

WireGuard Gateway

Purpose

The WireGuard gateway connects the public VPS to the Bytek home network through an encrypted tunnel.

The gateway allows public HTTPS traffic to reach Traefik without exposing individual home services directly through the Bell router.

The WireGuard gateway provides:

- Encrypted VPS-to-home connectivity.
- Controlled forwarding between the tunnel and the home LAN.
- Network address translation where required.
- Firewall enforcement between public ingress and private services.
- A private path from the VPS to Traefik.
- Separation between the public VPS and the rest of the home network.

Service Information

Setting	Value
Service name	WireGuard Gateway
Private IP address	192.168.2.64
Primary role	VPS-to-home tunnel gateway
Home LAN	192.168.2.0/24
Default gateway	192.168.2.1
Public peer	VPS at 38.29.213.101
Administrative access	SSH key authentication

Public exposure	WireGuard port only
Backup coverage	Proxmox scheduled backup
Guest type	Confirm VM or LXC
Proxmox guest ID	Confirm current VM or CT ID

Architecture Role

The WireGuard gateway sits between the public VPS and the home LAN.

When an external user opens a hosted application:

1. Public DNS resolves the application hostname to the VPS.
2. The client connects to the VPS.
3. The VPS forwards the approved connection through WireGuard.
4. The WireGuard gateway receives the encrypted traffic.
5. The gateway applies forwarding and firewall rules.
6. The gateway sends the request to Traefik at 192.168.2.182.
7. Traefik routes the request to the correct internal application.

The WireGuard gateway does not host the applications.

The gateway only provides secure transport and controlled forwarding.

Primary Traffic Flow

The normal public application path includes:

Stage	Component
Public DNS	WHC cPanel
Public entry point	VPS at 38.29.213.101
Encrypted transport	WireGuard
Home tunnel endpoint	WireGuard gateway at 192.168.2.64
HTTPS routing	Traefik at 192.168.2.182

Only required application traffic should pass through this path.

WireGuard Interface

The WireGuard interface contains:

- A private tunnel address.
- A listening port.
- A private key.
- A peer definition for the VPS.
- Allowed IP routes.
- Optional startup and shutdown firewall commands.

The existing interface name, tunnel subnet, listening port, and peer address should be recorded after checking the active configuration.

Useful checks:

- `sudo wg show`
- `sudo wg showconf wg0`
- `ip -br addr`
- `ip route`

If the interface is not named `wg0`, replace `wg0` with the actual interface name.

Do not copy the private key into BookStack.

Key Management

WireGuard authentication uses public and private key pairs.

Gateway Private Key

The gateway private key:

- Must remain only on the WireGuard gateway.
- Must have restrictive filesystem permissions.
- Must not be copied into BookStack.
- Must not be sent through chat or email.
- Must be stored in an encrypted backup or password manager if recovery requires it.

Gateway Public Key

The gateway public key may be stored in documentation.

The public key identifies the gateway to the VPS peer.

VPS Public Key

The gateway configuration contains the VPS public key.

The VPS private key remains only on the VPS.

Key Rotation

If a private key is exposed:

1. Generate a new key pair on the affected peer.
2. Update the opposite peer with the new public key.
3. Restart or reload the affected WireGuard interface.
4. Confirm a new handshake.
5. Revoke the compromised key.
6. Update the encrypted recovery record.
7. Document the rotation date.

Allowed IPs

WireGuard Allowed IPs control routing and peer address validation.

The gateway and VPS must use narrow Allowed IP entries.

Allowed IPs should contain only:

- The remote tunnel address.
- Explicit home or service routes required by the design.
- Any intentionally routed management destination.

Avoid using overly broad routes unless the architecture requires them.

Do not add the full home LAN to a peer merely for convenience without reviewing the security impact.

IP Forwarding

The gateway requires IPv4 forwarding.

Check the current value with:

```
sysctl net.ipv4.ip_forward
```

Expected result:

- `net.ipv4.ip_forward = 1`

The persistent setting should be stored in a file under `/etc/sysctl.d/`.

If forwarding is disabled, the WireGuard handshake may succeed while application traffic still fails.

Network Address Translation

The gateway may use NAT when home-LAN devices do not have a route back to the WireGuard tunnel subnet.

NAT allows an internal destination to see the connection as originating from the WireGuard gateway's LAN address.

Advantages:

- No static route is required on the Bell router.

- Existing LAN devices can return traffic normally.
- Deployment is simpler.

Limitations:

- Internal applications see the gateway address rather than the original tunnel source.
- Source-based application logging becomes less detailed.
- Firewall rules on the destination must allow the gateway address.

NAT rules should be narrow and limited to required source networks, destinations, and ports.

Firewall Policy

The WireGuard gateway must permit only required tunnel traffic.

Required Traffic

Typical required traffic includes:

- WireGuard UDP traffic between the VPS and gateway.
- Forwarded HTTPS traffic to Traefik.
- Established return traffic.
- SSH administration from the home LAN.
- Selected diagnostic traffic.

