client Provisioning Safety** — client deletion and re-creation no longer re-allocates previously reserved server addresses.

Backup Archive ZIP Integrity Fix

Fixed a critical archive generation bug that produced corrupted ZIP files:

- **Writer Close Sequence** — corrected Go `defer` LIFO ordering in `backups.go`. Explicitly flushes and closes `zip.Writer` before closing the underlying file descriptor, ensuring Central Directory (CD) and End of Central Directory (EOCD) records are written completely.
- **Archive Validation** — all system backup archives now pass strict verification (`zip -T` and `unzip -t`) with zero CRC errors or missing directory headers.
- **Cross-Host Migration** — verified cross-node transfer and extraction on secondary instances.

Local InfluxDB / Prometheus Metrics Endpoint (`/metrics`)

Added a dedicated metrics route to the web server:

- **Native Router Endpoint** — registered `/metrics` on the primary web server engine, delivering InfluxDB line-protocol metrics for online peers, network traffic, CPU, and RAM.
- **Local Telegraf Integration** — resolves HTTP 404 errors during local Telegraf metric scraping when Let's Encrypt SSL is disabled or when running on custom ports (`8098`).
- **Real-Time Monitoring** — provides zero-delay metrics scraping for external monitoring stacks (Grafana, InfluxDB, Prometheus).

Debian 12 (Bookworm) & Debian 13 (Trixie) Support for AmneziaWG

Enhanced repository management and installation scripts:

- **Direct ASC Keyring Support** — updated repository provisioning to use modern detached GPG keyrings (`/usr/share/keyrings/`) without deprecated `apt-key`.
- **Kernel Module & DKMS Automation** — streamlined automated DKMS build instructions across modern Linux kernel versions (Debian 12 Bookworm, Debian 13 Trixie, Ubuntu Jammy/Noble).

Automated System Testing & Stress Suite

Introduced a comprehensive test framework in the developer tools:

- **8-Module Test Runner** — automated validation covering REST API health, IPAM allocation, live protocol handshakes (WireGuard, AmneziaWG, OpenVPN, IKEv2), bandwidth traffic shaping (`iperf3`), firewall/ipset isolation, DNS adblocking, and clean node package installation.
- **High-Concurrency Stress Testing** — 50 simultaneous client tunnels across all 4 protocols generating continuous ICMP, HTTP, and DNS traffic with zero packet loss and low memory footprint (~50 MB RSS).

Official WHMCS Module Documentation & Branding

- **WHMCS Client Module Guide** — added complete documentation and integration guides for the official PUQVPNCP WHMCS provisioning module.
- **Brand Standardization** — unified naming to **PUQ Software** across all configuration files, templates, and documentation.

---

v2.3.0 — September 2026

Major release — AmneziaWG anti-censorship stealth WireGuard protocol, strict protocol package validation, reactive multi-tab network interface, and expanded REST API.

AmneziaWG Protocol Support (Anti-Censorship & DPI Bypass)

Full integration of AmneziaWG — a modified WireGuard protocol engineered to evade Deep Packet Inspection (DPI) and strict firewall censorship:

- **Stealth Obfuscation** — randomized magic headers (H1, H2, H3, H4) replace standard WireGuard packet headers, defeating protocol fingerprinting
- **Junk Packet Padding** — configurable junk packets (Jc count, Jmin/Jmax size range, S1/S2 sizes) injected during handshake to mask traffic patterns
- **Per-Network & Global Defaults** — configure default obfuscation parameters globally, with optional per-network custom overrides
- **One-Click Client Provisioning** — generate AmneziaWG configs and QR codes directly from the client management interface
- **Dual-Stack & Bandwidth Control** — supports IPv4/IPv6 dual-stack addressing and HTB bandwidth traffic shaping
- **Diagnostics & Monitoring** — live `awg show` status, peer handshake inspection, and real-time bandwidth metrics

---

Protocol Package Validation & Safety

Prevents enabling VPN protocols whose underlying system packages are not installed:

