

Serverrapport

everything-sune.dev

Opsætning af dedikeret hjemmeserver
med Cloudflare Tunnel og Nginx

Udarbejdet af: suneg

Studie: Software Engineering, 1. år

Universitet: Fróðskaparsetur Føroya

Dato: 17. maj 2026

Indholdsfortegnelse

- 1. **Introduktion****
- 2. **Systemarkitektur****
 - 2.1 Overordnet arkitektur
 - 2.2 Trafikflow
- 3. **Teknisk beskrivelse****
 - 3.1 Hardware
 - 3.2 Operativsystem
 - 3.3 Webserver (Nginx)
 - 3.4 Cloudflare Tunnel
 - 3.5 Firewall
- 4. **Udfordringer og løsninger****
- 5. **Resultat****
- 6. **Konklusion og fremtidigt arbejde****

1. Introduktion

Dette dokument beskriver opsætningen af en dedikeret hjemmeserver til hosting af det personlige domæne everything-sune.dev. Serveren fungerer som en samlet indgang til mine softwareprojekter – både som læringsplatform, portfolio til fremtidige arbejdsgivere, og som demonstrationsmiljø for projekter fra studiet.

Formål

Målet med projektet er at:

- Opsætte en fysisk server der kører 24/7 uden skærm eller tastatur (headless)
- Gøre webprojekter tilgængelige på internettet via eget domæne
- Lære om Linux-serveradministration, netværksopsætning og deployment
- Skabe en skalerbar platform hvor nye projekter let kan tilføjes

Motivation

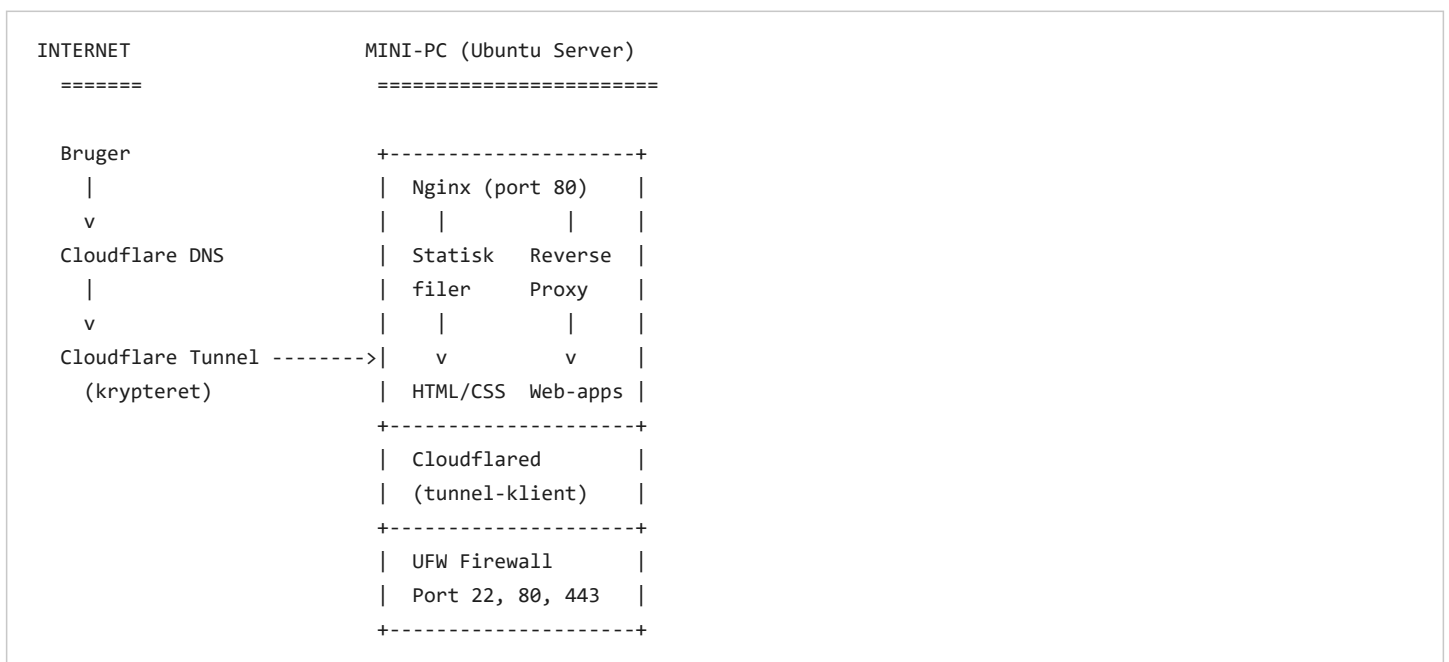
Som softwareingeniørrelev har jeg brug for et sted at fremvise mine projekter. I stedet for at bruge en cloud-tjeneste (som koster penge månedligt) valgte jeg at opsætte min egen server på en mini-PC. Dette giver både læring og fuld kontrol over infrastrukturen. Løsningen bruger Cloudflare Tunnel, hvilket eliminerer behovet for en fast IP-adresse (som koster 150 kr/md hos Føroya Tele).

