Image not found or type unknown

19. Use Cases

1. VPN Service Provider

Goal: Sell VPN subscriptions to end users with different speed tiers.

Setup

Network	Subnet	Bandwidth	Upstream
`basic`	10.100.1.0/24	10/5 Mbit	Direct (server IP)
`premium`	10.100.2.0/24	100/50 Mbit	Direct (server IP)
`business`	10.100.3.0/24	Unlimited	WireGuard upstream (dedicated IP)

How It Works

1. Create networks with different bandwidth limits
2. Use the official **PUQVPNCP WHMCS Provisioning Module** or custom REST API integration to automate client sales and provisioning
3. Send **One-Time Links** to customers so they can self-configure their devices
4. Monitor traffic via **InfluxDB + Grafana** dashboards

Key Features Used

- Per-client bandwidth limits
- REST API for automation
- One-Time Links for self-service
- All 3 protocols for maximum device compatibility
- Traffic statistics for usage billing

---

2. Multi-Country VPN with

Upstream Tunnels

Goal: Offer VPN exits in multiple countries from a single server.

Setup

```
PUQVPNCP Server (Frankfurt)
+-- Network "exit_us"   ---> Upstream wgup0 (New York)    ---> US IP
+-- Network "exit_uk"   ---> Upstream wgup1 (London)      ---> UK IP
+-- Network "exit_jp"   ---> Upstream wgup2 (Tokyo)       ---> JP IP
+-- Network "exit_de"   ---> Direct (ens18)               ---> DE IP
```

How It Works

1. Rent cheap WireGuard VPN servers in target countries
2. Add them as **Upstreams** in PUQVPNCP (import their `.conf` files)
3. Create a network per country, assign each to its upstream
4. Clients choose their exit country by connecting to the appropriate network
5. One-Time Links can be generated per-network for easy distribution

Key Features Used

- WireGuard Upstreams for geo-routing
- Per-network upstream assignment
- Network-level bandwidth control
- Multi-protocol support (same exit for WireGuard, AmneziaWG, OpenVPN, IKEv2 users)

---

3. Corporate Remote Access

Goal: Provide secure remote access for employees to internal company resources.

Setup

Network	Subnet	Purpose

`employees`	10.0.1.0/24	General employees — access email, intranet
`developers`	10.0.2.0/24	Developers — access staging servers
`admins`	10.0.3.0/24	IT admins — full infrastructure access

Network Peering

Rule	Network A	Network B	Direction
`dev_to_staging`	developers	staging_net	Bidirectional
`admin_full`	admins	all_servers	Bidirectional

How It Works

1. Create separate networks per department with different access levels
2. Use **Network Peering** to control which departments can access which resources
3. Use **Custom Routes** to push only necessary subnets (split tunnel)
4. IKEv2 is ideal here — native on all employee devices, no app to install
5. **Permission Groups** allow the IT team to manage VPN without full admin access

Key Features Used

- Network isolation with peering
- Split tunnel via custom routes
- IKEv2 for zero-install on employee devices
- Permission groups for delegated administration
- Per-client bandwidth to prevent abuse

---

4. IoT Device Management

Goal: Securely connect remote IoT devices (cameras, sensors, POS terminals) back to a central network.

Setup

Network	Subnet	Client-to-Client	Upstream

`iot_cameras`	10.10.1.0/24	No	Direct
`iot_sensors`	10.10.2.0/24	No	Direct
`monitoring`	10.10.3.0/24	Yes	Direct

How It Works

1. Each IoT device gets a WireGuard client with a static IP
2. **Client-to-Client** is disabled — devices cannot communicate with each other
3. The `monitoring` network peers with IoT networks for centralized monitoring
4. **Port Forwarding** rules expose specific device ports to the server's public IP
5. **Bandwidth limits** prevent devices from saturating the connection

Key Features Used

- WireGuard for minimal overhead on constrained devices
- Port forwarding for remote access to specific devices
- Network isolation (no client-to-client)
- Peering for controlled cross-network access
- Bandwidth limits per device

---

5. Hosting Provider — VPN as a Service

Goal: Offer managed VPN accounts to hosting customers, integrated with billing.

