

Pi-hole

Purpose

Pi-hole provides DNS services for the Bytek home network.

Pi-hole is responsible for:

- Resolving public internet hostnames.
- Blocking selected advertising and tracking domains.
- Providing split-DNS records for Bytek applications.
- Providing DHCP reservations for infrastructure guests.
- Directing internal application traffic to Traefik.
- Resolving the private Proxmox hostname.
- Providing DNS to selected Docker containers.

Pi-hole is foundational infrastructure. Pi-hole must remain accessible independently of Traefik and Authentik so DNS can be repaired during an application or identity outage.

Service Information

Setting	Value
Service name	Pi-hole
Private IP address	192.168.2.65
LAN subnet	192.168.2.0/24
DNS ports	TCP and UDP 53
Primary role	LAN DNS and split DNS
Secondary role	DHCP reservations

Management scope	Home LAN or trusted VPN
Authentication	Local Pi-hole credentials
Proxmox guest type	LXC
Proxmox guest ID	Confirm current CT ID
Backup coverage	Proxmox scheduled backup

Architecture Role

Pi-hole provides DNS to home-LAN clients and selected infrastructure containers.

When a LAN device opens a Bytek application:

1. The device asks Pi-hole to resolve the application hostname.
2. Pi-hole returns Traefik's private address, `192.168.2.182`.
3. The client connects directly to Traefik.
4. Traefik routes the request to the correct private application backend.

This allows the same hostname and HTTPS certificate to be used internally and externally.

Pi-hole is not part of the public application path for external users.

External users resolve Bytek applications through WHC public DNS instead.

DNS Responsibilities

Pi-hole provides two types of DNS resolution.

Public Internet Resolution

Pi-hole forwards normal public DNS queries to its configured upstream resolvers.

Examples include:

- Software repositories.

- Linux package mirrors.
- Let's Encrypt.
- Docker registries.
- Public websites.
- Public APIs.

Local Bytek Resolution

Pi-hole answers selected Bytek application hostnames using private addresses.

These local records create the split-DNS design.

Split-DNS Records

Hostname	Internal Destination	Purpose
cloud.bytek.ca	192.168.2.182	Nextcloud through Traefik
docs.bytek.ca	192.168.2.182	BookStack through Traefik
portal.bytek.ca	192.168.2.182	Authentik through Traefik
projects.bytek.ca	192.168.2.182	Vikunja through Traefik
proxy.bytek.ca	192.168.2.182	Traefik dashboard
status.bytek.ca	192.168.2.182	Uptime Kuma through Traefik
pve.bytek.ca	192.168.2.254	Direct Proxmox management

Application hostnames point internally to Traefik.

Proxmox is the exception because Proxmox does not sit behind Traefik.

Public and Private DNS Comparison

Hostname	Public WHC Result	Internal Pi-hole Result
cloud.bytek.ca	38.29.213.101	192.168.2.182
docs.bytek.ca	38.29.213.101	192.168.2.182
portal.bytek.ca	38.29.213.101	192.168.2.182
projects.bytek.ca	38.29.213.101	192.168.2.182
proxy.bytek.ca	38.29.213.101	192.168.2.182
status.bytek.ca	38.29.213.101	192.168.2.182
pve.bytek.ca	No public record	192.168.2.254

The split-DNS model provides:

- Lower latency for LAN clients.
- No unnecessary traffic through the VPS.
- No dependency on router hairpin NAT.
- Consistent application URLs.
- Consistent OIDC callback addresses.
- Valid HTTPS certificates on internal connections.

Proxmox DNS Exception

The Proxmox host intentionally does not use Pi-hole as its general DNS resolver.

Proxmox uses:

- **DNS resolver:** 9.9.9.9

This prevents the hypervisor from depending on a guest LXC for DNS.

As a result:

- LAN clients resolve portal.bytek.ca to 192.168.2.182.
- Proxmox normally resolves portal.bytek.ca to the VPS at 38.29.213.101.

Do not change the Proxmox resolver to Pi-hole without reviewing the dependency impact.

If Proxmox eventually needs a direct local route to Authentik, use a specific /etc/hosts entry for portal.bytek.ca instead of replacing the general resolver.

DHCP Reservations

Infrastructure guests should receive stable addresses through DHCP reservations.

Reservations are based on the virtual NIC MAC address assigned by Proxmox.

Known Reservations

Service	Reserved Address
Nextcloud AIO	192.168.2.100
Uptime Kuma	192.168.2.115
Vikunja	192.168.2.121
Authentik	192.168.2.162
Traefik	192.168.2.182
BookStack	Confirm current address
Plex	Confirm after deployment
Future game server	Confirm after deployment

The WireGuard gateway uses 192.168.2.64 .

