Engineer

Netzwerkport

Nummer, die auf einem Host einen Dienst auswählt

Was es ist

Ein Netzwerkport ist eine 16-Bit-Zahl im Header jedes TCP- und UDP-Segments, die dem empfangenden Rechner sagt, zu welchem seiner Programme die Daten gehören. Die Adresse bringt ein Paket zum richtigen Host; der Port bringt es dort zum richtigen Prozess. Für eine genaue Bezeichnung sprechen Fachleute von einer Portnummer. Mit Ports im Plural ist gewöhnlich die Menge der Ports gemeint, an denen ein Server lauscht. Das Wort bezeichnet hier weder einen physischen Anschluss an einem Switch noch das Portieren von Software auf eine andere Plattform.

Da das Feld 16 Bit breit ist, gibt es 65.536 Werte von 0 bis 65535. TCP und UDP haben jeweils eine eigene Menge: TCP-Port 53 und UDP-Port 53 sind zwei getrennte Dinge, die DNS beide nutzt. Die TCP-Spezifikation RFC 9293 definiert einen Socket als Internetadresse zusammen mit einem Port und eine Verbindung als Paar von Sockets. Daraus ergeben sich vier Werte: Client-Adresse, Client-Port, Server-Adresse und Server-Port. Das erklärt auch, warum ein Server-Port Tausende Verbindungen zugleich tragen kann. Jede unterscheidet sich in mindestens einem der anderen drei Werte, meist im Client-Port.

RFC 6335 teilt den Bereich in drei Teile. Die Ports 0–1023 sind System Ports, 1024–49151 sind User Ports und 49152–65535 sind Dynamic Ports, die IANA nie vergibt. IANA führt das Verzeichnis, das einem Dienstnamen eine Zahl zuordnet: 22 für SSH, 53 für DNS, 443 für HTTPS und 5432 für PostgreSQL. Eine Registrierung ist eine Konvention, auf die Clients sich verlassen. Jedes Programm kann aber an jeder freien Nummer lauschen. So landet SSH auf einem Port, den niemand erraten würde.

Die Client-Seite einer Verbindung bekommt gewöhnlich einen Port, den sie nie selbst gewählt hat. Baut ein Programm eine ausgehende Verbindung auf, wählt das Betriebssystem einen Port aus seinem kurzlebigen, sogenannten ephemeren Bereich. Linux verwendet standardmäßig 32768–60999 statt des dynamischen Bereichs der IANA. Eine Bindung an Port 0 anzufordern bedeutet dasselbe: irgendeinen freien Port zu erhalten.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist ein Netzwerkport?“ 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

Hinter jedem Port steht die Entscheidung über die Erreichbarkeit: Welche Programme beantworten Anfragen von außen und unter welcher Adresse? Ein Port ohne lauschendes Programm weist eine Verbindung ab. An einem Port mit lauschendem Programm ist Software samt ihren Fehlern erreichbar, mit der jeder sprechen kann, der die Adresse erreicht. Am wenigsten Aufwand verursacht die Absicherung eines Dienstes, der an 127.0.0.1 lauscht: Pakete anderer Rechner können die Loopback-Adresse nicht erreichen, unabhängig von der Firewall.

Die erste Frage für jeden Dienst lautet deshalb, ob er überhaupt von außen erreichbar sein muss. Eine Datenbank, mit der nur die Anwendung auf demselben Rechner spricht, gehört auf Loopback. Ein nur von Administratoren verwendetes Werkzeug kann auf Loopback bleiben und über einen Sprunghost oder ein VPN erreicht werden. Was übrig bleibt, üblicherweise 80 und 443 für eine Website und ein Port für SSH, ist die öffentliche Angriffsfläche. Sie klein zu halten ist ebenso eine Kosten- wie eine Sicherheitsentscheidung: Jedes öffentlich lauschende Programm braucht Patches, Protokollierung und eine passende Firewall-Regel.

Die zweite Frage ist die Nummer. Standardports verursachen keinen Konfigurationsaufwand: Browser, Clients und restriktive Netzwerke erwarten 443 und lassen den Verkehr durch. SSH von 22 zu verschieben verringert das

Rauschen von Scannern, die nur den Standard versuchen. Das macht Protokolle lesbarer, aber ein Scan über den gesamten Bereich findet den Dienst trotzdem. Außerdem muss jedes Werkzeug, das auf den Port verweist, die neue Nummer lernen. Genau diesen letzten Punkt vergessen Teams.

Die dritte Frage betrifft Privilegien und gemeinsame Nutzung. Unter Linux braucht ein Prozess root oder die Capability CAP_NET_BIND_SERVICE, um einen Port unter 1024 zu binden, sofern ip_unprivileged_port_start nicht vom Standard 1024 abgesenkt wurde. Gewöhnlich gehört eine bestimmte Kombination aus Adresse und Port einem Programm. Zwei Dienste, die beide 443 wollen, brauchen deshalb etwas davor, das den Verkehr aufteilt, meist einen Reverse-Proxy, der den vom Client angefragten Hostnamen liest.

Wie es in der Praxis aussieht

Als das Unternehmen im Juli 2026 seine eigene Git-Forge einrichtete, wurde der SSH-Listener der Forge an Loopback auf Port 23231 gebunden. Wer von außen kommt, erreicht ihn über einen Sprunghost. So blieb die Zahl der öffentlichen Ports unverändert. Ein Multiplexer, der HTTPS und Git über SSH an einem öffentlichen Port anhand der ersten Bytes jeder Verbindung bedienen würde, ist geschrieben und ausgeschaltet. Auch ein Listener ohne Nutzer müsste verteidigt werden.

