Engineer

Tor

Netz, das Verkehr über freiwillig betriebene Relays leitet

Was es ist

Tor verbirgt vor Beobachtern der Leitung, wer mit wem spricht. Dazu leitet es jede Verbindung durch eine Kette von Maschinen, sodass keine zugleich Herkunft und Ziel des Verkehrs erfährt. Die Kette wählt der Client, ein Programm namens tor auf dem eigenen Rechner des Lesers. Sie besteht gewöhnlich aus drei Relays: einem Guard als wiederholt genutztem Eingang, einem mittleren Relay und einem Exit, der die letzte Verbindung zum Ziel herstellt. Der Client baut diesen Circuit Abschnitt für Abschnitt auf und vereinbart mit jedem Relay einen eigenen Schlüssel. Jedes Datenstück erhält dann pro Abschnitt eine Verschlüsselungsschicht. Jedes Relay entfernt seine eigene Schicht und erfährt nur den vorhergehenden und den folgenden Nachbarn. Diese verschachtelten Schichten sind die Zwiebel im Titel des Entwurfsaufsatzes The Second-Generation Onion Router. Im allgemeinen Sprachgebrauch bilden alle Relays zusammen das Tor-Netzwerk. tor mit kleinem Anfangsbuchstaben ist das Programm, Tor Browser der dafür zusammengestellte Browser.

Daten bewegen sich in Zellen fester Größe. Ein Relay sieht deshalb einen Strom gleichförmiger Blöcke statt der Form der ursprünglichen Pakete. Mehrere TCP-Datenströme können einen Circuit gemeinsam nutzen, wenn ihre Isolationsanforderungen vereinbar sind. Der Client baut Circuits sowohl im Voraus als auch bei Bedarf und nimmt sie abhängig von Alter und Nutzung außer Betrieb. Ein langlebiger Datenstrom kann auf seinem ursprünglichen Circuit bleiben; manche Isolationseinstellungen erlauben eine längere Wiederverwendung. Anwendungen erreichen den Client über einen lokalen SOCKS-Proxy, und nur TCP reist auf diesem Weg. Der Entwurfsaufsatz sagt ausdrücklich, dass Protokolle wie UDP nicht getunnelt werden. Freiwillige betreiben die Relays. Jedes veröffentlicht einen Deskriptor mit Schlüsseln, Adresse, Bandbreite und Exit-Richtlinie. Eine kleine Gruppe von

Verzeichnisinstanzen stimmt darüber ab, welche aufgenommen werden, und signiert das Ergebnis als Konsensdokument. Jeder Client ruft es vor der Pfadauswahl ab.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist Tor?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. WLAN, Firewalls und Fernzugriff werden so eingerichtet, dass Mitarbeitende von überall arbeiten können und nur Berechtigte Zugang erhalten.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. 30 Ubuntu-Linux-VPS-Instanzen verwaltet — 30 verwaltete Linux-Server mit geplantem Wiederherstellungsweg

30 Ubuntu-Linux-VPS-Instanzen verwaltet Kontakt aufnehmen →

Warum es wichtig ist

Für Betreiber einer öffentlichen Website geht es bei Tor zunächst um die Leser. Ein Leser über Tor scheint von der Adresse eines Exit-Relays zu kommen statt von seiner eigenen. Die Relay-Liste ist absichtlich öffentlich, weil jeder Client sie zur Pfadauswahl braucht. Eine Website kann Tor-Verkehr daher erkennen und anders behandeln. Das Tor Project gesteht Betreibern ausdrücklich das Recht zu, ihn zu sperren, auch wenn es hofft, dass sie es nicht tun. Offen bleibt die Entscheidung über diese Leser: wie alle anderen bedienen, verlangsamen, vor eine Prüfaufgabe stellen oder ihnen einen Zugang anbieten, der das Tor-Netzwerk nie verlässt. Letzteres tut eine Onion-Adresse.

Es lohnt sich, Tors Versprechen genau zu benennen, denn der Entwurfsaufsatz nennt auch seine Grenzen. Tor beansprucht keinen Schutz gegen Beobachter, die beide Enden einer Verbindung sehen und den zeitlichen Verlauf und die Datenmenge vergleichen können. Die Projektdokumentation ergänzt, dass der Forschung kein praktischer Entwurf mit geringer Latenz bekannt ist, der das zuverlässig leistet. Das Grundprotokoll verbirgt die Tor-Nutzung selbst nicht. Bridges mit Pluggable Transports ergänzen Möglichkeiten, diesen Verkehr zu tarnen. Tor verschlüsselt auch nicht das letzte Stück vom Exit-Relay zum Ziel. Für diesen Abschnitt braucht die Anwendung eigene Verschlüsselung.