Confirm whether the WireGuard address is configured through a reservation or static guest networking before editing it.

Why Stable Addresses Matter

Infrastructure addresses are referenced by:

- Traefik backend definitions.
- Proxmox firewall rules.
- Uptime Kuma monitors.
- Authentik Proxy Providers.
- Docker DNS settings.

- BookStack documentation.
- Administrative bookmarks.
- WireGuard forwarding rules.
- Application trusted-proxy settings.

An unexpected address change can break several services simultaneously.

Always reserve a service address before treating the address as permanent.

Finding a Guest MAC Address

For a Proxmox VM, inspect the `net0` entry using:

```
qm config <VMID>
```

For an LXC, inspect the network entry using:

```
pct config <CTID>
```

Record the MAC address in Pi-hole when creating the reservation.

Do not manually duplicate a MAC address between guests.

DNS Clients

Pi-hole may provide DNS to:

- Windows workstations.
- Linux workstations.
- Mobile devices on the LAN.
- Service VMs.
- Service LXCs.
- Selected Docker containers.
- Future remote VPN clients.
- Uptime Kuma.
- Traefik.

- Nextcloud application containers.
- BookStack.
- Vikunja.

Not every infrastructure host must use Pi-hole.

Proxmox intentionally uses Quad9.

Other critical systems may also use an independent resolver if avoiding a dependency on Pi-hole is more important than split-DNS access.

Docker DNS

Some application containers require explicit Pi-hole DNS.

This is particularly important for containers that must resolve private Bytek hostnames.

Traefik

The Traefik container explicitly uses:

- `192.168.2.65`

This was added after Docker's embedded resolver failed to resolve the Let's Encrypt ACME endpoint following a power interruption.

Uptime Kuma

The Uptime Kuma container explicitly uses:

- `192.168.2.65`

This allows Kuma to monitor private Bytek application hostnames.

BookStack

BookStack may use Pi-hole explicitly so the container can resolve Authentik at `portal.bytek.ca`.

Nextcloud

Nextcloud containers must be able to resolve `portal.bytek.ca` for Authentik OIDC.

If the host resolves a name but the container does not, inspect the container's Docker DNS configuration.

Docker DNS Validation

Confirm a container's configured DNS servers using:

```
sudo docker inspect --format='DNS={{json .HostConfig.Dns}}' <container-name>
```

Test hostname resolution from inside a container using:

```
sudo docker exec <container-name> getent hosts portal.bytek.ca
```

Expected internal result:

- `portal.bytek.ca` resolves to `192.168.2.182`.

If the command returns no output, container DNS is failing.

Do not change OIDC client IDs or secrets before DNS is proven healthy.

Browser DNS Over HTTPS

Firefox-based browsers, including Zen, can bypass Pi-hole through DNS over HTTPS.

This previously prevented access to:

- `pve.bytek.ca`

Pi-hole correctly resolved the private hostname, but Zen attempted to use protected public DNS instead.

Recommended setting:

- **DNS over HTTPS:** Off or Default Protection.

If a private hostname works by IP but not by name:

1. Check the hostname from the operating system.
2. Query Pi-hole directly.
3. Disable forced browser DNS over HTTPS.
4. Clear the operating-system DNS cache.
5. Clear the browser DNS cache.
6. Close idle browser sockets.
7. Restart the browser.

Useful Windows commands include:

- `Resolve-DnsName pve.bytek.ca`
 - `nslookup pve.bytek.ca 192.168.2.65`
 - `Clear-DnsClientCache`
 - `ipconfig /flushdns`
-

DNS Validation

Test an Internal Application Record

Use:

```
nslookup portal.bytek.ca 192.168.2.65
```

Expected result:

- `portal.bytek.ca` resolves to `192.168.2.182`.

Test Proxmox Resolution

Use:

```
nslookup pve.bytek.ca 192.168.2.65
```

Expected result:

- `pve.bytek.ca` resolves to `192.168.2.254`.

Test Public Resolution

Use:

```
nslookup google.com 192.168.2.65
```

Expected result:

- One or more public addresses are returned.

Test DNS Over TCP

Use:

```
nc -zv -w 3 192.168.2.65 53
```

Test DNS Over UDP

Use:

```
nc -zvu -w 3 192.168.2.65 53
```

Both TCP and UDP DNS should be available to approved LAN clients.

Service Health

Confirm that the Pi-hole LXC is running from Proxmox.