- **Environment Verification** — checks for `wireguard` / `wireguard-tools`, `amneziawg` / `amneziawg-tools`, `openvpn`, and `strongswan`
- **Network Form Guard** — network settings validate installed packages before allowing a protocol to be activated
- **API Error Mapping** — returns structured, field-mapped error messages highlighting the exact protocol input in the web UI

Reactive Multi-Tab Network Interface

Seamless AJAX experience when managing networks with multiple VPN protocols:

- **Dynamic Tabs** — toggling a protocol checkbox in the main settings tab instantly creates or removes the corresponding protocol tab in the DOM
- **Multi-Tab AJAX Update** — clicking Save commits the network settings and automatically reloads data across all active protocol tabs without a full page reload
- **Zero Page Disruption** — preserve user input and navigation state across tabs

---

REST API & OpenAPI 3.0.3 Expansion

- **170+ REST API Endpoints** — full coverage for WireGuard, AmneziaWG, OpenVPN, IKEv2, firewall, DNS, and diagnostics
- **Full Swagger Documentation** — updated OpenAPI 3.0.3 schema with AmneziaWG parameters and response models

---

---

v2.1.1 — March 2026

Patch release — client bandwidth fix and OpenVPN session stability improvements.

Client Bandwidth Fix

Fixed an issue where per-client bandwidth limits were ignored during client creation:

- **Root cause:** when a network had default bandwidth limits configured, they always overwrote client-specified values — even when the user explicitly set a custom bandwidth

- **Fix:** network bandwidth is now used only as a default when the client has no custom value (`0` / not specified). Client-specified bandwidth always takes priority
- **Impact:** custom per-client bandwidth limits now work correctly on creation

---

OpenVPN Session Stability

Client operations (add, edit, delete) no longer restart the OpenVPN server, preserving all active sessions:

- **Client Add** — generates certificates, updates CCD and password files without restarting OpenVPN (re-read on next client connect)
- **Client Edit** — updates CCD and password files only, no restart needed
- **Client Delete** — revokes certificate and updates CCD/password files, CRL is re-read automatically
- **Management socket** — added per-network OpenVPN management socket for targeted client disconnect
- **DisconnectAllProtocols** — new function that disconnects a specific client from WireGuard, OpenVPN, and IKEv2 simultaneously. Called when a client is disabled or deleted
- **Firewall ordering** — `WaitForOvpnInterface()` ensures tun interfaces are up before committing policy routing after any OpenVPN restart

---

Documentation

- **DNS page** — rewritten documentation with new screenshots for all three tabs (Settings, Records, Ad Blocking)

---

v2.1.0 — March 2026

Feature release — DNS-based ad blocking for all VPN clients, plus stability improvements.

DNS Ad Blocking

Built-in ad and tracker blocking via bind9 Response Policy Zone (RPZ):

- **Block ads and trackers** for all VPN clients at the DNS level — no client-side software

needed

- **4 built-in block lists** — Steven Black Unified (~130k domains), OISD Small (~70k), Pete Lowe (~3k), Hagezi Multi Pro (~170k)
- **Custom lists** — add any blocklist URL (supports hosts format, domain-per-line, wildcard, and adblock filter format)
- **Whitelist** — exclude specific domains from blocking (including all subdomains)
- **Auto-update** — block lists are refreshed automatically every 24 hours
- **Per-list toggle** — enable/disable individual lists without deleting them
- **Statistics** — total blocked domains count, per-list domain count, last update timestamp
- **Zero client configuration** — works automatically for all clients using the built-in DNS server
- **Separate RPZ zone** — ad blocking uses its own zone (`rpz.adblock`), independent of static DNS records

Metric	Value
Default blocked domains	~130,000+
With all lists enabled	~250,000+
Update frequency	Every 24 hours
Supported formats	hosts, domain-per-line, wildcard, adblock filter

New API endpoints:

