Engineer

Проксі

сервер, що передає запит від вашого імені

Що це таке

Проксі — це програма, що стоїть між клієнтом і сервером, до якого той хоче дістатися. Вона приймає запит клієнта, робить власний запит до наступної машини на шляху й повертає відповідь. Клієнт говорить із проксі, а проксі — із сервером, тож два кінці ніколи не ділять одне з'єднання. Кажуть також «проксі-сервер» або просто «посередник». Різновидів кілька, і найкорисніше запитання до кожного — хто його туди поставив.

Специфікація HTTP, RFC 9110, називає три поширені види посередників (intermediaries). Проксі у вузькому сенсі обирає сам клієнт, зазвичай через локальне налаштування: параметр браузера чи змінну середовища. Шлюз (gateway), який RFC також називає зворотним проксі (reverse proxy), стоїть на боці сервера: для того, хто підключається, він поводиться як сервер-джерело, а кожен запит передає всередину, одному чи кільком серверам за собою. Тунель сліпо пересилає байти між двома з'єднаннями й перестає бути учасником розмови, щойно його відкрито. Клієнт просить про тунель методом CONNECT, називаючи хост і порт, а потім проводить крізь нього наскрізний TLS. RFC згадує й четвертий вид, якого ніхто не обирав: проксі-перехоплювач (interception proxy), який часто називають прозорим. Він перенаправляє вихідний трафік на порт 80 на межі мережі, як це роблять публічні точки доступу та корпоративні брандмауери.

Кожен перехід — окреме з'єднання. На діаграмі в RFC між браузером і сервером-джерелом стоять три посередники, тобто з'єднань чотири, і такі налаштування, як keep-alive чи тайм-аут, стосуються кожного з них окремо. HTTP також вимагає, щоб проксі себе називав: він мусить додавати поле заголовка Via до кожного повідомлення, яке пересилає, і саме так у трасуванні видно петлю чи несподіваного посередника.

HTTP — лише один рівень. Проксі SOCKS, версію 5 якого описує RFC 1928 від березня 1996 року, пересилає TCP-з’єднання та UDP-датаграми, не читаючи їхнього вмісту, тож ним може користуватися будь-який протокол. Відомий приклад — Tor: за замовчуванням він приймає на порту 9050 SOCKS-з’єднання від локальних програм.

Застосуйте поняття до рішення

Застосуйте «Що таке проксі?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Wi-Fi, фаєрволи і віддалений доступ налаштовані так, щоб працювати можна було звідусіль, а заходити — лише своїм.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Керувала 30 інстансами Ubuntu Linux VPS — 30 серверів Linux під управлінням, із планом відновлення

Керувала 30 інстансами Ubuntu Linux VPS Зв'язатися з нами →

Чому це важливо

Хто керує проксі, той може читати трафік. Для звичайного HTTP це кожен заголовок і кожне тіло повідомлення. Крізь тунель CONNECT він бачить ім’я хоста й порт, а решта — це TLS від клієнта до сервера. Зворотний проксі, що завершує TLS, бачить усе відкритим текстом, і це його робота, бо щоб маршрутизувати запит, його треба прочитати. Тож поставити перед сайтом зворотний проксі іншої компанії, а саме це й робить мережа доставки контенту (CDN), означає віддати тій компанії адресу кожного відвідувача й кожен URL раніше, ніж їх побачить сам сайт. Для інтернет-крамниці під атакою це може бути слушний обмін. Проте це однаково рішення про приватність, і сторінка про приватність має про нього сказати.

Другий наслідок — те, що дізнається сервер. RFC 7239 описує втрату прямо: за проксі запити видаються такими, що надходять з адреси проксі. Журнал, обмеження частоти запитів, список блокування чи список дозволених адрес, що спираються на адресу клієнта, бачать тоді одну адресу на всіх. Звичне виправлення — заголовок, який несе початкову адресу далі: стандартне поле Forwarded з RFC 7239 або старіший X-Forwarded-For. Котрій копії цього заголовка вірити — саме з цього питання починається більшість проблем.

Третій наслідок: проксі змінює значення помилки. Відповіді 502 Bad Gateway та 504 Gateway Timeout пише посередник про сервер, що стоїть за ним: проксі отримав недійсну відповідь або не отримав її вчасно. Код 407 означає,

що облікові дані хоче сам проксі, перш ніж щось переслати. Знати, яка машина написала помилку, — це половина діагнозу.

Четвертий: багато проксі кешують. RFC 9111 називає кеш, що зберігає відповіді для більш ніж одного користувача, спільним; зазвичай він є частиною посередника. Такий кеш робить сайт швидшим і дешевшим в обслуговуванні, і водночас саме в ньому відповідь, призначена одній людині, може дістатися іншій.

Як це виглядає на практиці

Три фрагменти роботи компанії обертаються навколо проксі: один поставили перед сайтом і знову прибрали, інші належать стороннім і принесли б свої заголовки до журналу, а ще один вимкнули навмисно.