Im Vergleich zu einem VPN ist das Vertrauen anders verteilt. Ein VPN-Anbieter sieht Nutzer und Ziel und soll wegsehen. Bei Tor sieht kein einzelnes Relay beides. Für die meisten Pfade halten die Auswahlregeln Relays derselben angegebenen Familie auseinander und vermeiden standardmäßig ein

gemeinsames IPv4-/16- oder IPv6-/32-Subnetz. Dokumentierte Ausnahmen und konfigurierbare Subnetzgrenzen gibt es. Der Preis ist Latenz: Ein gewöhnlicher Exit-Circuit durchläuft drei freiwillig betriebene Maschinen. Für Tor-Leser eignen sich deshalb leichte Seiten mit wenigen Netzwerk-Roundtrips am besten. Eine Onion-Adresse anzubieten braucht einen tor-Prozess und einen zu schützenden privaten Schlüssel, aber keinen offenen eingehenden Port. Deshalb eignet sich das für kleine Infrastrukturen.

Wie es in der Praxis aussieht

Die eigene Website des Unternehmens ist über Tor lesbar. Die Kommandozeilenseite veröffentlicht die Onion-Adresse neben einem `curl` -Befehl, der den Markdown-Text über einen lokalen tor-Client abruft. Die Adresse verwendet bewusst einfaches `http://` : Eine Onion-Adresse der Version 3 kodiert den öffentlichen Identitätsschlüssel des Dienstes zusammen mit einer Prüfsumme und einem Versionsbyte. Tor authentifiziert den Dienst und verschlüsselt den Circuit, bevor HTTP gesprochen wird. Für diese Authentifizierung über die Onion-Adresse ist kein öffentliches Webzertifikat nötig.

Ein Onion-Dienst brauchte Platz für einen tor-Prozess auf dem Host der öffentlichen Unternehmensdienste vom September 2026: ein Kern, ungefähr 464 MB nutzbarer Speicher, bereits TLS-Terminierung für sieben Domains und eine laufende Git-Forge. Der Platz wurde gemessen statt ausdiskutiert. An einem dokumentierten Tag mit Stichproben auf diesem Host mit 512 MB lag der geringste beobachtete verfügbare Speicher bei ungefähr 310 MB von 464. Die Swap-Nutzung betrug 2,6 Prozent, und der Prozessor war 99,7 Prozent der Zeit untätig. Tor kam deshalb neben den übrigen Diensten hinzu.

Die Datenschutzerklärung hält die Entscheidung des Unternehmens zum Zählen der Leser fest. Die Gemini- und Gopher-Anwendungen führen keine Zugriffsprotokolle; ihre täglichen Zähler erfassen nur die Gesamtzahl der Anfragen. Für Gopher gibt es außerdem ein Verbindungsprotokoll des Betriebssystems, das die Quelladresse enthält und nach 30 Tagen gelöscht wird. Für Tor erklärt das Unternehmen, überhaupt nichts zu zählen und auch keine Einstellung dafür anzubieten. Das ist eine Datenschutzentscheidung des Betreibers: Tor verbirgt die Netzwerkadresse eines Lesers, hindert einen Dienst aber nicht daran, Anfragen zu zählen. Laut Erklärung will das Unternehmen lieber nichts wissen, als es wissen zu können.

Wo es schiefgeht

Der Name wird außerhalb von Tor aufgelöst. Ein Programm, das seine Verbindungen an einen SOCKS-Proxy schickt, kann den Hostnamen trotzdem zuerst selbst über den gewöhnlichen Resolver auflösen. Diese Anfrage verrät dem lokalen Netzwerk und dem DNS-Anbieter die besuchte Website. Die Verbindung reist dann durch Tor, aber das Ziel ist bereits offengelegt. In `curl` macht eine einzige Option den Unterschied: `--socks5` löst den Namen lokal auf, `--socks5-hostname` übergibt ihn dem Proxy, und das Exit-Relay löst ihn auf.

```
# Resolves the name locally, then connects through the proxy
curl -s --socks5 127.0.0.1:9050 https://example.org/
# Hands the name to tor; the exit relay looks it up
curl -s --socks5-hostname 127.0.0.1:9050 https://example.org/
```

