Engineer

Proxy

en server som videresender en forespørsel på dine vegne

Hva det er

En proxy er et program som står mellom en klient og serveren klienten vil nå. Den tar imot forespørselen fra klienten, sender en egen forespørsel til neste maskin på veien og leverer svaret tilbake. Klienten snakker med proxyen og proxyen med serveren, så de to endene deler aldri en tilkobling. Det finnes flere slags, og det nyttige spørsmålet å stille om hver av dem er hvem som satte den der.

HTTP-spesifikasjonen selv, RFC 9110, navngir tre vanlige mellomledd. En proxy i snever forstand er en klienten har valgt, som regel gjennom lokal konfigurasjon, f.eks. en innstilling i nettleseren eller en miljøvariabel. En gateway, som RFC-en også kaller en omvendt proxy (reverse proxy), står på serverens side: overfor den som kobler seg til, oppfører den seg som opprinnelsesserveren, mens den sender hver forespørsel innover til én eller flere servere bak seg. En tunnel videreformidler byte i blinde mellom to tilkoblinger og slutter å være part i samtalen så snart den er åpen. En klient ber om en tunnel med metoden CONNECT, oppgir en vert og en port, og kjører så TLS gjennom den fra ende til ende. RFC-en legger til en fjerde type som ingen har valgt: den avskjærende proxyen, ofte kalt transparent proxy, som omdirigerer utgående trafikk på port 80 i kanten av et nettverk, slik offentlige tilgangspunkter og brannmurer i bedrifter gjør.

Hver etappe er en egen tilkobling. Diagrammet i RFC-en plasserer tre mellomledd mellom en nettleser og en opprinnelsesserver, noe som gir fire tilkoblinger, og innstillinger som keep-alive eller en tidsgrense gjelder én av de tilkoblingene om gangen. HTTP ber også en proxy om å presentere seg: den må legge til et Via-headerfelt i hver melding den videresender, og det er slik en sløyfe eller et uventet mellomledd viser seg i en sporing.

HTTP er bare ett lag. En SOCKS-proxy, der versjon 5 er spesifisert i RFC 1928 fra mars 1996, videreformidler TCP-tilkoblinger og UDP-datagrammer uten å lese hva som er inni dem, så enhver protokoll kan bruke den. Tor er et kjent eksempel: som standard lytter den på port 9050 etter SOCKS-tilkoblinger fra lokale programmer.

Bruk begrepet i en beslutning

Bruk «Hva er en proxy?» 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

Den som driver proxyen, kan lese trafikken. For ren HTTP betyr det hver header og hver meldingskropp. Gjennom en CONNECT-tunnel ser den vertsnavnet og porten, og resten er TLS fra klienten til serveren. En omvendt proxy som terminerer TLS, ser alt i klartekst, og det er jobben dens, fordi den må lese en forespørsel for å rute den. Å sette et annet selskaps omvendte proxy foran et nettsted, som er det et CDN gjør, gir derfor det selskapet hver besøkendes adresse og hver URL før nettstedet selv ser dem. Det kan være riktig byttehandel for en nettbutikk under angrep. Det er likevel en beslutning om personvern, og personvernsiden bør si det.

Den andre følgen er hva serveren får vite. RFC 7239 beskriver tapet rett ut: bak en proxy ser forespørslene ut til å komme fra proxyens adresse. En logg, en hastighetsbegrensning, en sperreliste eller en tillatelsesliste som bygger på klientadressen, ser da én adresse for alle. Den vanlige reparasjonen er en header som bærer den opprinnelige adressen videre: det standardiserte Forwarded-feltet fra RFC 7239, eller den eldre X-Forwarded-For. Hvilken kopi av den headeren man skal tro på, er der det meste av trøbbelet begynner.

For det tredje endrer en proxy hva en feilmelding betyr. En 502 Bad Gateway eller en 504 Gateway Timeout skrives av et mellomledd om serveren bak det: proxyen fikk et ugyldig svar, eller ikke noe svar i tide. En 407 betyr at proxyen selv vil ha innloggingsopplysninger før den videresender noe. Å vite hvilken maskin som skrev feilen, er halve diagnosen.

For det fjerde mellomlagrer mange proxyer. RFC 9111 kaller et mellomlager som lagrer svar for mer enn én bruker, en delt cache, og den er som regel en del av et mellomledd. Den gjør et nettsted raskere og billigere å servere, og den er også det ene stedet der et svar ment for én person kan nå en annen.

Slik ser det ut i praksis

Tre deler av selskapets arbeid dreier seg om en proxy: én som ble fjernet, én som drives av fremmede, og én som ble slått av.

