Engineer

Порт

число, що вибирає службу на вузлі

Що це таке

Порт — це 16-бітне число в заголовку кожного сегмента TCP і UDP, за яким машина-одержувач визначає, якій зі своїх програм призначені дані. Адреса доводить пакет до потрібного вузла, а порт — до потрібного процесу на ньому. Коли хочуть точності, кажуть «номер порту»; у множині, «порти сервера», зазвичай мають на увазі ті, на яких він слухає. Фізичний порт на комутаторі й портування програм на іншу платформу — зовсім інші речі, хоч і звучать схоже.

Поле має ширину 16 біт, тож значень 65 536, від 0 до 65535, і TCP та UDP мають кожен свій набір: TCP-порт 53 і UDP-порт 53 — дві окремі речі, які DNS використовує обидві. Специфікація TCP, RFC 9293, визначає сокет як інтернет-адресу, поєднану з портом, а з'єднання — як пару сокетів. Виходить чотири значення: адреса клієнта, порт клієнта, адреса сервера, порт сервера. Це й пояснює, чому один порт сервера може обслуговувати тисячі розмов одночасно: кожна відрізняється хоча б одним із трьох інших значень, найчастіше портом клієнта.

RFC 6335 ділить діапазон на три частини. Порти 0–1023 — це System Ports, 1024–49151 — порти користувача, а 49152–65535 — Dynamic Ports, яких IANA ніколи не призначає. IANA веде реєстр, що пов'язує назву служби з номером: 22 для SSH, 53 для DNS, 443 для HTTPS, 5432 для PostgreSQL. Реєстрація — це домовленість, на яку покладаються клієнти, а слухати на будь-якому вільному номері може будь-яка програма. Так SSH і опиняється на порту, якого ніхто не вгадає.

Клієнтський бік з'єднання зазвичай отримує порт, якого сам не обирав. Коли програма підключається назовні, операційна система бере його зі свого діапазону ефемерних портів, і Linux за замовчуванням використовує 32768–

60999, а не динамічний діапазон IANA. Запит прив’язати порт 0 означає те саме: дай будь-який вільний.

Застосуйте поняття до рішення

Застосуйте «Що таке порт?» 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

Рішення за кожним портом стосується відкритості: які програми відповідають ззовні і на якій адресі. Порт без слухача повертає відмову. Порт зі слухачем — це програма, з якою може говорити кожен, хто дістається до адреси, разом з усіма її вадами. Найдешевше захистити службу, що слухає на 127.0.0.1, бо пакети з інших машин до адреси loopback не потрапляють, хоч би що казав брандмауер.

Тож перше запитання для кожної служби — чи має вона взагалі бути досяжною ззовні. База даних, з якою говорить лише застосунок на тій самій машині, має слухати на loopback. Інструмент, яким користуються тільки адміністратори, може жити на loopback за проміжним хостом (jump host) або VPN. Те, що лишається, — зазвичай 80 і 443 для вебсайту й один порт для SSH, — і є публічна поверхня. Тримати її короткою — рішення про витрати не менше, ніж про безпеку, бо кожного публічного слухача треба оновлювати, журналювати й прикривати відповідним правилом брандмауера.

Друге запитання — номер. Стандартні порти нічого не коштують у налаштуванні: браузери, клієнти й суворі мережі очікують 443 і пропускають його. Перенесення SSH з 22 зменшує шум від сканерів, які пробують лише стандартний порт, і журнали стає легше читати, проте сканування всього діапазону однаково його знайде, а кожен інструмент, що згадує порт, мусить дізнатися новий номер. Саме про останнє команди й забувають.

Третє — привілеї та спільне використання. У Linux, щоб прив’язати порт нижче 1024, процесу потрібен root або можливість CAP_NET_BIND_SERVICE, якщо тільки ip_unprivileged_port_start не знижено від стандартного значення

1024. І зазвичай певною адресою й портом володіє одна програма, тож дві служби, яким обом потрібен 443, мусять мати попереду щось, що розділить трафік. Здебільшого це зворотний проксі, який читає ім'я хоста, запитане клієнтом.

Як це виглядає на практиці

Три фрагменти інфраструктурної роботи компанії обертаються навколо номерів портів: один про те, де стоїть слухач, другий про порт, що переїхав, і третій про те, як дізнатися, які порти насправді відкриті.