Confirm the guest address using:

```
ip -br -4 addr
```

Confirm DNS listeners using:

```
ss -lntup | grep ':53'
```

Confirm the Pi-hole DNS service using the appropriate service command for the installed Pi-hole version.

Review recent service logs if DNS queries fail.

Management Access

Pi-hole management should remain private.

Approved access paths include:

- Home LAN.
- Trusted remote-access VPN.
- Proxmox console.
- SSH from an approved administrative source.

Do not route Pi-hole administration through:

- The public VPS.
 - Traefik public HTTPS.
 - Authentik ForwardAuth.
 - A public WHC hostname.
 - Direct Bell router port forwarding.
-

Why Pi-hole Is Not Behind Authentik

Pi-hole is part of the dependency chain for internal Authentik and application resolution.

If Pi-hole management depended on Authentik, the following circular failure could occur:

1. Pi-hole DNS fails.
2. Authentik's hostname stops resolving internally.
3. Authentik login fails.
4. Pi-hole administration is protected by Authentik.
5. The administrator cannot reach Pi-hole to repair DNS.

Pi-hole therefore uses local credentials and private network access.

Upstream DNS

Pi-hole forwards non-local queries to configured upstream resolvers.

The chosen upstream resolver configuration should be documented after verification.

When changing upstream DNS:

1. Record the current configuration.
2. Change only one resolver setting at a time.
3. Test public DNS.
4. Test internal local records.
5. Test DNSSEC if enabled.
6. Test IPv4 and IPv6 behavior separately.
7. Confirm application containers still resolve external services.

Do not assume that an upstream failure means Pi-hole itself is unavailable.

IPv6 Considerations

The Bytek homelab mainly uses IPv4 addressing.

If IPv6 is not intentionally configured end to end:

- Avoid publishing unintended AAAA records.
- Verify clients are not receiving an unusable IPv6 DNS path.
- Check browser connection behavior.
- Check whether applications prefer IPv6 before IPv4.
- Document any intentional IPv6 disabling.

An incorrect public AAAA record can cause clients to attempt an unavailable path before falling back to IPv4.

Firewall Policy

Pi-hole must accept DNS from approved clients.

Typical requirements include:

- LAN clients to TCP 53.
- LAN clients to UDP 53.
- Approved Docker hosts to TCP and UDP 53.
- Future VPN clients to TCP and UDP 53.
- Administrative SSH from approved LAN sources.
- Private management access from the LAN.

Avoid exposing DNS publicly.

A publicly accessible recursive DNS resolver can be abused.

The Bell router must not forward public TCP or UDP port 53 to Pi-hole.

Monitoring

Recommended Uptime Kuma monitors include:

Monitor	Target
Pi-hole guest	Ping <code>192.168.2.65</code>
Pi-hole DNS UDP	<code>192.168.2.65:53</code>
Pi-hole DNS TCP	<code>192.168.2.65:53</code>
Internal Authentik record	<code>portal.bytek.ca</code>
Internal Proxmox record	<code>pve.bytek.ca</code>
Public DNS resolution	Selected external hostname

Monitoring should distinguish between:

- Guest unavailable.
- DNS service unavailable.

- Internal records unavailable.
 - Public forwarding unavailable.
 - Upstream DNS unavailable.
-

Common Failure Scenarios

Pi-hole Guest Is Down

Symptoms:

- Internal hostnames fail.
- Containers using Pi-hole cannot resolve names.
- Uptime Kuma reports several services down.
- Browser applications may fail internally.

Check:

- Proxmox guest status.
- LXC startup.
- Guest network address.
- Pi-hole service.
- Backup availability.

Public DNS Fails but Local Records Work

Possible causes:

- Upstream resolver unavailable.
- Internet outage.
- DNSSEC issue.
- Firewall blocking outbound DNS.
- Incorrect upstream configuration.

Local Records Fail but Public DNS Works

Possible causes:

- Local record deleted.
- Typographical error.
- Different Pi-hole instance answering.
- Client using another DNS server.
- Browser DNS over HTTPS.
- Cached stale record.

VM Host Resolves but Container Does Not

Possible causes:

- Docker embedded DNS issue.
- Container started before Pi-hole.
- Explicit Docker DNS is missing.
- Docker network is unhealthy.
- Guest firewall blocks DNS.

Recovery may require recreating the container after Pi-hole is working.

One Host Uses the Wrong Address

Possible causes:

- Client DNS cache.
 - Browser DNS cache.
 - Forced DNS over HTTPS.
 - Public resolver used instead of Pi-hole.
 - Hosts-file override.
 - Multiple active network adapters.
 - VPN DNS override.
-

Power-Failure Recovery

After an electrical outage, Pi-hole should start before applications that depend on split DNS.