Method	Endpoint	Description
GET	`/dns/adblock`	Get ad blocking configuration
PUT	`/dns/adblock`	Enable/disable, toggle lists
POST	`/dns/adblock/update`	Force update block lists
POST	`/dns/adblock/lists`	Add custom block list
DELETE	`/dns/adblock/lists/{name}`	Delete custom list
GET	`/dns/adblock/whitelist`	List whitelisted domains
POST	`/dns/adblock/whitelist`	Add domain to whitelist
DELETE	`/dns/adblock/whitelist/{name}`	Remove from whitelist

DNS Page Redesign

The DNS configuration page now uses a **tabbed layout**:

- **Settings** tab — DNS server configuration (forwarders, TTL, ACL)
- **Records** tab — static DNS records (RPZ overrides)
- **Ad Blocking** tab — enable/disable, manage block lists and whitelist

---

OpenVPN Startup Fix

Fixed a race condition where OpenVPN clients could lose internet connectivity after server restart:

- **Root cause:** policy routing rules were committed before OpenVPN tunnel interfaces were fully initialized
- **Fix:** replaced fixed 500ms delay with a polling loop (up to 10 seconds) that waits for all OpenVPN `tun` interfaces to appear before committing firewall and routing rules
- **Impact:** eliminates intermittent "no internet" issues for OpenVPN clients after system restart

---

Other Improvements

- **Client creation simplified** — only `network_name` is required; `name`, `username`, `password`, and `ipv4` are auto-filled from suggested data
- **Client fields renamed** — `allowed_IPs` / `allowed_IPs_ipv6` replaced with cleaner `ipv4` / `ipv6` field names (backward-compatible migration)
- **OpenAPI documentation** — 8 new endpoints documented for DNS ad blocking (156+ total)

---

v2.0.0 — March 2026

Major release — complete rewrite with new architecture, third VPN protocol, and dozens of new features.

New Architecture: Network → Client → Protocol

The core data model has been redesigned from the ground up:

- **Networks** replace the old per-server configuration — each network has its own subnet, protocol settings, firewall, bandwidth limits, and upstream
- **Clients** belong to a network and inherit its settings — one client, one IP address, all protocols
- **Protocols** are configured per-network: enable or disable WireGuard, OpenVPN, and IKEv2 independently for each network

This architecture allows running multiple isolated VPN networks on a single server, each with completely different configurations.

OpenVPN Support

PUQVPNCP now supports **three VPN protocols simultaneously**:

Protocol	New in v2.0
WireGuard	Updated — now per-network
OpenVPN	**New** — full support with PKI, certificates, custom config
IKEv2/IPsec	Updated — now per-network

OpenVPN features include:

- TLS/SSL security with configurable cipher suites
- Built-in certificate management (Root CA, Server, Client certificates)
- Custom server and client configuration directives
- Connection logging and real-time status
- Certificate revocation and renewal
- Automatic `.ovpn` configuration file generation

Upstream WireGuard Tunnels

Route network traffic through external WireGuard VPN servers:

- **Import configuration** — paste a standard WireGuard `.conf` file to create an upstream
- **Per-network routing** — assign different upstreams to different networks
- **Geographic IP rotation** — clients appear from different countries depending on their network
- **Multi-hop privacy** — add an extra encryption layer between your server and the internet
- **System upstreams** — view and manage all WireGuard tunnel interfaces on the system

---

Diagnostics & Environment Integrity

New real-time diagnostics system for monitoring server health:

- **VPN Protocol Status** — WireGuard interfaces, OpenVPN processes, strongSwan IKE SAs
- **DNS & Monitoring** — bind9 status, rsyslog, telegraf, InfluxDB connectivity
- **Network Configuration** — per-network checks: interfaces up, routes configured, firewall chains loaded, traffic control active
- **Integrity Check** — verifies that all system-level configurations match the panel's expected state
- **Real-time logs** — view system and VPN service logs directly from the web interface

---

Per-Network Firewall

Each network now has its own firewall chain set:

- **Filter rules** — control traffic flow within and between networks
- **NAT rules** — source/destination NAT per network
- **DNAT rules** — port forwarding to specific clients
- **Mangle rules** — packet marking for advanced routing and QoS
- **ipset integration** — automatic IP sets per network for high-performance matching
- **Global settings** — client isolation and peering rules at the system level