Коли в липні 2026 року компанія розгорнула власний git-сервер, його SSH-слухача прив'язали до loopback на порту 23231, а ззовні до нього потрапляють через проміжний хост, тож кількість публічних портів не змінилася. Мультиплексор, який віддавав би HTTPS і git через SSH з одного публічного порту, розпізнаючи перші байти кожного з'єднання, написаний і вимкнений: слухача без користувачів однаково довелося б захищати.

У серпні 2026 року перевірка захисту виявила, що демон SSH перенесли на порт 1986 ще під час першого посилення захисту сервера, а механізм, що блокує зловмисників, і далі стежив за 22. Кожне блокування, яке він записував, стосувалося порту, де ніщо не слухало, тож захист жодного разу не спрацював. Виправлення було невеликим. Тривалим наслідком стало правило: кожен засіб безпеки отримує перевірку-твердження (assertion), яка його тестує.

Сервер має 464 МБ пам'яті й жодної панелі моніторингу, тож на його запитання відповідають сім ролей, які лише читають живу машину й описують її у файлах Markdown. Звіт із фактами містить перелік сокетів, що слухають, тобто той список портів, який насправді тримає ядро, а не той, що описаний у конфігурації. Серед реальних знахідок звітів був порт, що слухав, хоча жодна декларація його не згадувала.

Де виникають проблеми

Служба, призначена для самої машини, відповідає всьому інтернету. Багато програм за замовчуванням слухають на всіх адресах, що в списку сокетів видно як `0.0.0.0`, `[::]` або `*`, і база даних чи адмінпанель, яка

мала бути локальною, опиняється за одну помилку брандмауера від відкритого доступу. Явно задайте адресу прив'язки `127.0.0.1` або `::1` і перевірте її в стовпці локальної адреси:

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

Захист стежить за портом, з якого служба вже пішла. Службу перенесли на новий номер, а правило брандмауера, механізм блокування, проба моніторингу чи конфігурація клієнта досі називають старий. Кожен із цих інструментів і далі коректно працює — з портом, на якому не слухає жоден демон. Тримайте номер в одній змінній, яку читають усі споживачі, і додайте перевірку, що порівнює сокет, який слухає, з тим, за чим стежить кожен засіб захисту.

Вихідні з'єднання під навантаженням починають падати з «Cannot assign requested address». Кожне вихідне з'єднання забирає локальний порт з ефемерного діапазону, а закрите з'єднання ще якийсь час тримає свої чотири значення в стані TIME_WAIT. Зі стандартним для Linux діапазоном 32768–60999 одна адреса клієнта може мати 28 232 з'єднання з тією самою адресою й портом сервера, відкриті або щойно закриті. Проксі чи воркер, що відкриває нове з'єднання на кожен запит до одного бекенда, вичерпує їх. Спершу повторно використовуйте з'єднання через keep-alive або пул; ширший діапазон чи додаткові адреси-джерела лише дають запас.

Служба, запущена від звичайного користувача, зупиняється з «Permission denied» на порту 80. Прив'язка нижче 1024 потребує привілеїв. Надайте службі можливість CAP_NET_BIND_SERVICE через її unit-файл, перенесіть її на високий порт за зворотним проксі або, на сервері, де всім локальним користувачам довіряють, знизьте ip_unprivileged_port_start. Запускати всю службу від root заради одного порту — найгірший із компромісів.

Запуск зупиняється з «Address already in use». Порт тримає інший процес: часто стара копія тієї самої служби, яка ще завершується, або інша служба з тим самим стандартним номером (8080 трапляється особливо часто). `ss -tlnp` покаже власника. Вирішіть, якій службі належить номер, і задайте кожній явний порт у конфігурації замість того, щоб покладатися на значення за замовчуванням.

Що почитати далі

- RFC 9293, section 4: Glossary — визначення порту, сокета й з’єднання зі специфікації TCP.
- RFC 6335: IANA Procedures for the Management of the Service Name and Transport Protocol Port Number Registry — три діапазони портів і те, як у них призначають номери.
- Service Name and Transport Protocol Port Number Registry — перелік IANA: яка служба зареєстрована на якому номері.
- RFC 768: User Datagram Protocol — специфікація UDP на трьох сторінках, разом із портами джерела й призначення.
- RFC 6056: Recommendations for Transport-Protocol Port Randomization — як клієнтові обирати ефемерні порти, щоб їх було важко вгадати.
- IP Sysctl — налаштування ядра Linux, серед них ip_local_port_range та ip_unprivileged_port_start.

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 порт?» 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/port/