У серпні 2026 року компанія перенесла свою DNS-зону з приблизно двадцяти записів до репозиторію й узгодила її з API Cloudflare. Частиною цієї роботи було пустити вебтрафік крізь мережу доставки контенту провайдера як зворотний проксі, а потім робочий CDN-проксі перед сайтом розібрали заради приватності. Заперечення стосувалося його місця: адреса кожного читача й кожен запитаний шлях спершу потрапляли б до третьої сторони й оброблялися б за її правилами, а для компанії, яка каже читачам, що за ними ніхто не стежить, це важило більше, ніж захист від атак, який він давав. Рішення записане разом з обґрунтуванням, тож його не відкриють знову випадково.

Місяцем пізніше компанія почала рахувати читачів за власним журналом вебсервера, і довелося зважати на проксі, якими вона не керує. Журнал доступу викидає сім заголовків, що передають адресу клієнта, ще до запису рядка. Логер копіює заголовки без змін, тож у читача за корпоративним проксі незамаскована адреса інакше лягла б у файл під назвою заголовка, якої правило маскування не охоплювало.

Того ж місяця сайт опублікували як onion-дзеркало в мережі Tor, і власний проксі-порт демона Tor для вихідного трафіку в конфігурації відключено. Доступ до onion-сервісу відбувається через ланцюжки, які демон встановлює назовні, тож для публікації цього дзеркала не потрібен окремий вхідний порт з інтернету. Оскільки SOCKS-слухач вимкнено, цей демон Tor не надає локальним застосункам проксі для вихідних з’єднань.

Де виникають проблеми

Усі запити в журналі приходять з однієї адреси. Застосунок стоїть за зворотним проксі й записує адресу з'єднання, яке отримав, тобто адресу проксі. Обмеження частоти тоді гальмує всіх одразу, а блокування, записане засобом виявлення вторгнень, блокує проксі, тобто всю аудиторію. Нехай проксі передає адресу клієнта в заголовку, а застосунок читає цей заголовок лише на з'єднаннях із власної адреси проксі. У nginx для цього є директива set_real_ip_from модуля realip; більшість вебфреймворків мають аналогічне налаштування довірених проксі.

Клієнт обирає собі адресу, надіславши заголовок. Застосунок читає X-Forwarded-For у кожному запиті, зокрема в тих, що дійшли до нього напряму, і в тих, що прибули з підробленим значенням у заголовку від самого початку. RFC 7239 прямо каже, що поле може змінити будь-який вузол на шляху, клієнт теж. Довіряйте лише записам, які додав ваш власний проксі, рахуючи справа, і прив'яжіть застосунок до loopback або приватної мережі, щоб проксі був єдиним входом.

Журнал, який мав бути анонімним, зберігає повні адреси. Адресу клієнта замасковано в полі, яке хтось перевірив, але та сама адреса їде в Forwarded, X-Forwarded-For, X-Real-IP та подібних заголовках, і логер, що записує набір заголовків у тому вигляді, в якому їх отримав, зберігає їх повністю. Складіть перелік усіх заголовків, що можуть нести адресу, відкиньте або замаскуйте кожен у момент запису рядка, а тоді надішліть тестовий запит і прочитайте рядок на диску.

Сторінки повертають 502 або 504, а власний журнал бекенда мовчить. Помилку написав проксі. Код 502 означає, що він отримав недійсну відповідь: бекенд не працював, слухав на іншому порту або говорив звичайним HTTP там, де проксі чекав TLS, чи навпаки. Код 504 означає, що бекенд відповідав довше за тайм-аут читання проксі, який у nginx за замовчуванням становить 60 секунд. Спершу прочитайте журнал помилок проксі, бо там названо адресу upstream, яку він пробував, і збільшуйте тайм-аут лише для маршрутів, про які відомо, що вони повільні.

Користувач бачить сторінку іншого користувача. Спільний кеш у проксі чи в мережі доставки контенту зберіг персональну відповідь, наприклад сторінку облікового запису, віддану разом із сесійним cookie, і віддав її наступному, хто запитав той самий URL. Позначайте кожну відповідь, що

стосується окремого користувача, директивою Cache-Control: private, бо таку відповідь, за RFC 9111, спільний кеш зберігати не повинен, або no-store там, де копію не варто зберігати ніде, і перевірте заголовки, які проксі повертає насправді.

Що почитати далі

- RFC 9110, section 3.7: Intermediaries — визначення проксі, шлюзу, тунелю та проксі-перехоплювача.
- RFC 9110, section 9.3.6: CONNECT — як клієнт просить проксі відкрити тунель до хоста й порту.
- RFC 7239: Forwarded HTTP Extension — стандартний заголовок для початкової адреси клієнта та його застереження.
- RFC 9111: HTTP Caching — спільні й приватні кеші та директиви, що їх розмежовують.
- RFC 1928: SOCKS Protocol Version 5 — протокольно нейтральний проксі, що пересилає TCP-з’єднання й UDP.

Словник

Обирайте за фактичними умовами

Опишіть шлях розгортання, експлуатаційні ризики та чергову команду. Ми можемо визначити зміну з перевірками, шляхом відновлення та чіткою відповідальністю.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке проксі?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

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

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/proxy/