Engineer

Omvendt proxy

← en server, der modtager trafik på webtjenestens vegne

Hvad det er

En omvendt proxy er den server, browseren først kontakter på et websteds offentlige adresse. Den tager imod browserens anmodning, sender den til den tjeneste, der kan svare, og giver browseren svaret tilbage. Tjenesten bagved kaldes her oprindelsesserveren. Den kan flytte til en anden maskine eller få en ny intern adresse, mens brugerne bliver ved med at åbne det samme websted. HTTP-standarden kalder denne rolle en gateway. En almindelig, fremadrettet proxy repræsenterer klienten, der vil kontakte en anden tjeneste. Den omvendte proxy er sat op af den, der driver webstedet.

Tænk på et websted, hvor forsiden hentes fra én tjeneste, mens `/api/rapporter/42` sendes til en anden. Browseren kender stadig ét domæne. Den omvendte proxy læser anmodningen og vælger en vej ud fra stien. Den kan samtidig vælge en server blandt flere, der kan svare på samme anmodning. Den sidste opgave er belastningsfordeling. En proxy foran en enkelt tjeneste er stadig en omvendt proxy, og det er ofte den enkleste måde at se, hvilke opgaver komponenten faktisk løser.

For at sortere anmodninger efter HTTP-stien skal komponenten have adgang til HTTP-beskeden. Nogle produkter kan videresende en krypteret TCP-forbindelse direkte til en bagvedliggende tjeneste. I den tilstand kan de bruge oplysninger om forbindelsen og eventuelt synlige felter i det første trin af TLS, hvis de er tilgængelige. Stien i en krypteret anmodning bliver først læsbar, hvor TLS-forbindelsen ender.

Brug begrebet i en beslutning

Brug »Hvad er en omvendt proxy?« 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 offentlige indgang giver et hold mulighed for at føre hver anmodning hen, hvor den hører til, og for at definere, hvilke tjenester der kan nås udefra. Proxyen kan håndtere webstedets certifikat, mens applikationerne ligger på et privat net. Det giver også ét sted at fastsætte logning, grænser for anmodningers størrelse og regler for omdirigering. Fælles regler fungerer bedst, når undtagelser har en begrundelse. En sti til filoverførsel kan for eksempel have brug for en større grænse end en sti til at kontrollere tjenestens status.

Når proxyen afslutter TLS, kan den læse de oplysninger, der bliver sendt videre. Det gælder også cookies og indholdet af en anmodning. En ny krypteret forbindelse beskytter trafikken på vej til oprindelsesserveren, mens operatøren af proxyen stadig kan se de oplysninger, serveren behandler. Beslut derfor, hvem der driver indgangen, har adgang til loggene og sletter dem, inden I skriver noget om, hvordan personoplysninger bliver håndteret.

Den ekstra forbindelse har også betydning, når noget fejler. En gateway kan sende `502`, hvis den får et ugyldigt svar bagfra, eller `504`, hvis svaret ikke kommer inden for dens tidsfrist. Applikationen kan også selv sende en fejlkode. Find den samme testanmodning ved begge led, og læs svartid og loghændelser sammen. Så bliver det lettere at se, om anmodningen overhovedet nåede applikationen.

Sådan ser det ud i praksis

I arbejdet med at holde DNS-zonen i overensstemmelse med koden beskriver Engineer ApS en CDN-proxy, der virkede og senere blev taget ud af drift. Da den var aktiv, håndterede udbyderens netværk besøgendes adresser og de URL'er, de bad om, før selskabet modtog anmodningerne. Virksomheden vurderede, at hensynet til de besøgendes data vejede tungere end den beskyttelse mod angreb, proxyen gav. DNS-zonen kunne stadig versionsstyres og kontrolleres. Sagen viser, at DNS-drift og valg af en proxy er to beslutninger, selv når udbyderen kan stå for begge.

På et testmiljø kan man tage udgangspunkt i `/health` og `/api/rapporter/42?visning=kort`. Kontrollér, hvilken sti og metode der når applikationen, hvor lang tid svaret tager, og hvilket offentligt navn browseren ser i en eventuel omdirigering. Skift den interne adresse og gentag de samme anmodninger på det offentlige navn. På den måde dækker testen også den almindelige driftssituation, hvor holdet flytter en tjeneste uden at ændre de links, brugerne allerede kender.

Test derefter en overførsel tæt på den aftalte størrelsesgrænse. Åbn to adskilte sessioner med hver sin testbruger, inden I bruger en delt cache, og kontrollér, at hver bruger kun får sit eget indhold. Sluk til sidst den bagvedliggende tjeneste i testmiljøet. En måling af svar, ventetid og loglinjer fortæller, hvad holdet kan regne med at finde, hvis samme svigt opstår senere.

Hvor det går galt

En anmodning kan allerede indeholde en `Forwarded`-header, når den kommer fra internettet. Hvis applikationen bruger oplysningerne ukritisk, kan den knytte anmodningen til en forkert klientadresse. Gør det eksplicit, hvilke proxyer der må videregive disse oplysninger, og hvilke værdier grænsen erstatter eller fjerner. Prøv en anmodning med en opdigtet adresse og sammenhold både det, applikationen registrerer, og det, den bruger i sine egne regler.

En anden fejl ses ofte først ved login. Browseren brugte HTTPS, mens applikationen byggede sit returlink ud fra den interne forbindelse og valgte HTTP eller et internt portnummer. Giv applikationen det verificerede offentlige navn og skema. Kontrollér omdirigeringer, returlinks og cookies på flere stier. Hvis proxyen skal fjerne en del af stien, så brug en anmodning med flere led og parametre til at kontrollere, at kun det aftalte er ændret.

Vurdér til sidst, hvilke anmodninger proxyen må prøve igen. Hvis et svar på en betaling udebliver, kan applikationen stadig have registreret betalingen. Aftal, hvordan gentagelser håndteres, før der indføres automatiske nye forsøg. Test en kontrolleret anmodning, hvor den bagvedliggende tjeneste er færdig med arbejdet og holder svaret tilbage. Så kan holdet se, hvilken komponent der afbryder forbindelsen, og om klienten får oplysninger nok til at gå videre.

Læs mere

- RFC 9110 om HTTP-mellemled forklarer rollerne som proxy, gateway og tunnel.
- RFC 7239 om Forwarded beskriver feltet og hensyn til sikkerhed og privatliv.
- NGINX' HTTP-proxymodul dokumenterer ruter, headere, buffering og tidsgrænser.
- Caddys reverse_proxy-direktiv gennemgår servervalg, videresendte headere og forbindelser.
- Apache HTTP Servers vejledning viser opsætninger med en fælles indgang til flere tjenester.

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 en omvendt proxy?« 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/reverse-proxy/