Engineer

Omvendt proxy

← en server som tar imot trafikk på vegne av en tjeneste

Hva det er

En omvendt proxy tar imot forespørsler på vegne av en tjeneste. Når nettleseren åpner nettstedets offentlige adresse, kobler den seg til proxyen. Proxyen sender forespørselen videre til den aktuelle tjenesten og gir nettleseren svaret. Tjenesten i den andre enden kalles gjerne opphavsserveren. Den kan ligge i et privat nett eller flytte til en annen maskin mens publikum fortsatt bruker den samme adressen. I HTTP-standarden heter rollen «gateway». En vanlig, fremoverrettet proxy er på klientens side og kontakter andre tjenester for den.

Tenk deg et nettsted med en forside og et endepunkt for rapporter. Nettleseren bruker samme domene til begge, men `/api/rapporter/42` sendes til rapporttjenesten mens forsiden kommer fra et annet sted. Valget kan styres av stien i forespørselen. En proxy kan også velge blant flere likeverdige tjenere for den samme stien. Det valget er lastbalansering. Det er fullt mulig å ha en omvendt proxy foran én tjener, for eksempel for å skille en offentlig adresse fra den interne adressen som applikasjonen faktisk bruker.

Noen produkter kan videresende en kryptert TCP-forbindelse direkte til en tjener. I den modusen kan de bruke informasjon om forbindelsen og eventuelle synlige felt i TLS-forhandlingen. Ruting som avhenger av de krypterte HTTP-stiene krever at komponenten kan lese HTTP-meldingen. I et driftsdiagram bør det derfor stå hvor TLS termineres, og om proxyen åpner en ny kryptert forbindelse videre. Da fremgår det hvilke beslutninger hvert ledd kan ta.

Bruk begrepet i en beslutning

Bruk «Hva er en omvendt 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

En felles inngang gir den som drifter løsningen, et sted å bestemme hvilke tjenester som er tilgjengelige utenfra. Proxyen kan håndtere sertifikatet for det offentlige domenet, mens applikasjonstjenerne bare er tilgjengelige på et privat nett. Den kan også håndheve tidsgrenser, begrense hvor store forespørsler den godtar og skrive en tilgangslogg. Reglene bør gjenspeile hvordan hver rute brukes. Opplastinger og strømming har andre behov enn et enkelt svar fra en statusside.

Når en proxy terminerer TLS, blir innholdet i HTTP-meldingen tilgjengelig for den. Det kan omfatte informasjonskapsler, søkeparametere og det brukeren sender i en forespørsel. Kryptering videre til opphavsserveren beskytter neste forbindelse under transport. Den som driver proxyen, kan fortsatt behandle innholdet. Avklar hvem som har tilgang til dataene, hvor lenge logger blir lagret og hvilke opplysninger som faktisk trengs i dem.

Det blir dessuten flere steder å lete når en feil oppstår. En proxy kan svare med `502` når den mottar et ugyldig svar, eller `504` når den ikke får svar i tide fra tjeneren bak seg. Applikasjonen kan også lage en feilrespons selv. Finn logghendelsene for samme forespørsel på begge sider, og sammenlign status og tidspunkt. Hvis bare proxyen har registrert hendelsen, må undersøkelsen omfatte forbindelsen mellom den og opphavsserveren.

Slik ser det ut i praksis

I arbeidet med å avstemme DNS-sonen som kode beskriver Engineer ApS en CDN-proxy som virket og senere ble skrudd av. Leverandørens nett håndterte de besøkendes adresser og URL-ene de ba om, før forespørslene nådde selskapet. I den saken veide personvernhensynet tyngre enn beskyttelsen mot angrep som leverandøren tilbød. Arbeidet med å versjonere og kontrollere DNS-sonen fortsatte. Å bruke en DNS-tjeneste og å føre trafikken gjennom leverandørens proxy er to valg man kan vurdere hver for seg.

I et testmiljø kan du starte med `/health` og `/api/rapporter/42?visning=kort`. Sjekk hvilken metode og sti applikasjonen mottar, og hvilken status og eventuelt videresendingsmål nettleseren ser. Flytt deretter opphavstjenesten til en ny intern adresse og gjenta testen med den samme offentlige adressen. Prøv også den dype stien med flere parametere. Da kan du oppdage om en ruteregel har slukt et stisegment eller endret forespørselen.

Før du innfører mellomlagring, åpner du to økter med forskjellige testbrukere. La hver bruker se et ufarlig ord som er unikt for økten, og kontroller at responsen aldri byttes mellom dem. Stans så opphavstjeneren med vilje, gjør én kontrollert forespørsel og noter når feilen dukker opp. Se hvor hendelsen logges, og om den kan skilles fra en tilsvarende feil applikasjonen rapporterer selv.

Hvor det går galt

En forespørsel utenfra kan allerede inneholde `Forwarded` eller `X-Forwarded-For`. Applikasjonen må vite hvilke mellomledd den kan stole på før den bruker disse feltene. Fjern eller erstatt verdier fra kilder som ikke er godkjent, og begrens hvem som kan koble direkte til opphavstjeneren. Bruk en bevisst feil klientadresse i en test. Sammenlign både adressen som havner i loggen og den applikasjonen bruker i en tilgangsbeslutning.

Omdirigeringer viser ofte at en applikasjon har gjettet feil om den offentlige adressen. Nettleseren kom inn over HTTPS, men applikasjonen bygde en URL med HTTP, eller valgte den interne porten. Gi applikasjonen det verifiserte vertsnavnet og skjemaet. Følg hele påloggingsløpet og sjekk omdirigeringer og informasjonskapsler. En vellykket forespørsel på forsiden gir lite informasjon om hva som skjer ved et mer sammensatt endepunkt.

Tidsgrenser og automatiske nye forsøk må prøves sammen. En betaling kan være registrert selv om proxyen avbryter ventingen før den får svar. Bestem hvordan gjentatte forespørsler skal behandles før proxyen får fullmakt til å sende den typen handling på nytt. Forsink et svar etter at en kontrollert handling er gjennomført i testmiljøet. Noter hvem som avbryter ventingen og hvilke spor som gjør det mulig å skille en gjentakelse fra den første forespørselen.

Les mer

- RFC 9110 om mellomledd i HTTP beskriver proxyer, gatewayer og tunneler.

- RFC 7239 om Forwarded forklarer feltet og hensynene til sikkerhet og personvern.
- NGINX' HTTP-proxymodul dekker ruting, hoder, buffering og tidsgrenser.
- Caddys reverse_proxy-direktiv beskriver valg av opphavstjener og videresendte hoder.
- Veiledningen til Apache HTTP Server viser et oppsett der en offentlig tjener ruter til flere tjenester.

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 omvendt 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/reverse-proxy/