Engineer

Proxy

en server som vidarebefordrar en begäran åt dig

Vad det är

En proxy är ett program som är placerat mellan en klient och den server klienten vill nå. Den tar emot klientens begäran, gör en egen begäran till nästa maskin i kedjan och lämnar tillbaka svaret. Klienten talar med proxyn och proxyn talar med servern, så de två ändarna delar aldrig en anslutning. Proxyer finns i flera sorter, och den användbara frågan att ställa om var och en är vem som har placerat den där.

HTTP:s egen specifikation, RFC 9110, nämner tre vanliga mellanhänder. En proxy i snäv mening är en som klienten har valt, oftast genom lokal konfiguration som en inställning i webbläsaren eller en miljövariabel. En gateway, som RFC:n också kallar omvänd proxy (reverse proxy), sitter på serverns sida: mot den som ansluter uppträder den som ursprungsservern, medan den skickar varje begäran inåt till en eller flera servrar bakom sig. En tunnel förmedlar byte blint mellan två anslutningar och slutar vara en part i samtalet när den väl är öppen. En klient begär en tunnel med metoden CONNECT, anger en värd och en port och kör sedan TLS genom den hela vägen. RFC:n lägger till en fjärde sort som ingen har valt: den avlyssnande proxyn (interception proxy), ofta kallad transparent proxy, som styr om utgående trafik på port 80 vid kanten av ett nätverk, så som publika åtkomstpunkter och företagsbrandväggar gör.

Varje hopp är en separat anslutning. RFC:ns diagram placerar tre mellanhänder mellan en webbläsare och en ursprungsserver, vilket blir fyra anslutningar, och inställningar som keep-alive eller en tidsgräns gäller en av de anslutningarna i taget. HTTP ber också en proxy att presentera sig: den måste lägga till ett Via-fält i varje meddelande den vidarebefordrar, och det är så en loop eller en oväntad mellanhand syns i en spårning.

HTTP är bara ett lager. En SOCKS-proxy, vars version 5 specificeras i RFC 1928 från mars 1996, förmedlar TCP-anslutningar och UDP-datagram utan att läsa vad de innehåller, så vilket protokoll som helst kan använda den. Tor är ett välkänt exempel: som standard lyssnar Tor på port 9050 efter SOCKS-anslutningar från lokala program.

Använd begreppet i ett beslut

Använd ”Vad är en proxy?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Wi-Fi, brandväggar och fjärråtkomst uppsatta så att personalen kan arbeta varifrån som helst och bara rätt personer kommer in.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Förvaltade 30 VPS-instanser med Ubuntu Linux — 30 förvaltade Linux-servrar, med en plan för återställning

Förvaltade 30 VPS-instanser med Ubuntu Linux Hör av er →

Varför det spelar roll

Den som driver proxyn kan läsa trafiken. För okrypterad HTTP betyder det varje header och varje meddelandekropp. Genom en CONNECT-tunnel ser den värdnamnet och porten, och resten är TLS som löper från klienten till servern. En omvänd proxy som terminerar TLS ser allt i klartext, och det är dess uppgift, eftersom den måste läsa en begäran för att kunna dirigera den. Att ställa ett annat företags omvända proxy framför en webbplats, vilket är vad ett CDN gör, lämnar alltså över varje besökares adress och varje URL till det företaget innan webbplatsen själv ser dem. Det kan vara rätt avvägning för en webbutik som är under attack. Det är ändå ett beslut om integritet, och integritetssidan bör säga det.

Den andra följden gäller vad servern får veta. RFC 7239 beskriver förlusten rakt: bakom en proxy ser förfrågningar ut att komma från proxyns adress. En logg, en hastighetsbegränsning, en spärrlista eller en tillåtelselista som utgår från klientens adress ser då en enda adress för alla. Den vanliga lagningen är en header som för den ursprungliga adressen vidare: standardfältet Forwarded från RFC 7239, eller det äldre X-Forwarded-For. Vilken kopia av den headern man ska tro på är där de flesta problemen börjar.

Den tredje följden är att en proxy ändrar vad ett fel betyder. En 502 Bad Gateway eller en 504 Gateway Timeout skrivs av en mellanhand och handlar om servern bakom den: proxyn fick ett ogiltigt svar, eller inget svar i tid. En 407 betyder att

proxyn själv vill ha inloggningsuppgifter innan den vidarebefordrar något. Att veta vilken maskin som skrev felet är halva diagnosen.

Den fjärde är att många proxyer cachar. RFC 9111 kallar en cache som lagrar svar åt mer än en användare för en delad cache, och en sådan är oftast en del av en mellanhand. Den gör en webbplats snabbare och billigare att servera, och den är också det enda stället där ett svar som var menat för en person kan hamna hos en annan.

Så ser det ut i praktiken

Tre delar av företagets arbete hänger på en proxy: en som togs bort, en som drivs av främlingar och en som stängdes av.

