Engineer

Tor

🧅 et netværk der sender trafik gennem frivillige relæer

Hvad det er

Hvem der taler med hvem, er netop det, Tor skjuler for den, der holder øje med forbindelsen. Hver forbindelse sendes gennem en kæde af maskiner, og ingen af dem får at vide både, hvor trafikken begyndte, og hvor den ender. Kæden vælges af klienten, programmet tor, der kører på læserens egen computer, og den er som regel tre relæer lang: en guard, som klienten bliver ved med at bruge som indgang, et mellemrelæ og et exit-relæ, der åbner den sidste forbindelse til målet. Klienten bygger kredsløbet ét hop ad gangen, aftaler en særskilt nøgle med hvert relæ og pakker hver bid data ind i ét krypteringslag pr. hop. Hvert relæ fjerner sit eget lag og kender kun naboen før og naboen efter. Lagene er løget i titlen på designartiklen, The Second-Generation Onion Router. I daglig tale hedder relæerne samlet Tor-netværket, tor med lille t er programmet, og Tor Browser er den browser, der er pakket til at bruge det.

Data rejser i celler af fast størrelse, så et relæ ser en strøm af ensartede blokke i stedet for de oprindelige pakkers form. Flere TCP-strømme kan dele et kredsløb, når deres isolationskrav er forenelige. Klienten bygger kredsløb både på forhånd og efter behov og udfaser dem efter alder og brug; en langvarig strøm kan blive på sit oprindelige kredsløb, og nogle isolationsindstillinger tillader længere genbrug. Programmer når klienten gennem en lokal SOCKS-proxy, og kun TCP kan komme den vej; designartiklen siger ligeud, at protokoller som UDP ikke tunneleres. Relæerne drives af frivillige. Hvert relæ offentliggør en beskrivelse med sine nøgler, sin adresse, sin båndbredde og sin exit-politik. En lille gruppe directory authorities stemmer om, hvilke relæer der kommer på listen, og underskriver resultatet som et konsensusdokument, som hver klient henter, før den vælger en sti.

Brug begrebet i en beslutning

Brug »Hvad er Tor?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Wi-Fi, firewalls og fjernadgang sat op, så medarbejdere kan arbejde hvor som helst, og kun de rette kommer ind.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har administreret 30 Ubuntu Linux VPS-instanser — 30 Linux-servere driftet, med en plan for gendannelse

Har administreret 30 Ubuntu Linux VPS-instanser Kontakt os →

Hvorfor det betyder noget

Den, der driver et offentligt website, møder først Tor som en kendsgerning om publikummet. En læser, der kommer via Tor, ser ud til at komme fra et exit-relæs adresse og ikke fra sin egen, og listen over relæer er offentlig med vilje, fordi hver klient skal bruge den for at vælge sin sti. Et site kan altså genkende Tor-trafik og behandle den anderledes, og Tor-projektet mener selv, at operatører har ret til at blokere den, selv om det håber, at de lader være. Tilbage står beslutningen om, hvad man gør med de læsere: betjener dem som alle andre, bremser dem, stiller en udfordring foran dem eller giver dem en indgang, der aldrig forlader Tor-netværket. Det sidste er præcis, hvad en onion-adresse gør.

Det kan betale sig at være nøjagtig med, hvad Tor lover, for designartiklen opregner selv, hvad den lader ligge. Den påstår ikke at kunne stå imod en modstander, der ser begge ender af en forbindelse og sammenligner timing og mængde, og projektets dokumentation tilføjer, at forskningen ikke kender noget praktisk design med lav latens, der pålideligt kan. Grundprotokollen skjuler ikke selve brugen af Tor; broer med pluggable transports giver ekstra muligheder for at skjule den trafik. Og Tor krypterer ikke den sidste strækning fra exit-relæet til målet: applikationen skal selv bruge kryptering på det hop.

Sammenlignet med en VPN er tilliden fordelt på en anden måde. En VPN-udbyder ser både brugeren og målet og bliver bedt om at se væk. På Tor ser intet enkelt relæ begge dele. På de fleste stier holder udvælgelsesreglerne relæer fra samme erklærede familie adskilt og undgår som standard et fælles IPv4-/16- eller IPv6-/32-subnet; der findes dokumenterede undtagelser og konfigurerbare subnetgrænser. Prisen er latens, fordi et almindeligt exit-kredsløb krydser tre frivilliges maskiner, og derfor tjener lette sider med få rundture Tor-læserne bedst. En onion-adresse kræver en tor-proces og en privat nøgle at passe på, men ingen åben indgående port, og derfor passer den på en lille infrastruktur.

Sådan ser det ud i praksis

Virksomhedens eget site kan læses over Tor. Kommandolinjesiden offentliggør onion-adressen sammen med en `curl` -kommando, der henter Markdown-teksten gennem en lokal tor-klient. Adressen er med vilje almindelig `http://` : en version 3-onion-adresse koder tjenestens offentlige identitetsnøgle sammen med en kontrolsum og en versionsbyte. Tor autentificerer tjenesten og krypterer kredsløbet, før der tales HTTP. Et offentligt webcertifikat er ikke nødvendigt for den autentificering af onion-adressen.

