**Engineer**

# Offentlige tjenester på en liten server

Driftet selskapets offentlige tjenester på en vert med én kjerne og 512 MB — en git-forge, en webserver for sju domener, Tor, to servere for alternative protokoller og utestenging av inntrengere — med 464 MB brukbart minne som designbegrensning; backupautomatisering er en separat implementasjon.

september 2026

## 512 MB

som et helt selskaps drift kjører på

**Situasjon.** Selskapets produksjonsvert er en skyinstans med én kjerne, 512 MB minne, hvorav rundt 464 MB er brukbart, og 10 GB disk. Alt virksomheten kjører offentlig, ligger på den: webserveren som terminerer TLS for sju domener, git-forgen, en Tor-onion-tjeneste, en Gemini-server, en Gopher-server, utestenging av inntrengere og separat implementert backupautomatisering. Den vanlige reaksjonen på den listen er å kjøpe en større maskin.

**Oppgave.** Begrensningen måtte behandles som et arkitektonisk premiss i stedet for et problem å bruke penger på, fordi det ærlige spørsmålet ikke var om en større maskin ville fungere, men om designet trengte en.

**Handling.** Minne ble argumentet som avgjorde beslutninger. Det finnes ingen overvåkingsagent, ingen metrikkpipeline og intet dashbord — rapportering er pull-basert, sju kommandoer som leser verten og gjengir Markdown, endrer ingenting og bare kjører når noen ber om det. Tjenestestyreren er systemd i stedet for enda en prosessbehandler lagt oppå den, og containerlaget er Quadlet-enheter under den samme tjenestestyreren i stedet for en daemon med sin egen. Plattformer med webpanel ble utelukket allerede på designstadiet av samme grunn. Da spørsmålet kom opp om verten kunne bære en onion-tjeneste, kom svaret fra en dags målte stikkprøver i stedet for fra en mening: tilgjengelig minne falt aldri under rundt 310 MB av 464, swap lå på 2,6 prosent, og prosessoren var 99,7 prosent ledig.

**Resultat.** Det dokumenterte oppsettet av offentlige tjenester kjørte på en maskin som kostet mindre per måned enn en lunsj, og designet er bedre takket være disiplinen, ikke bare billigere. Prisen er at det ikke finnes margin for slurv — e-post ligger med vilje ikke på denne maskinen i det hele tatt — den er skrevet som et installasjonsskjelett som venter på en egen vert, fordi en e-postserver trenger margin som denne maskinen allerede har brukt opp, og det er skrevet ned i stedet for å bli oppdaget senere.

### Bruk erfaringen i prosjektet deres

Bruk «Offentlige tjenester på en liten server» som utgangspunkt for en lignende utfordring i organisasjonen deres. Tjenestene bak arbeidet er tilgjengelige for prosjektet deres. Formen på et system tegnet opp på papiret først, slik at delene passer sammen og kostnaden er kjent før noen skriver kode.

**Se tjenestene våre** Ta kontakt →

### Se bak resultatet

Les om situasjonen, beslutningene og resultatet i eksemplet. Vi kan drøfte hva som passer deres rammer, og hva som krever en annen tilnærming. Offentlige tjenester på en liten server — 512 MB som et helt selskaps drift kjører på

**Se tjenestene våre** Ta kontakt →

## Betaler du for kapasitet du aldri bruker?

Snakk om prosjektet med ingeniøren som har ansvaret for leveransen. Vi avtaler omfang, prisgrunnlag og overlevering skriftlig før arbeidet begynner.

Rolle i oppdraget: Site Reliability Engineer. Det bygger på Plattformarkitektur, Løsningsarkitektur, Infrastruktur, Pålitelighet og sikkerhetskopier, Linux og servere, Ytelsesoptimalisering og Systemadministrasjon. Og nå som dere er her: vi er ganske gode på Plattform- og løsningsarkitektur, Skyinfrastruktur og migrering, Infrastruktur som kode og Systemadministrasjon.

**Ta kontakt** Se tjenestene våre

Ett stykke av **en lengre oversikt** — og den ligger i sin helhet her.

Hver oppføring er skrevet på samme måte: situasjonen, oppgaven, hva vi gjorde, og hva som endret seg.

**Se hele oversikten**

Definer forbedringen dere trenger

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

**Se hva det koster** Ta kontakt →

# Betaler du for kapasitet du aldri bruker?

Spør om beslutningene bak «Offentlige tjenester på en liten server». Dere snakker med folk som kan forklare arbeidet og vurdere deres eget prosjekt.

**Se tjenestene våre**

welcome@engineer.company +45 50 10 61 84 København, Danmark

Ta med deg denne siden

Visittkort

**Laget med kjærlighet av D.**

**CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N**

**Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.**

**Skrevet ut fra https://engineer.company/nb/portfolio/a-whole-company-on-one-small-server/**