

Service Startup Order

Purpose

This page documents the preferred startup and validation order after:

- A power interruption.
- A planned Proxmox shutdown.
- A Proxmox reboot.
- Network maintenance.
- Storage maintenance.
- A full-stack restart.

Starting services in dependency order reduces DNS, OIDC, proxy, and monitoring failures.

Preferred Startup Order

Order	Service
1	Proxmox VE
2	Pi-hole
3	WireGuard Gateway
4	Authentik database and Redis
5	Authentik server and worker
6	Traefik
7	Uptime Kuma
8	Nextcloud AIO
9	Vikunja

10	BookStack
11	Plex, when deployed
12	Game server, when deployed
13	Navidrome, when deployed

1. Proxmox VE

Confirm that the hypervisor completed booting.

Validate:

- Management IP is `192.168.2.254`.
- Network bridge is active.
- Storage pools are active.
- `pveproxy` is active.
- TCP port 8006 is listening.
- Backup HDD is mounted.

Useful checks:

- `systemctl is-active pveproxy`
- `ss -lntp | grep ':8006'`
- `pvesm status`
- `findmnt /mnt/pve/pve-backup`

Do not start troubleshooting application guests until required Proxmox storage is active.

2. Pi-hole

Pi-hole should start before services that require split DNS.

Validate:

- Pi-hole owns `192.168.2.65`.

- TCP port 53 responds.
- UDP port 53 responds.
- Internal records resolve.
- Public records resolve.

Test:

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

Expected result:

- 192.168.2.182

Test:

```
nslookup google.com 192.168.2.65
```

Expected result:

- A public address.

Containers that started before Pi-hole may need to be restarted after Pi-hole becomes healthy.

3. WireGuard Gateway

Validate:

- Gateway owns 192.168.2.64.
- WireGuard interface exists.
- VPS peer appears.
- Recent handshake exists.
- Transfer counters increase.
- IPv4 forwarding is enabled.
- Firewall and NAT rules loaded.

Useful checks:

- `sudo wg show`
- `sysctl net.ipv4.ip_forward`
- `ip route`

- `sudo ufw status numbered`

The WireGuard tunnel must be working before external application access can recover.

4. Authentik Database and Redis

Authentik depends on PostgreSQL and Redis.

Validate the Authentik Compose stack using:

```
sudo docker compose ps
```

Confirm:

- PostgreSQL is running.
- Redis is running.
- Persistent data is available.
- No storage or permission errors appear.

Do not treat Authentik as ready merely because the server container has started.

5. Authentik Server and Worker

After PostgreSQL and Redis are healthy, validate:

- Authentik server.
- Authentik worker.
- Direct readiness endpoint.
- OIDC discovery endpoint.
- Embedded outpost.

Direct readiness should return HTTP 200.

The outpost ping should return HTTP 204.

OIDC-dependent applications should not be treated as healthy until Authentik is ready.

6. Traefik

Traefik depends on:

- Pi-hole for its configured Docker DNS.
- Authentik for ForwardAuth and Proxy Providers.
- WireGuard for public ingress.

Validate:

- Traefik VM owns `192.168.2.182`.
- Docker is active.
- Traefik container is running.
- TCP port 443 is listening.
- Pi-hole DNS works inside the container.
- Dynamic routers loaded.
- Certificates are available.
- Authentik route works.

Useful checks:

- `sudo docker compose ps`
- `sudo docker inspect --format='DNS={{json .HostConfig.Dns}}' traefik`
- `sudo tail -n 100 /opt/traefik/logs/traefik.log`

If Traefik started before Pi-hole and DNS remains broken, recreate the Traefik container after Pi-hole is healthy.

7. Uptime Kuma

Start monitoring after DNS, identity, and routing are operational.

Validate:

- VM owns `192.168.2.115`.

- Docker is active.
- Container is running.
- Container health is healthy.
- Embedded database is available.
- Container DNS resolves internal hostnames.
- Dashboard route works.
- Monitors begin recovering.

Do not immediately modify every failed monitor after a reboot. Allow dependencies to recover first.