Klartext nach dem Exit. Tor verschlüsselt die Abschnitte zwischen Relays, nicht den letzten. Der Exit öffnet eine gewöhnliche TCP-Verbindung zum Ziel und sendet, was die Anwendung ihm gegeben hat. Ein über unverschlüsseltes HTTP abgeschicktes Anmeldeformular ist deshalb für den Exit-Betreiber und jeden im Netz dahinter lesbar. Alles, was das Tor-Netzwerk verlässt, über HTTPS ausliefern. Bei einem Onion-Dienst stellt sich die Frage nicht, weil sein Verkehr das Netzwerk nie verlässt.

Eine Adresse steht für viele Leser. An die Clientadresse gebundene Ratenlimits und Sperren behandeln ein Exit-Relay wie einen einzigen Besucher, obwohl viele Menschen es gleichzeitig nutzen. Ein auf eine Person abgestimmtes Limit sperrt alle aus, deren Circuit zufällig dort endet. Eine durch einen missbräuchlich handelnden Nutzer ausgelöste Sperre trifft ebenfalls alle. Die Behandlung von Exit-Adressen bewusst entscheiden, statt sie einer allgemeinen Regel pro Adresse zu überlassen.

Die Identität reist im Circuit mit. Tor verbirgt die Netzwerkadresse und ändert nichts an den Informationen, die die Anwendung übermittelt. Eine Anmeldung beim persönlichen Konto, eine auffällige Browserkonfiguration oder ein Dokument, das beim Öffnen etwas nachlädt, verknüpft die Sitzung mit einer Person, unabhängig von der Zahl der Relays dazwischen. Die Projektdokumentation merkt ausdrücklich an, dass Anwendungsverkehr auf Nutzerseite weiter bereinigt werden muss. Diese Aufgabe übernimmt Tor Browser, ein gewöhnlicher auf den Proxy eingestellter Browser dagegen nicht.

Ein Exit-Relay am Heimanschluss. Guard- und mittlere Relays ziehen selten Missbrauchsbeschwerden nach sich. Auf der Zielseite erscheint der Exit als Absender des Verkehrs. Beschwerden und gelegentliche rechtliche Anfragen gehen deshalb an dessen Betreiber. Die Relay-Anleitung des Tor Project rät davon ab, eines zu Hause zu betreiben. Freiwillige mit nur einer privaten Anschlussadresse können stattdessen eine Bridge betreiben: ein nicht öffentlich gelistetes Relay, das dieselbe Anleitung für diese Situation empfiehlt.

UDP soll funktionieren. Sprachanrufe, viele Spiele und Werkzeuge mit eigenen DNS-Anfragen nutzen UDP. Tor befördert nur TCP-Datenströme. Diese Programme funktionieren darüber nicht. Wird ein Programm nicht zur Nutzung des Proxys gezwungen, verbindet es sich womöglich einfach direkt, außerhalb von Tor und ohne Hinweis.

Weiterführende Lektüre

- Tor Specifications — die Spezifikationen für Protokoll, Pfadauswahl, Verzeichnis und Onion-Dienste an einem Ort.
- Tor: The Second-Generation Onion Router — der Entwurfsaufsatz: Circuits, Zellen, Exit-Richtlinien und eine ausdrückliche Liste dessen, was der Entwurf nicht versucht.
- Tor Path Specification — wie ein Client Guard, mittleres Relay und Exit auswählt und welche Relays keinen Pfad teilen dürfen.
- Types of relays on the Tor network — Guard-, mittlere, Exit- und Bridge-Relays mit den jeweiligen rechtlichen Risiken für den Betreiber.
- HTTPS encryption and Tor — eine interaktive Darstellung dessen, was ein Mitleser mit Tor, mit HTTPS, mit beidem und mit keinem von beiden sieht.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Bereitstellungsweg, die Betriebsrisiken und die Bereitschaftsverantwortlichen. Wir können eine Änderung mit Prüfungen, einem Wiederherstellungsweg und klaren Zuständigkeiten abgrenzen.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist Tor?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

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

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/tor/