Architecture

```
WHMCS / Billing ---> PUQVPNCP API ---> VPN Networks
                                          |
                     +--------------------+
                     |                    |
              Customer A              Customer B
              Network: acme_co        Network: globex_inc
              Clients: 50             Clients: 200
```

```
              Upstream: dedicated     Upstream: shared
```

How It Works

1. Each customer gets a dedicated network with their own subnet
2. Use the official **PUQVPNCP WHMCS Provisioning Module** or direct REST API calls to:
   - Create networks when a customer signs up
   - Add/remove clients based on subscription
   - Suspend clients on non-payment
   - Delete everything on cancellation
3. Customers receive **One-Time Links** for self-setup or download profiles directly from their WHMCS client area
4. **API Tokens** per customer allow limited self-management

Turnkey WHMCS Integration

With the **PUQVPNCP WHMCS Module**, this entire workflow is 100% automated out of the box — including client account provisioning, suspension, quota enforcement, and config delivery (WireGuard, AmneziaWG, OpenVPN, IKEv2).

Key Features Used

- Full REST API (170+ endpoints) for automation
- Per-network isolation for multi-tenancy
- One-Time Links for end-user self-service
- Upstream tunnels for dedicated exit IPs per customer
- Traffic monitoring for usage-based billing

---

6. Privacy-Focused Multi-Hop VPN

Goal: Maximum privacy with double encryption and IP masking.

Setup

```
User ---> PUQVPNCP (Entry Server) ---> Upstream wgup0 (Exit Server) ---> Internet
         WireGuard encrypted            WireGuard encrypted
```

```
          User sees Entry IP            Sites see Exit IP
```

How It Works

1. PUQVPNCP acts as the **entry node** — users connect here
2. All traffic is forwarded through a WireGuard **upstream tunnel** to an exit node
3. The exit node has no knowledge of the original user's IP
4. Even if the entry server is compromised, traffic is encrypted to the exit
5. Different networks can use different exit nodes for variety

Key Features Used

- WireGuard Upstreams for multi-hop
- Multiple upstreams for exit diversity
- All 4 protocols on the entry side for device compatibility
- No logging configuration for privacy

---

7. Small Office / Home Office (SOHO)

Goal: Simple VPN for a small team (5-20 people) with shared file access.

Setup

Network	Subnet	Protocols	Special
`office`	10.0.0.0/24	WG + IKEv2	Client-to-Client: Yes

How It Works

1. Single network with client-to-client enabled
2. Team members connect via **IKEv2** (no app needed) or **WireGuard** (better speed)
3. File sharing works directly between connected devices
4. **One-Time Links** sent to each team member for easy setup
5. The admin manages everything via the web panel with 2-3 permission groups

Key Features Used

- Client-to-client communication
- IKEv2 for no-install setup
- One-Time Links for onboarding
- Simple permission model
- Built-in DNS for local hostname resolution

---

8. Anti-Censorship & DPI Bypass (AmneziaWG)

Goal: Bypass strict government or ISP Deep Packet Inspection (DPI) censorship systems that block standard WireGuard or OpenVPN protocols.

Setup

1. Enable **AmneziaWG** in the network settings
2. Configure custom obfuscation parameters:
   - **H1–H4** random packet headers
   - **Jc** (3–5 junk packets) with sizes **Jmin** (40) and **Jmax** (70)
   - **S1** (15) and **S2** (20) initiation and response junk
3. Generate **One-Time Links** for clients
4. Users scan the QR code or import `.conf` into official AmneziaWG client apps (Android, iOS, Windows, macOS)

Key Features Used

- AmneziaWG kernel-level DPI bypass
- Custom per-network obfuscation parameters
- One-Time Links with direct AmneziaWG client download links
- Upstreams combined with AmneziaWG for external routing

Protocol Selection Guide

Scenario	Recommended Protocol	Why
Mobile users	WireGuard	Fast reconnection, battery efficient
Strict censorship & DPI	AmneziaWG	Header obfuscation, junk packet injection to bypass ISP DPI firewalls
Corporate BYOD	IKEv2	Native on all OS, no app required
Restrictive networks	OpenVPN (TCP)	Works through firewalls, proxies
IoT devices	WireGuard	Minimal resource usage
Maximum compatibility	All four	Users choose what works for them
Speed-critical	WireGuard / AmneziaWG	Lowest latency, highest kernel-level throughput
Legacy devices	OpenVPN	Widest platform support

Revision #9
Created 28 September 2026 17:06:01 by Ruslan
Updated 26 October 2026 16:37:34 by Ruslan