8. Nextcloud AIO

Before starting the Nextcloud application stack, validate:

- VM owns `192.168.2.100`.
- 500 GB data disk is attached.
- `/mnt/nextcloud-data` is mounted.
- `/mnt/nextcloud-data/ncdata` exists.
- Docker is active.
- AIO management is reachable.

Then confirm:

- PostgreSQL.
- Redis.
- Nextcloud.
- Apache.
- Notify Push.
- Collabora.

Test:

- TCP port 11000.
- `cloud.bytek.ca`.
- Authentik login.
- File access.

Do not run Nextcloud normally if the data disk is not mounted.

9. Vikunja

Validate:

- VM owns `192.168.2.121`.
- Docker is active.
- PostgreSQL is healthy.
- Vikunja container is running.
- `/health` returns HTTP 200.
- Authentik login works.

If the container cannot reach Authentik, confirm DNS before changing OIDC credentials.

10. BookStack

Validate:

- VM owns its reserved private address.
- Docker is active.
- MariaDB is running.
- BookStack is running.
- Application key is available.
- Direct backend responds.
- `docs.bytek.ca` works.
- Authentik login works.
- Existing pages open.

If OIDC fails, temporarily restore standard authentication rather than rebuilding BookStack.

Future Plex Startup

When Plex is deployed, validate:

- Plex VM boots.
- RTX 2060 is present.
- NVIDIA driver loads.
- `nvidia-smi` works.
- Docker has GPU access.
- Plex configuration storage is mounted.
- Media storage is mounted.
- Plex web interface responds.
- Hardware transcoding is available.

Do not start Plex if expected media storage is missing.

Future Game-Server Startup

When a game server is deployed, validate:

- VM owns its reserved address.
- Game-data disk is mounted.
- SteamCMD or LinuxGSM files exist.
- Game server starts.
- Required game ports listen.
- World or save data loads.
- Public game-port forwarding works.
- Monitoring works.

Do not expose RCON or management ports publicly without additional protection.

Docker DNS Recovery

A recurring risk after a full power loss is that containers start before Pi-hole is ready.

Symptoms include:

- OIDC discovery endpoint unreachable.
- Let's Encrypt DNS lookup failures.
- Internal hostnames failing inside containers.
- Uptime Kuma monitors remaining down.
- Host DNS works while container DNS fails.

Recovery process:

1. Confirm Pi-hole.
2. Confirm DNS from the VM host.
3. Confirm DNS from the affected container.
4. Check the container's explicit DNS configuration.
5. Restart the affected container.
6. Recreate the container if Docker-level DNS configuration changed.
7. Retry the failed operation.

Do not regenerate application secrets because of a DNS-only failure.

Full Validation Checklist

- Proxmox web interface works.
- Required Proxmox storage is active.
- Backup HDD is mounted.
- Pi-hole resolves internal records.
- Pi-hole resolves public records.
- WireGuard handshake exists.
- Public VPS ingress works.
- Authentik direct readiness returns HTTP 200.
- Authentik routed readiness returns HTTP 200.
- Authentik outpost returns HTTP 204.
- Traefik routers loaded.
- Traefik certificates are available.
- Uptime Kuma is healthy.
- Nextcloud status is healthy.
- Nextcloud data disk is mounted.

- Collabora works.
 - Vikunja health returns HTTP 200.
 - BookStack opens existing documentation.
 - OIDC login works on all configured applications.
 - External access works from cellular or another remote network.
-

Document Control

- **Owner:** Bryan Gagne-Plante
 - **Startup authority:** Proxmox VE
 - **DNS dependency:** Pi-hole
 - **Public-ingress dependency:** VPS and WireGuard
 - **Authentication dependency:** Authentik
 - **HTTPS dependency:** Traefik
 - **Last verified:** YYYY-MM-DD
 - **Last power-failure recovery:** YYYY-MM-DD
 - **Last full-stack restart:** YYYY-MM-DD
 - **Known issue:** Containers may require restart or recreation if started before Pi-hole
-

Revision #2

Created 2026-08-17 14:52:45 UTC by Bryan Gagne-Plante

Updated 2026-08-18 00:46:19 UTC by Bryan Gagne-Plante