# Reverse Proxy (Caddy)

# Reverse Proxy (Caddy)

Caddy serves as the primary ingress point for all web services in the Flora Family homelab. It handles SSL termination (ACME), local CA management, and request routing.

## □□ Structure

- **Location:** `/srv/caddy`
- **User:** `51004:61004` (Caddy Service Account)
- **Snippets Path:** `/main/appdata/caddy/files/snippets` (Mapped internally to `/etc/caddy/snippets`)

## □□ Log Exporting

- **Mechanism:** UDP/514 forwarding via `rsyslog` on `pluto`.
- **Path:** Raw JSON logs exported to `/var/log/caddy/access.log`.
- **Fail2Ban:** Integrated via local file watcher (jail: `caddy`).

## □□ Caddyfile Strategy

The Caddyfile is organized into modular sites located in `sites-enabled/`. This mimics traditional Apache/Nginx structures for clean management.

### Common Snippets

1. **`security_headers.caddy`**: Implements HSTS, X-Frame-Options (DENY), and nosniff.
2. **`common_tls_internal.caddy`**: Uses `tls internal` for services only accessible on the LAN, backed by the local Caddy CA.
3. **`common_tls_external.caddy`**: Standard ACME/Let's Encrypt for public-facing services.

# □□ Internal CA & Trust

When using `tls internal`, Caddy acts as its own Certificate Authority.

- **Root Cert Location:** `/main/appdata/caddy/data/caddy/pki/authorities/local/root.crt`
- **Permissions:** Access to the `pki` directory requires membership in the **Caddy Group (GID 61004)**.
- **Distribution:** This root certificate is mounted into downstream containers (like SearXNG or OpenWebUI) and added to their system trust stores to allow secure internal communication.

# □□ Typical Route Block

```
searxng.flora.family {
    reverse_proxy searxng:8080
    import /etc/caddy/snippets/security_headers.caddy
    import /etc/caddy/snippets/common_tls_internal.caddy
}
```

---

Revision #3
Created 2026-02-09 14:40:47 UTC by Chris
Updated 2026-03-18 20:34:13 UTC by Flobot