Engineer

プロキシ

利用者に代わってリクエストを転送するサーバー

どのようなものか

プロキシは、クライアントと、そのクライアントが接続したいサーバーの間に置くプログラムです。クライアントのリクエストを受け取り、次のマシンへ自分自身のリクエストを送り、その応答を返します。クライアントはプロキシと通信し、プロキシはサーバーと通信するため、両端が同じ接続を共有することはありません。プロキシにはいくつかの種類があり、それぞれを理解する手掛かりは、誰が設置したかという点です。

HTTP自身の仕様であるRFC 9110は、一般的な仲介者を3種類挙げています。狭義のプロキシはクライアントが選んだもので、通常はブラウザー設定や環境変数など、手元の設定で指定します。RFCがリバースプロキシとも呼ぶゲートウェイはサーバー側に置かれます。接続してくる側にはオリジンサーバーとして振る舞いながら、各リクエストを背後にある1台以上のサーバーへ転送します。トンネルは2つの接続間で内容を解釈せずにバイトを中継し、開通後はやり取りの当事者ではなくなります。クライアントはCONNECTメソッドでホストとポートを指定してトンネルを要求し、その中でサーバーまでエンドツーエンドのTLS通信を行います。RFCは、誰も選んでいない4つ目の種類も挙げています。透過プロキシと呼ばれることの多いインターセプションプロキシです。公衆アクセスポイントや企業のファイアウォールのように、ネットワークの境界でポート80への外向き通信を振り向けます。

経由する各区間は、別々の接続です。RFCの図ではブラウザーとオリジンサーバーの間に3つの仲介者があり、接続は4本になります。keep-aliveやタイムアウトなどの設定が適用されるのは、そのうちの1本ずつです。HTTPはプロキシに、自身の存在を知らせることも求めます。転送するすべてのメッセージへViaヘッダーフィールドを追加しなければならず、これによってループや予想外の仲介者が通信記録に現れます。

HTTPは1つの層にすぎません。SOCKSプロキシはTCP接続とUDPデータグラムの内部を読まずに中継するため、どのプロトコルでも使えます。そのバージョン5は、1996年3月のRFC 1928で規定されています。よく知られた例がTorで、標準ではローカルアプリケーションからのSOCKS接続をポート9050で待ち受けます。

概念を判断に生かす

「プロキシとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 Wi-Fi、ファイアウォール、リモートアクセスを整え、どこからでも働け、権限のある人だけが接続できるようにします。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 Ubuntu Linux VPSインスタンス30台を管理 — 30 復旧計画とともに管理したLinuxサーバー

Ubuntu Linux VPSインスタンス30台を管理 お問い合わせ →

なぜ重要なのか

プロキシを運用する側は通信を読めます。平文HTTPなら、すべてのヘッダーと本文です。CONNECTトンネルではホスト名とポートが見え、それ以外はクライアントからサーバーまでのTLS通信になります。TLSを終端するリバースプロキシには、すべてが平文で見えます。振り分け先を決めるためにリクエストを読む必要があるので、それが仕事でもあります。したがって、コンテンツ配信ネットワークのように他社のリバースプロキシをサイトの前に置くと、サイト自身が受け取るより先に、訪問者全員のアドレスと各URLをその会社へ渡します。攻撃を受けている店舗には、妥当な選択かもしれません。それでもプライバシーに関わる判断であり、プライバシーページにはそのことを書くべきです。

2つ目の影響は、サーバーが何を知るかです。RFC 7239は失われる情報を明確に説明しています。プロキシの背後では、リクエストがプロキシのアドレスから来たように見えます。そのため、クライアントアドレスをキーにしたログ、レート制限、拒否リスト、許可リストには、全員が同じアドレスとして映ります。一般的な対処は、元のアドレスを引き継ぐヘッダーを付けることです。RFC 7239の標準フィールドForwardedか、より古いX-Forwarded-Forを使います。そのヘッダーのどの記載を信じるかが、問題の大半の出発点です。

3つ目に、プロキシはエラーの意味を変えます。502 Bad Gatewayや504 Gateway Timeoutは、背後のサーバーについて仲介者が書くものです。プロキシが不正な応答を受け取ったか、時間内に応答を受け取れなかったことを示します。407は、何かを転送する前にプロキシ自身が認証情報を求めているという意味です。どのマシンがそのエラーを書いたかを知るだけで、診断の半分は進みます。

4つ目に、多くのプロキシはキャッシュを持ちます。RFC 9111は、複数の利用者向けに応答を保存するものを共有キャッシュと呼び、通常は仲介者の一部だとしています。サイトの配信を速く安くできる一方、1人向けの応答が他人へ届いてしまう場所でもあります。

実際の活用例

2026年8月、同社は約20レコードのDNSゾーンをリポジトリへ移し、Cloudflare APIを使って実際の状態と整合させました。事業者のCDNをリバースプロキシとしてウェブ通信を通すことも、その作業に含まれていました。その後、サイトの前で実際に動くCDNプロキシを、プライバシーのために撤去しました。問題は、その置かれる位置でした。読者全員のアドレスと要求したすべてのパスが、先に第三者へ届きます。「読者は監視されない」と伝える会社にとって、それは得られる攻撃対策より重い代償でした。