---

Role-Based Access Control (RBAC)

Fine-grained permission system replacing the old single-admin model:

- **Permission Groups** — create custom groups with granular access (36+ permissions)
- **Multi-user** — multiple administrators with different access levels
- **Default groups**: `admin` (full access), `operator` (all except system settings), `viewer` (read-only)
- **API Tokens** — create tokens with IP restrictions, expiration dates, and user association
- **Session security** — IP-pinned sessions with configurable timeout

---

Traffic Control & Bandwidth

Per-network and per-client bandwidth management:

- **Network-level limits** — set download/upload bandwidth caps for the entire network
- **Client-level limits** — override per-client with individual bandwidth settings
- **Traffic classes** — `tc` (Linux Traffic Control) integration for precise QoS
- **Real-time stats** — view live throughput, drops, overlimits per client
- **Daily breakdown** — traffic statistics aggregated by day with network/client granularity

---

Network Peering

Allow communication between VPN networks on the same server:

- **Pair-based peering** — select which networks can communicate with each other
- **Automatic routing** — firewall and routing rules are generated automatically
- **Apply/revert** — review changes before applying, revert if needed
- **Isolation by default** — networks are fully isolated unless peering is explicitly configured

---

One-Time Links (OTL)

Self-service configuration links for all three protocols:

- **WireGuard** — QR code + downloadable `.conf` file
- **OpenVPN** — downloadable `.ovpn` profile with embedded certificates
- **IKEv2** — downloadable `.mobileconfig` (iOS/macOS) or connection instructions
- **Configurable defaults** — set which protocols are shown, branding, expiration
- **Single-use or time-limited** — links expire after first use or a configured time

---

REST API Expansion

The API has been completely rewritten and expanded:

Metric	v1.x	v2.0
Endpoints	~30	**148+**
Documentation	None	**OpenAPI 3.0**
Authentication	Single hash	**Bearer tokens + RBAC**
Swagger UI	No	**Yes** (`/api_docs`)

New API areas: Networks, Clients, OpenVPN, Upstreams, Peering, Firewall, Diagnostics, Permission Groups, System Users, Profile management.

UI & Theme

- **Dark / Light / Auto** — full theme support with system preference detection
- **Bootstrap 5.3** — modern responsive design
- **AJAX everywhere** — all operations via REST API, no page reloads
- **DataTables** — sortable, searchable tables with responsive layout
- **Toastr notifications** — real-time feedback for all actions
- **Lightbox images** — click to enlarge screenshots in documentation

---

Other Improvements

- **Let's Encrypt auto-SSL** — automatic HTTPS certificate management
- **Port forwarding** — forward external ports to specific VPN clients
- **Custom routes** — push routes to clients per-network
- **Per-network DNS** — configure DNS servers per network
- **Backup to FTP** — automatic backup upload to remote FTP server
- **InfluxDB + Grafana** — export traffic metrics for external dashboards
- **Single binary** — ~34MB Go binary with all assets embedded, no runtime dependencies
- **Systemd integration** — clean service management with auto-restart

---

Migration from v1.x

PUQVPNCP v2.0 is a **ground-up rewrite** and is not directly upgradeable from v1.x. Key differences:

Aspect	v1.x	v2.0
Architecture	Account → WireGuard Server	Network → Client → Protocol
Protocols	WireGuard + IKEv2	WireGuard + OpenVPN + IKEv2
Auth	Single admin + API hash	Multi-user + RBAC + API tokens
Firewall	Global rules	Per-network chains
API	~30 endpoints, no docs	148+ endpoints, OpenAPI 3.0
UI	jQuery + page reloads	Bootstrap 5 + AJAX

For migration assistance, see Troubleshooting or contact PUQ Support.

Revision #30

Created 23 November 2022 07:14:48 by Ruslan

Updated 26 October 2026 16:37:20 by Ruslan