I augusti 2026 flyttade företaget sin DNS-zon, ungefär tjugo poster, in i repot och stämde av den mot Cloudflares API. Att skicka webbtrafiken genom leverantörens CDN som en omvänd proxy var en del av det arbetet, och sedan monterades en fungerande CDN-proxy framför webbplatsen ned igen av integritetsskäl. Invändningen gällde var den sitter: varje läsares adress och varje begärd sökväg skulle nå en tredje part först, och för ett företag som säger till sina läsare att ingen tittar vägde det tyngre än det skydd mot attacker som den gav.

Månaden därpå skulle läsarna räknas utifrån den åtkomstlogg som webbservern redan för, och då gällde det proxyer som företaget inte själv driver. Åtkomstloggen kastar sju headers som vidarebefordrar en klients adress innan den skriver en rad. Loggens kodare kopierar headers oförändrade, så utan det steget hade en läsare bakom en företagsproxy fått sin omaskerade adress lagrad under ett headernamn som maskeringsregeln inte täckte.

När webbplatsen samma månad fick en onion-spegel på Tor stängdes Tor-daemonens egen proxyport för utgående trafik av i konfigurationen. Onion-tjänsten nås genom kretsar som daemonen upprättar utåt, så publiceringen av den här spegeln kräver ingen separat inkommande internetport. Med SOCKS-lyssnaren avstängd erbjuder den här Tor-daemonen ingen utgående proxy för lokala program.

Var det går fel

Varje begäran i loggen kommer från samma adress. Applikationen sitter bakom en omvänd proxy och registrerar adressen på den anslutning den tog emot, alltså proxyns. En hastighetsbegränsning stryper då alla på en gång, och en spärr som ett intrångsverktyg skriver spärrar proxyn, vilket är detsamma som hela publiken. Låt proxyn skicka med klientens adress i en header och konfigurera applikationen att läsa den headern bara på anslutningar från proxyns egen adress. I nginx är det direktivet set_real_ip_from i modulen realip, och de flesta webbramverk har en motsvarande inställning för betrodda proxyer.

En klient väljer sin egen adress genom att skicka en header. Applikationen läser X-Forwarded-For på varje begäran, även på förfrågningar som nådde den direkt och på sådana som kom med ett förfalskat värde redan i headern. RFC 7239 säger uttryckligen att fältet kan ändras av vilken nod som helst på vägen, klienten inräknad. Lita bara på de poster som din egen proxy lade till, räknat från höger, och bind applikationen till loopback eller ett privat nät så att proxyn är den enda vägen in.

Loggen som skulle anonymiseras behåller fullständiga adresser. Klientadressen maskeras i det fält som någon kontrollerade, men samma adress färdas i Forwarded, X-Forwarded-For, X-Real-IP och liknande headers, och en loggare som skriver ut headertabellen som den togs emot lagrar dem i sin helhet. Lista varje header som kan bära en adress, ta bort eller maskera var och en i samma ögonblick som raden skrivs, och skicka sedan en testbegäran och läs raden på disk.

Sidor svarar med 502 eller 504 medan loggen i backend står tyst. Felet skrevs av proxyn. En 502 betyder att den fick ett ogiltigt svar: backend var nere, lyssnade på en annan port eller talade ren HTTP där proxyn väntade sig TLS, eller tvärtom. En 504 betyder att backend tog längre tid än proxyns tidsgräns för läsning, som nginx sätter till 60 sekunder som standard. Läs proxyns fellogg först, eftersom den namnger den uppströmsadress den försökte nå, och höj en tidsgräns bara för de vägar som man vet är långsamma.

En användare får se en annan användares sida. En delad cache i en proxy eller ett CDN lagrade ett personligt svar, t.ex. en kontosida som returnerades tillsammans med en sessionskaka, och serverade det till nästa person som bad om samma URL. Märk varje svar per användare med Cache-Control: private,

som RFC 9111 säger att en delad cache inte får lagra, eller med no-store där ingen kopia ska sparas någonstans, och kontrollera de headers som proxyn faktiskt returnerar.

Läs mer

- RFC 9110, section 3.7: Intermediaries — definitionerna av proxy, gateway, tunnel och avlyssnande proxy.
- RFC 9110, section 9.3.6: CONNECT — hur en klient ber en proxy att öppna en tunnel till en värd och en port.
- RFC 7239: Forwarded HTTP Extension — standardheadern för klientens ursprungliga adress, med de förbehåll som hör till.
- RFC 9111: HTTP Caching — delade och privata cacheminnen, och de direktiv som håller isär dem.
- RFC 1928: SOCKS Protocol Version 5 — den protokollneutrala proxyn som förmedlar TCP-anslutningar och UDP.

Ordlista

Välj utifrån era faktiska förutsättningar

Ta med driftsättningsvägen, driftsriskerna och dem som har jour. Vi kan avgränsa en förändring med kontroller, en väg tillbaka och tydligt ansvar.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är en proxy?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

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

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/proxy/