翌月、ウェブサーバーがすでに持つアクセスログから読者を数えるには、同社が運用していないプロキシにも対処する必要がありました。アクセスログは、クライアントのアドレスを引き継ぐ7つのヘッダーを、行を書き込む前に破棄します。ログのエンコーダーはヘッダーを変更せずコピーします。この処理がなければ、企業のプロキシを使う読者の完全なアドレスが、マスキング規則の対象外のヘッダー名で保存されていました。

同じ月にサイトへTorのonionミラーを加えた際には、Torデーモン自身が外向き通信向けに提供するプロキシポートを、設定で無効にしました。onionサービスへは、デーモンが外向きに確立する回線を通じてアクセスするため、このミラーの公開用に別のインターネット向け受信ポートは必要ありません。SOCKSの待ち受けを無効にしているため、このTorデーモンはローカルアプリケーションへ外向き通信のプロキシを提供しません。

問題が起きるところ

ログの全リクエストが同じアドレスから来る。 アプリケーションがリバースプロキシの背後にあり、受け取った接続のアドレスを記録しています。それはプロキシのアドレスです。レート制限は全員を一度に抑え込み、侵入検知ツールの拒否設定はプロキシ、つまり読者全員を拒否します。プロキシにクライアントアドレスをヘッダーで渡させ、アプリケーションはプロキシ自身のアドレスから来た接続だけでそのヘッダーを読むよう設定します。nginxではrealipモジュールのset_real_ip_fromディレクティブがこれにあたり、多くのウェブフレームワークにも、信頼するプロキシを指定する同等の設定があります。

クライアントがヘッダーを送って自分のアドレスを選ぶ。 アプリケーションがすべてのリクエストでX-Forwarded-Forを読んでおり、直接届いたリクエストも、すでに偽の値を入れられたリクエストも含めて信用しています。RFC 7239は、クライアントを含む経路上のどのノードでもフィールドを変更できると明言しています。右端から数えて、自分たちのプロキシが追加した項目だけを信頼します。また、アプリケーションをループバックかプライベートネットワークで待ち受けさせ、プロキシが唯一の入口になるようにします。

匿名化したはずのログに完全なアドレスが残る。 誰かが確認したフィールドではクライアントアドレスが隠されていても、同じアドレスがForwarded、X-Forwarded-For、X-Real-IPなどのヘッダーで運ばれています。受け取ったヘッダーの集合をそのまま書くロガーは、完全なアドレスを保存します。アドレスを運べるヘッダーをすべて洗い出し、行を書く時点でそれぞれを削除するか隠します。そのうえでテストリクエストを送り、ディスク上の行を読みます。

ページは502や504を返すのに、バックエンド自身のログは静かなまま。 エラーを書いたのはプロキシです。502は不正な応答を受け取ったことを示します。バックエンドが停止していた、別のポートで待ち受けていた、プロキシがTLSを期待したところへ平文HTTPを話していた、またはその逆だった、といった原因です。504はバックエンドの応答がプロキシの読み取りタイムアウトを超えたことを示し、nginxの既定値は60秒です。まずプロキシのエラーログを読みます。そこには接続を試みた上流のアドレスが記録されています。タイムアウトを延ばすのは、遅いと分かっているルートだけにします。

ある利用者に別の利用者のページが表示される。 プロキシやコンテンツ配信ネットワークの共有キャッシュが、セッションCookieとともに返されたアカウントページなど、個人向けの応答を保存し、同じURLを次に要求した人へ渡しています。利用者ごとの応答にはすべてCache-Control: privateを付けます。RFC 9111は、共有キャッシュがこの応答を保存してはならないと定めています。どこにもコピーを残すべきでなければno-storeを使い、プロキシが実際に返すヘッダーを確認します。

参考資料

- RFC 9110、第3.7節：仲介者 — プロキシ、ゲートウェイ、トンネル、インターセプションプロキシの定義です。
- RFC 9110、第9.3.6節：CONNECT — クライアントがホストとポートへのトンネルを開くようプロキシへ要求する方法です。
- RFC 7239：Forwarded HTTP拡張 — 元のクライアントアドレスを伝える標準ヘッダーと、その注意点を説明します。

- RFC 9111：HTTPキャッシュ — 共有キャッシュとプライベートキャッシュ、および両者を分けるディレクティブです。
- RFC 1928：SOCKSプロトコルのバージョン5 — TCP接続とUDPデータグラムを中継する、上位プロトコルを限定しないプロキシの仕様です。

用語集

実際の制約に合わせて選ぶ

デプロイの流れ、運用上のリスク、緊急対応の担当者をお知らせください。検査、復旧手段、明確な責任の所在を伴う変更の範囲を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「プロキシとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/proxy/