En onion-tjeneste krævede plads til en tor-proces på værten til virksomhedens offentligt tilgængelige tjenester fra september 2026: én kerne og omkring 464 MB brugbar hukommelse. Værten terminerede allerede TLS for syv domæner og kørte en git-forge. Pladsen til den proces blev målt frem for diskuteret gennem et registreret døgns stikprøver på værten med 512 MB: den laveste målte tilgængelige hukommelse var omkring 310 MB af de 464, swap brugte 2,6 procent, og processoren var inaktiv 99,7 procent af tiden, så Tor kom til ved siden af resten.

Privatlivssiden beskriver virksomhedens valg om at tælle læsere. Gemini- og Gopher-applikationerne fører ingen adgangslog, og deres daglige tællere registrerer kun det samlede antal forespørgsler. Gopher har desuden en forbindelseslog i styresystemet, som indeholder kildeadressen; den log slettes efter 30 dage. Virksomheden siger, at den slet intet tæller over Tor og ikke tilbyder nogen indstilling til at slå det til. Det er operatørens privatlivsvalg: Tor skjuler læserens netværksadresse, men forhindrer ikke en tjeneste i at tælle forespørgsler. Ifølge siden vil virksomheden hellere lade være med at vide det end kunne.

Hvor det går galt

Navnet slås op uden for Tor. Et program, der sender sine forbindelser til en SOCKS-proxy, kan stadig selv slå værtsnavnet op først gennem den almindelige resolver, og det opslag fortæller det lokale netværk og DNS-udbyderen, hvilket site der besøges. Forbindelsen går bagefter gennem Tor, men målet er allerede sluppet ud. I `curl` er forskellen en enkelt option: med `--socks5` slås navnet op lokalt, med `--socks5-hostname` gives det videre til proxyen, og exit-relæet slår det op.

```
# Slår navnet op lokalt og forbinder derefter gennem proxyen
curl -s --socks5 127.0.0.1:9050 https://example.org/
# Giver navnet videre til tor; exit-relæet slår det op
curl -s --socks5-hostname 127.0.0.1:9050 https://example.org/
```

Klartekst efter exit. Tor krypterer hoppene mellem relæerne, ikke det sidste. Exit-relæet åbner en almindelig TCP-forbindelse til målet og sender det, programmet gav det, så en loginformular, der sendes over almindelig HTTP, kan læses af den, der driver exit-relæet, og af alle på nettet bag det. Brug HTTPS til alt, der forlader Tor-netværket. En onion-tjeneste slipper for spørgsmålet, fordi dens trafik aldrig forlader det.

Én adresse dækker mange læsere. Hastighedsgrænser og blokeringer, der hænger på klientens adresse, behandler et exit-relæ som én besøgende, selv om mange mennesker bruger det samtidig. En grænse, der er afstemt efter én person, lukker alle ude, hvis kredsløb tilfældigvis ender der, og en blokering, som én misbruger har gjort sig fortjent til, rammer dem alle. Beslut med vilje, hvordan exit-adresser behandles, i stedet for at lade en generel regel pr. adresse gøre det.

Identiteten rejser med inde i kredsløbet. Tor skjuler netværksadressen og gør intet ved det, programmet siger. At logge ind på en personlig konto, en særpræget browseropsætning eller et dokument, der henter noget, når det åbnes, knytter sessionen til en person, uanset hvor mange relæer der står imellem. Projektets egen dokumentation påpeger, at programtrafikken stadig skal renses på brugerens side. Det er den opgave, Tor Browser påtager sig, og som en almindelig browser, der peges mod proxyen, lader ligge.

Et exit-relæ på en hjemmeforbindelse. Guard- og mellemrelæer får sjældent klager over misbrug. Exit-relæet er det sted, målene ser trafikken komme fra, så klager og en sjælden henvendelse fra myndighederne går til den, der driver det, og Tor-projektets relæguide siger, at man ikke skal køre et hjemmefra. En frivillig med én hjemmeadresse kan i stedet køre en bridge, et relæ uden for den offentlige liste, som samme guide anbefaler til netop den situation.

At forvente, at UDP virker. Taleopkald, mange spil og værktøjer, der sender deres egne DNS-forespørgsler, bygger på UDP, og Tor bærer kun TCP-strømme. De programmer virker ikke gennem Tor, og et program, der ikke er tvunget

gennem proxyen, kan uden videre forbinde direkte uden om Tor uden at sige det.

Læs mere

- Tor Specifications — specifikationerne for protokollen, stivalget, directory-systemet og onion-tjenester samlet ét sted.
- Tor: The Second-Generation Onion Router — designartiklen: kredsløb, celler, exit-politikker og en udtrykkelig liste over det, designet ikke forsøger.
- Tor Path Specification — hvordan en klient vælger guard, mellemrelæ og exit, og hvilke relæer der ikke må dele en sti.
- Types of relays on the Tor network — guard-, mellem-, exit- og bridge-relæer og det juridiske ansvar, hvert af dem lægger på den, der driver det.
- HTTPS encryption and Tor — en interaktiv visning af, hvad en aflytter ser med Tor, med HTTPS, med begge dele og med ingen af dem.

Ordliste

Vælg ud fra jeres faktiske vilkår

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er Tor?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

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

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/tor/