Set up subdomain routing NGINX

Dynamic Subdomain Routing for *.qool.ovh (Nginx)

0. Introduction

Goal

Automatically route requests for `<subdomain>.qool.ovh` to the corresponding directory `/var/www/qool/<subdomain>`, without creating a new DNS record or Nginx config for each site.

Use case

- `test.qool.ovh` → `/var/www/qool/test`
- `blog.qool.ovh` → `/var/www/qool/blog`
- `demo.qool.ovh` → `/var/www/qool/demo`

---

1. DNS Configuration (Wildcard)

Create wildcard DNS record

At your DNS provider (OVH), create the following record:

```
Type: A
Name: *.qool.ovh
Value: <SERVER_IP>
TTL: Auto
```

This ensures that **all subdomains** of `qool.ovh` resolve to your server.

---

2. Directory Structure

Base webroot layout

```
/var/www/qool/
 ├─ test/
 │    └─ index.php
 ├─ blog/
 │    └─ index.php
 └─ demo/
      └─ index.php
```

Ownership and permissions

```
sudo chown -R max:www-data /var/www/qool
find /var/www/qool -type d -exec chmod 750 {} \;
find /var/www/qool -type f -exec chmod 640 {} \;
```

---

3. Nginx Wildcard Configuration

Create Nginx site file

```
sudo nano /etc/nginx/sites-available/qool.ovh
```

Wildcard server block

```
server {
    listen 80;

    # Capture subdomain name dynamically
    server_name ~^(?<site>[a-z0-9-]+)\.qool\.ovh$;

    root /var/www/qool/$site;
    index index.php index.html;

    # Reject non-existing site directories
    if (!-d /var/www/qool/$site) {
```

```
        return 404;
    }

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php8.3-fpm.sock;
    }
}
```

Enable the site

```
sudo ln -s /etc/nginx/sites-available/qool.ovh /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl reload nginx
```

---

4. How the Routing Works

Nginx logic

- Nginx matches `<subdomain>.qool.ovh` using a regex
- The subdomain is captured into variable `$site`
- The document root is set to `/var/www/qool/$site`
- If the directory does not exist, a **404** is returned

Example

Request:

```
GET http://test.qool.ovh
```

Results in:

```
root = /var/www/qool/test
```

---

5. Testing

Create a test file

```
nano /var/www/qool/test/index.php
```

```
<?php
echo "Site: test.qool.ovh";
```

Test locally

```
curl -H "Host: test.qool.ovh" http://localhost
```

Test in browser

Open:

```
http://test.qool.ovh
```

---

6. Notes & Best Practices

- No per-site Nginx configuration required
- Adding a new site = create a directory only
- Works with PHP, static files, and frameworks
- Compatible with wildcard TLS certificates
- Production-safe when directory existence is validated

Final Summary

- Wildcard DNS resolves all subdomains
- Nginx regex captures subdomain name
- Filesystem routing maps subdomain → directory
- Scales to unlimited sites with zero config changes

Revision #1
Created 2026-02-09 20:48:30 UTC by Max
Updated 2026-02-09 21:05:40 UTC by Max