**Engineer**

# Tor

🧅 et nettverk som ruter trafikk gjennom frivilliges reléer

## Hva det er

Tor skjuler hvem som snakker med hvem for alle som lytter på linjen, ved å sende hver forbindelse gjennom en kjede av maskiner der ingen enkelt maskin kjenner både starten og slutten. Kjeden velges av klienten, et program som heter tor og kjører på leserens egen datamaskin, og den er som regel tre reléer lang: en vakt (*guard*) som klienten bruker om igjen som inngang, et mellomrelé og et utgangsrelé (*exit*) som tar den siste forbindelsen til målet. Klienten bygger denne kretsen ett ledd om gangen og forhandler fram en egen nøkkel med hvert relé. Deretter pakker den hver bit data inn i ett krypteringslag per ledd. Hvert relé skreller av sitt eget lag og får bare vite hvem som kom før, og hvem som kommer etter. Lagene inni hverandre er løken i tittelen på designartikkelen, The Second-Generation Onion Router. I dagligtale er reléene samlet Tor-nettverket, tor med liten forbokstav er programmet, og Tor Browser er nettleseren som er pakket for å bruke det.

Data flyttes i celler med fast størrelse, så et relé ser en strøm av like store blokker i stedet for formen på de opprinnelige nettverkspakkene. Flere TCP-strømmer kan dele en krets når isolasjonskravene deres er forenlige. Klienten bygger kretser både på forhånd og ved behov og tar dem ut av bruk etter alder og bruk; en langvarig strøm kan bli på sin opprinnelige krets, og noen isolasjonsinnstillinger tillater lengre gjenbruk. Applikasjonene når klienten gjennom en lokal SOCKS-proxy, og bare TCP går denne veien; designartikkelen sier rett ut at den ikke tunnelerer protokoller som UDP. Frivillige driver reléene, og hvert av dem publiserer en beskrivelse med nøkler, adresse, båndbredde og utgangsregler. Et lite sett katalogautoriteter stemmer over hvilke reléer som skal listes, og signerer resultatet som et konsensusdokument som hver klient henter før den velger rute.

Bruk begrepet i en beslutning

Bruk «Hva er Tor?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Wi-Fi, brannmurer og fjerntilgang satt opp slik at de ansatte kan jobbe hvor som helst, og bare de rette personene slipper inn.

**Se tjenestene våre** Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Administrerte 30 Ubuntu Linux-VPS-instanser — 30 Linux-servere administrert, med gjenoppretting planlagt på forhånd

**Administrerte 30 Ubuntu Linux-VPS-instanser** Ta kontakt →

# Hvorfor det er viktig

For den som driver et offentlig nettsted, er Tor først og fremst et faktum om publikum. En leser som kommer over Tor, ser ut til å komme fra adressen til et utgangsrelé og ikke fra sin egen, og listen over reléer er offentlig med vilje, fordi hver klient trenger den for å velge rute. Et nettsted kan altså kjenne igjen Tor-trafikk og behandle den annerledes. Tor Project står selv på at operatører har rett til å blokkere den, selv om prosjektet håper de lar være. Det du da må avgjøre, er hva du gjør med disse leserne: betjener dem som alle andre, bremser dem, setter en utfordring foran dem eller gir dem en vei inn som aldri forlater Tor-nettverket, og det siste er det en onion-adresse gjør.

Det lønner seg å være presis om hva Tor lover, for designartikkelen lister selv opp det som er holdt utenfor. Den hevder ikke å slå en observatør som kan se begge ender av en forbindelse og sammenligne tidspunkter og volum, og prosjektets dokumentasjon legger til at forskerne ikke kjenner noe praktisk design med lav latens som klarer det pålitelig. Grunnprotokollen skjuler ikke selve bruken av Tor; broer med pluggable transports gir flere måter å kamuflere den trafikken på. Og Tor krypterer ikke den siste strekningen fra utgangsreléet til målet: applikasjonen trenger egen kryptering for det leddet.

Sammenlignet med et VPN er tilliten fordelt annerledes. En VPN-leverandør ser både brukeren og målet og blir bedt om å se bort. På Tor ser ingen av reléene begge deler. På de fleste ruter holder utvalgsreglene reléer fra samme erklærte familie fra hverandre og unngår som standard et felles IPv4-/16- eller IPv6-/32-subnett; det finnes dokumenterte unntak og konfigurerbare subnettgrenser. Prisen er latens, siden en vanlig utgangskrets krysser tre maskiner drevet av frivillige, så sider som er lette og trenger få rundturer, fungerer best for lesere over Tor. Å tilby en onion-adresse krever en tor-prosess og en privat nøkkel å passe på, men ingen åpen innkommende nettverksport, og derfor passer det på liten infrastruktur.

## Slik ser det ut i praksis

Selskapets eget nettsted kan leses over Tor. Kommandolinjesiden publiserer onion-adressen ved siden av en `curl` -kommando som henter Markdown-teksten gjennom en lokal tor-klient. Adressen er vanlig `http://` med vilje: en onion-adresse av versjon 3 koder tjenestens offentlige identitetsnøkkel sammen med en kontrollsum og en versjonsbyte. Tor autentiserer tjenesten og krypterer kretsen før HTTP utveksles. Et offentlig websertifikat trengs ikke for denne autentiseringen av onion-adressen.