2. Systemarkitektur

2.1 Overordnet arkitektur

Systemet består af tre hoveddele: en mini-PC (serveren), Cloudflares netværk (DNS og tunnel), og brugerens browser. Mini-PC'en kører Nginx som webserver og kan hoste både statiske sites og dynamiske webapplikationer.

Arkitekturoverblik:



2.2 Trafikflow

Når en bruger besøger everything-sune.dev sker følgende:

1. Browseren slår domænet op via DNS og får Cloudflares IP-adresse
2. Cloudflare modtager forespørgslen og sender den via en krypteret tunnel til mini-PC'en (udgående forbindelse fra serveren – ingen åbne porte)
3. Cloudflared-klienten på serveren modtager forespørgslen
4. Nginx router baseret på hostname og serverer det relevante indhold
5. Svaret sendes tilbage samme vej til brugeren

Fordelen ved Cloudflare Tunnel er at serveren aldrig eksponerer sin IP-adresse direkte. Al trafik går gennem Cloudflare, som også tilbyder DDoS-beskyttelse (Distributed Denial of Service) og automatisk HTTPS-kryptering.

3. Teknisk beskrivelse

3.1 Hardware

Komponent	Detalje
Model	M2 plus-16G/512GB/N100-PG
Processor	Intel N100 (4 kerner, op til 3,4 GHz)
RAM	16 GB DDR4
Lager	AOSENKE SSD 512 GB
Netværk	WiFi + Ethernet (RJ45)
Formfaktor	Mini-PC (ca. 12×12×5 cm)

Mini-PC'en kører headless (uden skærm og tastatur). Al administration foregår via SSH (Secure Shell) fra en anden computer på netværket.

3.2 Operativsystem

Komponent	Detalje
OS	Ubuntu Server 26.04 LTS
Kerne	Linux 7.0
LTS	Long Term Support – opdateringer til 2031

Ubuntu Server blev valgt frem for Windows fordi:

- Lavere ressourceforbrug (ingen grafisk brugerflade)
- Industristandard for webservere
- Bedre sikkerhed til en interneteksponeret server
- Gratis og open source

Installationen blev foretaget via et bootbart USB-drev oprettet med Rufus. OpenSSH-server blev aktiveret under installationen for fjernadgang.

3.3 Webserver (Nginx)

Nginx (udtales "engine-x") er en højtydende webserver og reverse proxy. Den håndterer to typer opgaver på serveren:

1. Serverer statiske filer (HTML/CSS/JS) direkte
2. Fungerer som reverse proxy for dynamiske webapplikationer

En reverse proxy modtager forespørgsler udefra og videresender dem til et internt program. Brugeren ser kun Nginx, ikke det bagvedliggende program. Dette giver bedre sikkerhed og fleksibilitet.

Eksempel på Nginx-konfiguration for et statisk site:

```

server {
    listen 80 default_server;
    listen [::]:80 default_server;

    root /var/www/everything-sune.dev;
    index index.html;
    server_name everything-sune.dev;

    location / {
        try_files $uri $uri/ =404;
    }
}

```

Hvert projekt på serveren får sin egen Nginx-konfiguration og sit eget subdomæne (f.eks. projekt.everything-sune.dev). Nye projekter kan tilføjes uden at påvirke eksisterende sites.

3.4 Cloudflare Tunnel

Cloudflare Tunnel er en tjeneste der forbinder en server til Cloudflares netværk uden at åbne indgående porte i routeren. Programmet cloudflared opretter en udgående, krypteret forbindelse fra serveren til Cloudflares edge-servere.

Tunnel-konfiguration:

```

tunnel: 08d6380f-8a3c-43da-a4b4-506a52cdef28
credentials-file: /etc/cloudflared/credentials.json

ingress:
- hostname: everything-sune.dev
  service: http://localhost:80
- hostname: *.everything-sune.dev
  service: http://localhost:80
- service: http_status:404

```

Ingress-reglerne definerer hvilke domæner der routes til hvilke services. Alle domæner peges til Nginx på port 80, som så router videre baseret på hostname. Den sidste regel (http_status:404) er et catch-all for ukendte domæner.