Im August 2026 stellte eine Härtungsprüfung fest, dass SSH bei der ersten Härtung des Hosts auf Port 1986 verschoben worden war, während die Jail zur Sperrung von Angreifern weiterhin 22 überwachte. Jede Sperre der Jail blockierte einen Port ohne lauschenden Dienst. Der Schutz hatte somit noch nie gegriffen. Die Korrektur war klein. Geblieben ist eine Regel: Jeder Sicherheitskontrolle wird eine Assertion zugeordnet, die ihre Wirkung tatsächlich prüft.

Der Host hat 464 MB Arbeitsspeicher und kein Dashboard. Fragen zu seinem Zustand beantworten deshalb sieben nur lesende Berichtsrollen, die den laufenden Rechner in Markdown abbilden. Der Faktenbericht listet die lauschenden Sockets auf. Das ist der Portbestand, den der Kernel tatsächlich führt, statt des Bestands, den die Konfiguration beschreibt. Zu den tatsächlichen Funden der Berichte gehörte ein lauschender Port, den keine Deklaration erwähnte.

Wo es schiefgeht

Ein nur für den eigenen Rechner bestimmter Dienst antwortet dem ganzen Internet. Viele Programme lauschen standardmäßig an jeder Adresse, in der Socket-Liste als `0.0.0.0`, `[::]` oder `*` dargestellt. Eine eigentlich lokale Datenbank oder Verwaltungsoberfläche ist dann nur einen Firewall-Fehler von öffentlicher Erreichbarkeit entfernt. Setzen Sie die Bind-Adresse ausdrücklich auf `127.0.0.1` oder `::1` und bestätigen Sie sie in der Spalte der lokalen Adresse:

```
ss -tlnp
ss -ulnp
```

Ein Schutz überwacht weiterhin einen Port, den der Dienst verlassen hat. Der Dienst ist auf eine neue Nummer umgezogen, aber die Firewall-Regel, die sperrende Jail, die Überwachungsprobe oder eine Client-Konfiguration nennt noch die alte. Jedes dieser Werkzeuge arbeitet weiterhin korrekt gegen einen Port, an dem kein Daemon lauscht. Halten Sie die Nummer in einer einzigen Variablen, die alle Verbraucher lesen. Ergänzen Sie eine Prüfung, die den lauschenden Socket mit dem Ziel jeder Kontrolle vergleicht.

Ausgehende Verbindungen scheitern unter Last mit „Cannot assign requested address“. Jede ausgehende Verbindung belegt einen lokalen Port aus dem ephemeren Bereich. Nach dem Schließen bleiben ihre vier Werte noch eine Weile in TIME_WAIT reserviert. Mit dem Linux-Standardbereich 32768–60999 kann eine Client-Adresse 28.232 offene oder kürzlich geschlossene Verbindungen zu derselben Server-Adresse und demselben Port halten. Ein Proxy oder Worker, der pro Anfrage eine neue Verbindung zu einem Backend öffnet, verbraucht sie alle. Verwenden Sie zuerst Verbindungen über Keep-alive oder einen Pool erneut. Ein größerer Bereich oder zusätzliche Quelladressen schaffen lediglich Spielraum.

Ein als gewöhnlicher Nutzer gestarteter Dienst endet an Port 80 mit „Permission denied“. Eine Bindung unter 1024 erfordert Privilegien. Geben Sie dem Dienst über seine Unit-Datei die Capability CAP_NET_BIND_SERVICE, legen Sie ihn auf einen hohen Port hinter einem Reverse-Proxy oder senken Sie auf einem Host, auf dem jeder lokale Nutzer vertrauenswürdig ist, ip_unprivileged_port_start. Den gesamten Dienst für einen einzigen Port als root auszuführen ist die ungünstigste dieser Möglichkeiten.

Der Start endet mit „Address already in use“. Ein anderer Prozess hält den Port, häufig eine alte Instanz desselben Dienstes, die noch herunterfährt, oder ein zweiter Dienst mit demselben Standardport; 8080 ist beliebt. `ss -tlnp` nennt den belegenden Prozess. Legen Sie fest, welchem Dienst die Nummer gehört, und weisen Sie jedem Dienst in seiner Konfiguration explizit einen Port zu, statt sich auf Standardwerte zu verlassen.

Weiterführende Lektüre

- RFC 9293, Abschnitt 4: Glossar — die eigenen Definitionen der TCP-Spezifikation für Port, Socket und Verbindung.
- RFC 6335: IANA-Verfahren zur Verwaltung des Verzeichnisses für Dienstnamen und Transportprotokoll-Portnummern — die drei Portbereiche und die Vergabe ihrer Nummern.
- Verzeichnis für Dienstnamen und Transportprotokoll-Portnummern — die IANA-Liste der Dienste und der jeweils registrierten Nummern.
- RFC 768: User Datagram Protocol — die dreiseitige UDP-Spezifikation einschließlich Quell- und Zielports.
- RFC 6056: Empfehlungen zur Zufallsauswahl von Transportprotokoll-Ports — wie ein Client seine ephemeren Ports so wählen sollte, dass sie schwer zu erraten sind.
- IP Sysctl — Einstellungen des Linux-Kernels, darunter ip_local_port_range und ip_unprivileged_port_start.

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 ein Netzwerkport?“ 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/port/