En onion-tjeneste trengte plass til en tor-prosess på selskapets vert for offentlig tilgjengelige tjenester fra september 2026: én kjerne og omtrent 464 MB brukbart minne. Den terminerte allerede TLS for sju domener og kjørte en git-forge. Plassen til denne prosessen ble målt i stedet for diskutert, gjennom et dokumentert døgn med stikkprøvemålinger på denne verten med 512 MB: den minste registrerte mengden tilgjengelig minne var omtrent 310 MB av de 464, swap-bruken var 2,6 %, og prosessoren var ledig 99,7 % av tiden. Tor ble derfor lagt til ved siden av resten.

Personvernsiden beskriver selskapets valg om å telle lesere. Gemini- og Gopher-applikasjonene fører ingen tilgangslogg, og de daglige tellerne registrerer bare samlet antall forespørsler. Gopher har også en tilkoblingslogg i operativsystemet som inneholder kildeadressen; den loggen slettes etter 30 dager. Selskapet sier at det ikke teller noe som helst over Tor og ikke tilbyr noen innstilling for å slå det på. Det er operatørens personvernvalg: Tor skjuler leserens nettverksadresse, men hindrer ikke en tjeneste i å telle forespørsler. Siden sier at selskapet heller vil la være å vite enn å kunne vite.

## Hvor det går galt

**Navnet slås opp utenfor Tor.** Et program som sender forbindelsene sine til en SOCKS-proxy, kan likevel slå opp vertsnavnet selv først, gjennom den vanlige navneoppslagstjenesten, og det oppslaget forteller det lokale nettverket og DNS-leverandøren hvilket nettsted som besøkes. Forbindelsen går deretter gjennom Tor, men målet har allerede lekket. I `curl` er forskjellen ett eneste valg: med `--socks5` slås navnet opp lokalt, med `--socks5-hostname` overlates det til proxyen, og utgangsreléet slår det opp.

```
# Slår opp navnet lokalt, kobler så til gjennom proxyen
curl -s --socks5 127.0.0.1:9050 https://example.org/
# Gir navnet til tor; utgangsreléet slår det opp
curl -s --socks5-hostname 127.0.0.1:9050 https://example.org/
```

**Klartekst etter utgangen.** Tor krypterer leddene mellom reléene, men det siste leddet går som det er. Utgangsreléet åpner en vanlig TCP-forbindelse til målet og sender det applikasjonen ga det, så et innloggingsskjema som sendes over vanlig HTTP, kan leses av den som driver utgangsreléet, og av alle på nettet bortenfor. Lever HTTPS for alt som forlater Tor-nettverket. En onion-tjeneste slipper unna spørsmålet, fordi trafikken dens aldri forlater nettverket.

**Én adresse står for mange lesere.** Grenser for antall forespørsler og utestenginger knyttet til klientadressen behandler et utgangsrelé som én besøkende, selv om mange bruker det samtidig. En grense tilpasset én person stenger ute alle hvis krets tilfeldigvis ender der, og en utestenging som én misbruker har gjort seg fortjent til, rammer dem alle. Bestem med vilje hvordan utgangsadresser skal behandles, i stedet for å la en generell regel per adresse bestemme for deg.

**Identiteten reiser inne i kretsen.** Tor skjuler nettverksadressen og gjør ingenting med det applikasjonen sier. En innlogging på en personlig konto, et særpreget nettleseroppsett eller et dokument som henter noe når det åpnes, knytter økten til en person, uansett hvor mange reléer som ligger imellom. Prosjektets egen dokumentasjon påpeker at applikasjonstrafikken fortsatt må renses på brukerens side. Den jobben tar Tor Browser på seg, og den blir liggende når en vanlig nettleser bare pekes mot proxyen.

**Et utgangsrelé på en hjemmelinje.** Vakt- og mellomreléer trekker sjelden til seg klager på misbruk. Utgangen er der målene ser trafikken komme fra, så klager og en og annen juridisk henvendelse går til den som driver den, og relé-veiledningen fra Tor Project sier at du ikke bør drive en utgang hjemmefra. En frivillig med én hjemmeadresse kan heller drive en bro (*bridge*), et relé som står utenfor den offentlige listen, og som den samme veiledningen anbefaler i den situasjonen.

**Forventningen om at UDP skal virke.** Taleanrop, mange spill og verktøy som sender egne DNS-spørringer, er avhengige av UDP, og Tor frakter bare TCP-strømmer. Slike programmer går ikke gjennom Tor, og et program som ikke

tvinges gjennom proxyen, kan rett og slett koble direkte til, utenfor Tor, uten å si fra.

## Les mer

- Tor Specifications — protokollen, rutevalget, katalogen og spesifikasjonene for onion-tjenester, samlet på ett sted.
- Tor: The Second-Generation Onion Router — designartikkelen: kretser, celler, utgangsregler og en uttrykkelig liste over det designet ikke prøver på.
- Tor Path Specification — hvordan en klient velger vakt, mellomrelé og utgang, og hvilke reléer som ikke får dele rute.
- Types of relays on the Tor network — vakt-, mellom-, utgangs- og broreléer, med den juridiske risikoen hvert av dem gir den som driver det.
- HTTPS encryption and Tor — en interaktiv visning av hva en avlytter ser med Tor, med HTTPS, med begge og med ingen av dem.

Ordliste

Velg ut fra deres faktiske rammer

Ta med utrullingsløpet, driftsrisikoene og dem som har vakt. Vi kan avgrense en endring med kontroller, en vei tilbake og tydelig ansvar.

**Se hva det koster** Ta kontakt →

## Gjør begrepene praktiske

Spør hvordan «Hva er Tor?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

**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/glossary/tor/**