Recommended validation order:

1. Confirm Proxmox is running.
2. Confirm the Pi-hole LXC started.
3. Confirm Pi-hole owns `192.168.2.65`.
4. Confirm TCP and UDP port 53.
5. Confirm one local record.
6. Confirm one public hostname.
7. Start or validate Traefik.
8. Validate Authentik.
9. Restart affected containers whose DNS remained broken.
10. Confirm Uptime Kuma monitors recover.

Application containers may need to be recreated or restarted if Docker cached a broken DNS path during startup.

Backup

The Pi-hole LXC is included in the Proxmox scheduled backup job.

The backup should preserve:

- Pi-hole configuration.
- Local DNS records.
- DHCP reservations.
- Adlists.
- Allow and deny lists.
- Custom DNS settings.
- Network configuration.
- Administrative configuration.

A Pi-hole configuration export may be maintained in addition to the Proxmox backup.

The export should be stored securely because it may contain internal network details.

Recovery Procedure

If Pi-hole is unavailable:

1. Open Proxmox using `192.168.2.254` if private DNS is unavailable.
2. Confirm the Pi-hole LXC is running.
3. Open the LXC console.
4. Confirm the interface owns `192.168.2.65`.
5. Confirm the default route.
6. Confirm TCP and UDP port 53.
7. Confirm the Pi-hole DNS service.
8. Test a local record from inside the LXC.
9. Test a public record.
10. Review logs.
11. Restore from a Proxmox backup if the configuration is damaged.

If Pi-hole is restored under a temporary CT ID:

- Keep the restored container disconnected from the live LAN initially.
 - Verify the configuration.
 - Do not run two Pi-hole instances with the same IP.
 - Shut down the failed instance before activating the restored one.
-

Address-Change Procedure

Changing Pi-hole's address would affect most of the environment.

Before changing `192.168.2.65`:

1. Document the current state.
2. Update the Bell router or DHCP DNS settings.
3. Update Docker Compose files with explicit DNS.
4. Update Uptime Kuma.
5. Update future VPN client DNS.
6. Update firewall rules.
7. Update service documentation.
8. Flush client DNS caches.
9. Recreate containers that use explicit Docker DNS.
10. Test all split-DNS applications.

Avoid changing Pi-hole's address unless required.

Security Checklist

- Pi-hole management is LAN or VPN only.
 - Public TCP and UDP port 53 are not forwarded.
 - Local administrator credentials are stored securely.
 - SSH uses key-based authentication.
 - DNS access is limited to approved networks.
 - Local DNS records contain only intended hostnames.
 - Proxmox does not depend on Pi-hole.
 - Pi-hole is not behind Authentik.
 - Backups include the Pi-hole LXC.
 - A configuration export exists or is planned.
 - Browser DNS over HTTPS behavior is documented.
 - Upstream DNS settings are documented.
 - Recovery by private IP has been tested.
-

Validation Checklist

- Pi-hole LXC is running.
- Pi-hole owns `192.168.2.65`.
- TCP port 53 responds.
- UDP port 53 responds.
- Public names resolve.
- Internal hostnames resolve to Traefik.
- `pve.bytek.ca` resolves directly to Proxmox.
- No public WHC record exists for `pve.bytek.ca`.
- Traefik containers can use Pi-hole.
- Uptime Kuma can use Pi-hole.
- Nextcloud can resolve Authentik.
- BookStack can resolve Authentik.

- Vikunja can resolve Authentik.
 - DNS over HTTPS is not bypassing Pi-hole on administrative browsers.
 - Proxmox backup includes Pi-hole.
-

Document Control

- **Owner:** Bryan Gagne-Plante
 - **Private address:** 192.168.2.65
 - **Primary role:** LAN DNS and split DNS
 - **Secondary role:** DHCP reservations
 - **Guest type:** LXC
 - **Guest ID:** Confirm current CT ID
 - **Management scope:** LAN or trusted VPN
 - **Upstream resolvers:** Confirm current configuration
 - **Last verified:** YYYY-MM-DD
 - **Last DNS recovery test:** YYYY-MM-DD
 - **Last backup test:** YYYY-MM-DD
 - **Last configuration export:** YYYY-MM-DD
 - **Known limitations:** Internal application resolution depends on one Pi-hole instance
-

Revision #2

Created 2026-08-17 14:50:24 UTC by Bryan Gagne-Plante

Updated 2026-08-18 00:20:08 UTC by Bryan Gagne-Plante