Restricted Traffic

The VPS should not automatically receive unrestricted access to:

- Proxmox.
- Pi-hole administration.
- Nextcloud AIO management.
- Uptime Kuma's private backend.
- Authentik's private backend.
- Vikunja's private backend.
- BookStack's private backend.

- SSH on every home service.
- The Docker API.
- Other LAN client devices.

Each additional route should have a documented purpose.

Traefik Forwarding

The primary application destination behind the WireGuard gateway is Traefik.

Destination	Value
Traefik address	192.168.2.182
Public application port	TCP 443
Purpose	HTTPS routing to internal applications

The gateway should forward approved public HTTPS traffic to Traefik rather than directly to individual application VMs.

This keeps public routing centralized.

Proxmox Management Through the Tunnel

Proxmox management is not part of the normal public ingress path.

If emergency Proxmox access through the VPS is intentionally enabled, use a narrow route such as:

- VPS tunnel address to 192.168.2.254.
- TCP port 8006 only.
- Optional TCP port 22 using a separate rule.
- SSH local port forwarding from the remote workstation.

Do not expose Proxmox TCP 8006 directly on the VPS public interface.

The preferred long-term remote-management method is a dedicated road-warrior VPN for the administrator's laptop or phone.

Separation From a Future Remote-User VPN

The current WireGuard connection is used for VPS ingress.

A future laptop or phone VPN must use:

- A separate WireGuard interface.
- A separate UDP port.
- A separate VPN subnet.
- Separate peer keys.
- A separate firewall policy.
- Unique addresses for every device.

Do not reuse the VPS key pair or peer configuration for remote-user devices.

The VPS tunnel and a road-warrior VPN serve different purposes.

Service Startup

The WireGuard gateway should start early after a host reboot.

Preferred dependency order:

1. Proxmox VE.
2. Pi-hole.
3. WireGuard gateway.
4. Authentik.
5. Traefik.
6. Public applications.

The WireGuard service should be enabled at boot.

Check the active service using:

```
systemctl status wg-quick@wg0 --no-pager
```

If the interface name differs, replace `wg0` with the correct name.

Expected service state:

- Active.
 - Interface created.
 - Peer loaded.
 - No repeated startup errors.
-

Health Validation

Confirm the Interface

Use:

```
sudo wg show
```

Confirm:

- Interface exists.
- Listening port is present.
- VPS public key is listed.
- Endpoint is correct.
- Allowed IPs are correct.
- Latest handshake is recent.
- Transfer counters increase.

Confirm Tunnel Addressing

Use:

```
ip -br addr
```

The WireGuard interface should show the expected tunnel address.

Confirm Routes

Use:

```
ip route
```

The required tunnel and LAN routes should be present.

Confirm Forwarding

Use:

```
sysctl net.ipv4.ip_forward
```

Expected:

- Forwarding enabled.

Confirm Firewall

If UFW is used:

```
sudo ufw status numbered
```

If iptables NAT is used:

```
sudo iptables -t nat -L -n -v --line-numbers
```

If nftables is used:

```
sudo nft list ruleset
```

Review only the firewall system actually used by the gateway.

End-to-End Validation

A successful WireGuard handshake does not prove application forwarding works.

Validate the complete path.

From the VPS

Test the home WireGuard peer or tunnel gateway address.

Then test Traefik through the tunnel on the intended destination and port.

From an External Client

Use:

- Cellular data.
- A mobile hotspot.
- A remote network.

Open a public application such as:

[Open Authentik](#)

The application should load through:

- VPS.
- WireGuard.
- Gateway.
- Traefik.
- Application backend.

From the Home LAN

The same hostname should resolve through Pi-hole and bypass the VPS.

This verifies that public and private paths both work.

Monitoring

Recommended Uptime Kuma monitors include:

Monitor	Target
Gateway VM or LXC	Ping <code>192.168.2.64</code>
WireGuard tunnel peer	Tunnel address or supported monitor
Public VPS	<code>38.29.213.101</code>
Traefik through tunnel	Approved tunnel path
Public Authentik	<code>portal.bytek.ca</code>
Public Nextcloud	<code>cloud.bytek.ca</code>

Monitoring should distinguish between:

- Gateway unavailable.
 - WireGuard handshake unavailable.
 - Tunnel routing unavailable.
 - Traefik unavailable.
 - Individual application unavailable.
-

Common Failure Scenarios

Handshake Is Missing

Possible causes:

- VPS is down.
- Home internet connection is down.
- Gateway is down.
- WireGuard service is stopped.
- UDP port is blocked.
- Peer endpoint changed.
- Public key mismatch.
- Private key was changed.
- Allowed IPs are incorrect.

Handshake Exists but Applications Fail

Possible causes:

- IP forwarding is disabled.
- Forwarding firewall rule is missing.
- NAT rule is missing.
- Traefik is unavailable.
- Traefik address changed.
- Destination firewall denies the gateway.
- Return traffic follows the wrong path.
- VPS forwarding configuration is incorrect.

Internal Applications Work but Public Applications Fail

This usually indicates a problem with:

- Public DNS.
- VPS.
- WireGuard.
- Gateway forwarding.
- VPS or home-end firewalling.

One Public Application Fails

If other public applications work, the tunnel is probably healthy.