Cloudflared kører som en systemd-service – Ubuntu's system til at styre baggrundsprogrammer. Det betyder at den starter automatisk når serveren bootes og genstarter automatisk ved fejl.

3.5 Firewall (UFW)

UFW (Uncomplicated Firewall) er aktiveret med følgende regler:

Port	Formål
22/tcp (SSH)	Fjernadministration via terminal
80/tcp (HTTP)	Webserver-trafik
443/tcp (HTTPS)	Krypteret webserver-trafik
Alt andet	BLOKERET (deny incoming)

Alle indgående forbindelser der ikke matcher reglerne ovenfor blokeres. Udgående trafik er tilladt (nødvendigt for

tunnel-forbindelsen og systemopdateringer).

4. Udfordringer og løsninger

Fast IP-adresse (150 kr/md)

Føroya Tele kræver 150 kr/md for en statisk IP-adresse, hvilket er ca. 4–5 gange dyrere end tilsvarende tjenester i Danmark og Tyskland. Løsningen blev Cloudflare Tunnel, som eliminerer behovet for en fast IP helt. Tunnelen opretter en udgående forbindelse, så serverens IP-adresse er irrelevant.

Headless installation uden skærm

Mini-PC'en havde ingen dedikeret skærm. Under Ubuntu-installationen blev et TV brugt som midlertidig skærm via HDMI. Skærmens opløsning matchede ikke installerens layout, så dele af menuen var uden for synsfeltet. Dette blev løst ved at navigere med tastaturet (Tab, piletaster, Enter) baseret på kendskab til installerens menustruktur.

Fjernadministration via SSH

For at undgå afhængigheden af skærm og tastatur blev SSH-nøglebaseret autentificering sat op. En SSH-nøgle (ed25519) blev genereret på hovedcomputeren og tilføjet til serverens `authorized_keys`-fil. Derefter kan alle kommandoer udføres fra en anden computer – enten manuelt via en terminal eller automatiseret via Python-scripts.

5. Resultat

Serveren er fuldt operationel og tilgængelig på internettet via domænet everything-sune.dev. Platformen understøtter både statiske websites og dynamiske webapplikationer.

Services på serveren

Service	Status
Nginx (webserver)	Aktiv, autostart ved boot
Cloudflared (tunnel)	Aktiv, autostart ved boot
UFW (firewall)	Aktiv, autostart ved boot
OpenSSH (fjernadgang)	Aktiv, autostart ved boot

Alle services kører som systemd-services og starter automatisk ved boot. Serveren kan administreres fuldstændigt via SSH fra enhver computer på netværket.

Tilføjelse af nye projekter

Platformen er designet så nye projekter let kan tilføjes som subdomæner:

1. Opret en mappe med projektets filer på serveren
2. Tilføj en Nginx-konfiguration for det nye subdomæne
3. Tilføj en DNS-route i Cloudflare
4. Genstart de relevante services

Hvert projekt kører isoleret og påvirker ikke de andre.

6. Konklusion og fremtidigt arbejde

Konklusion

Projektet har resulteret i en fuldt funktionel hjemmeserver der hoster et personligt domæne med mulighed for flere webprojekter. Løsningen undgår løbende udgifter til cloud-hosting og fast IP-adresse ved at bruge Cloudflare Tunnel (gratis). De eneste udgifter er domænenavnet (ca. 80 kr/år) og strøm til mini-PC'en.

Opsætningen har givet praktisk erfaring med:

- Linux-serveradministration (Ubuntu Server)
- Webserver-konfiguration (Nginx)
- Netværkssikkerhed (firewall, SSH, tunneling)
- DNS og domænehåndtering (Cloudflare)
- Systemd service management
- Deployment af webapplikationer

Fremtidigt arbejde

- Konfigurere BIOS til automatisk genstart efter strømafbrydelse
- Opsætte automatiske sikkerhedsopdateringer (unattended-upgrades)
- Bygge det egentlige portfolio-site med projektbeskrivelser
- Tilføje flere projekter fra studiet som subdomæner
- Implementere automatisk backup af databaser
- Overveje monitorering (f.eks. Uptime Kuma) for at overvåge server-status
- Flytte serveren til Færøerne når den permanente placering er klar