I august 2026 flyttet selskapet DNS-sonen sin, med omtrent tjue poster, inn i repoet og avstemte den mot API-et til Cloudflare. Å sende webtrafikken gjennom leverandørens CDN som omvendt proxy var en del av det arbeidet, og deretter ble en fungerende CDN-proxy foran nettstedet bygd ned av hensyn til personvernet. Innvendingen gjaldt plasseringen: hver lesers adresse og hver sti som ble etterspurt, ville nådd en tredjepart først, og for et selskap som forteller leserne sine at ingen ser på, veide det tyngre enn angrepsvernet det kjøpte.

Da selskapet måneden etter begynte å telle lesere fra tilgangsloggen som webserveren allerede fører, måtte det forholde seg til proxyer det ikke selv driver. Tilgangsloggen forkaster sju headere som videreformidler en klients adresse før den skriver en linje. Enkoderen for loggen kopierer headerne uendret, så uten det steget ville en leser bak en bedriftsproxy fått sin umaskerte adresse lagret under et headernavn som maskeringsregelen ikke dekket.

Da nettstedet fikk et onion-speil på Tor samme måned, ble Tor-daemonens egen proxyport for utgående trafikk slått av i konfigurasjonen. Onion-tjenesten nås gjennom kretser som daemonen oppretter utover, så publiseringen av dette speilet krever ingen egen inngående internettport. Med SOCKS-lytteporten deaktivert tilbyr denne Tor-daemonen ikke lokale programmer en utgående proxy.

Hvor det går galt

Hver forespørsel i loggen kommer fra samme adresse. Applikasjonen står bak en omvendt proxy og registrerer adressen til tilkoblingen den fikk, og det er proxyens. En hastighetsbegrensning struper da alle på én gang, og en sperring skrevet av et verktøy mot inntrengere sperrer proxyen, altså hele publikummet. La proxyen sende klientadressen i en header, og konfigurer applikasjonen til å

lese den headeren bare på tilkoblinger fra proxyens egen adresse. I nginx er det direktivet set_real_ip_from i realip-modulen; de fleste webrammeverk har en tilsvarende innstilling for betrodde proxyer.

En klient velger sin egen adresse ved å sende en header. Applikasjonen leser X-Forwarded-For på hver forespørsel, også forespørsler som nådde den direkte, og forespørsler som kom med en forfalsket verdi allerede i headeren. RFC 7239 er tydelig på at feltet kan endres av hver node på veien, klienten inkludert. Stol bare på oppføringene din egen proxy la til, regnet fra høyre, og bind applikasjonen til loopback eller et privat nettverk, slik at proxyen er eneste vei inn.

Loggen som skulle anonymiseres, beholder fulle adresser. Klientadressen er maskert i feltet noen sjekket, men den samme adressen reiser i Forwarded, X-Forwarded-For, X-Real-IP og lignende headere, og en logger som skriver alle headerne slik de kom inn, lagrer dem i sin helhet. List opp hver header som kan bære en adresse, fjern eller masker hver av dem i det øyeblikket linjen skrives, og send så en testforespørsel og les linjen på disken.

Sider svarer 502 eller 504, og backenden logger ingenting. Feilen ble skrevet av proxyen. En 502 betyr at den fikk et ugyldig svar: backenden var nede, lyttet på en annen port, eller snakket ren HTTP der proxyen ventet TLS, eller omvendt. En 504 betyr at backenden brukte lenger tid enn lesefristen til proxyen, som nginx setter til 60 sekunder som standard. Les feilloggen til proxyen først, fordi den navngir oppstrømsadressen den prøvde, og øk en tidsgrense bare for rutene dere vet er trege.

Én bruker får se en annen brukers side. En delt cache i en proxy eller et innholdsnettverk lagret et svar som var personlig, f.eks. en kontoside som kom sammen med en informasjonskapsel for økten, og serverte det til nestemann som ba om den samme URL-en. Merk hvert svar per bruker med Cache-Control: private, som RFC 9111 sier at en delt cache ikke får lagre, eller med no-store der ingen kopi skal finnes noe sted, og sjekk headerne proxyen faktisk returnerer.

Les mer

- RFC 9110, section 3.7: Intermediaries — definisjonene av proxy, gateway, tunnel og avskjærende proxy.
- RFC 9110, section 9.3.6: CONNECT — hvordan en klient ber en proxy om å åpne en tunnel til en vert og en port.

- RFC 7239: Forwarded HTTP Extension — den standardiserte headeren for klientens opprinnelige adresse, med forbeholdene.
- RFC 9111: HTTP Caching — delte og private mellomlagre, og direktivene som holder dem fra hverandre.
- RFC 1928: SOCKS Protocol Version 5 — den protokollnøytrale proxyen som videreformidler TCP-tilkoblinger og UDP.

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 en proxy?» 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/proxy/