Check:

- Traefik router.
- Certificate.
- Backend address.
- Backend firewall.
- Application health.

Tunnel Stops After a Power Failure

Check startup order:

1. Confirm the gateway booted.
 2. Confirm WireGuard started.
 3. Confirm the network interface has its LAN address.
 4. Confirm the VPS peer.
 5. Confirm forwarding.
 6. Confirm firewall rules loaded.
 7. Confirm a new handshake.
 8. Confirm Traefik is available.
-

Troubleshooting Order

When public ingress fails:

1. Confirm the VPS is running.
2. Confirm the WireGuard gateway is running.
3. Confirm the gateway LAN IP is `192.168.2.64`.
4. Confirm the WireGuard interface exists.
5. Confirm the latest handshake.
6. Confirm transfer counters.
7. Confirm IPv4 forwarding.
8. Confirm forwarding firewall rules.
9. Confirm NAT rules.
10. Confirm Traefik at `192.168.2.182`.
11. Test a Traefik backend directly.
12. Test the public hostname externally.
13. Review logs before changing keys.

Do not regenerate WireGuard keys as an early troubleshooting step.

Logging

Useful log sources include:

- WireGuard service journal.

- System journal.
- UFW logs.
- Kernel network logs.
- Traefik access logs.
- VPS firewall logs.

Review the WireGuard unit using:

```
journalctl -u wg-quick@wg0 --since "30 minutes ago" --no-pager
```

Replace the interface name if necessary.

Look for:

- Interface startup failures.
 - Invalid configuration syntax.
 - Address conflicts.
 - Missing commands.
 - Firewall command failures.
 - Route conflicts.
-

Configuration Backup

Protect the following gateway data:

- WireGuard configuration.
- Private key.
- Public key.
- Peer definitions.
- Firewall rules.
- NAT rules.
- Sysctl forwarding configuration.
- SSH configuration.
- Authorized SSH keys.
- Network configuration.
- Recovery notes.

The VM or LXC is included in the Proxmox backup system.

The WireGuard private key should also exist in an encrypted recovery location.

Do not put the private key directly into BookStack.

Recovery Procedure

If the gateway VM or LXC is corrupted:

1. Use Proxmox to identify the most recent good backup.
2. Restore under a temporary guest ID if testing is required.
3. Keep the restored guest disconnected from the LAN during initial inspection.
4. Verify the WireGuard configuration and private key.
5. Verify the LAN address.
6. Verify forwarding and firewall rules.
7. Shut down the failed production guest.
8. Assign the expected production identity and network configuration.
9. Start the replacement.
10. Confirm a WireGuard handshake.
11. Confirm Traefik reachability.
12. Test an application externally.

If the key is unavailable:

1. Generate a new gateway key pair.
 2. Update the VPS peer with the new gateway public key.
 3. Preserve the VPS key.
 4. Restart the WireGuard interfaces.
 5. Confirm the new handshake.
 6. Store the new private key securely.
-

Security Checklist

- WireGuard private key is not stored in BookStack.

- VPS and gateway use unique private keys.
 - Allowed IPs are narrow.
 - Public UDP exposure is limited to the WireGuard port.
 - SSH uses keys.
 - SSH is restricted to approved sources.
 - IPv4 forwarding is enabled only because routing is required.
 - NAT rules are narrow.
 - VPS access to the LAN is limited.
 - Proxmox is not publicly reachable.
 - Pi-hole administration is not publicly reachable.
 - AIO management is not publicly reachable.
 - Docker API is not exposed.
 - Backup coverage is active.
 - Recovery keys are stored securely.
-

Validation Checklist

- Gateway is running.
 - LAN address is `192.168.2.64`.
 - WireGuard interface is active.
 - VPS peer is listed.
 - Latest handshake is recent.
 - Transfer counters increase.
 - IP forwarding is enabled.
 - Required routes exist.
 - Required firewall rules exist.
 - NAT rules are active where needed.
 - Traefik at `192.168.2.182` is reachable.
 - Public Authentik works externally.
 - Public Nextcloud works externally.
 - Internal split-DNS access still works without the VPS.
 - Proxmox backup includes the gateway.
 - WireGuard recovery material is stored securely.
-

Document Control

- **Owner:** Bryan Gagne-Plante
- **Private address:** 192.168.2.64
- **Primary role:** VPS-to-home WireGuard gateway
- **Public peer:** VPS at 38.29.213.101
- **Guest type:** Confirm VM or LXC
- **Guest ID:** Confirm current ID
- **WireGuard interface:** Confirm current interface name
- **Listen port:** Confirm current WireGuard UDP port
- **Tunnel subnet:** Confirm current tunnel subnet
- **Last verified:** YYYY-MM-DD
- **Last handshake test:** YYYY-MM-DD
- **Last external ingress test:** YYYY-MM-DD
- **Last restore test:** YYYY-MM-DD
- **Known limitation:** External application access depends on one VPS-to-home WireGuard tunnel

Revision #2

Created 2026-08-17 14:50:08 UTC by Bryan Gagne-Plante

Updated 2026-08-17 15:33:01 UTC by Bryan Gagne-Plante