Engineer

Linux

ядро, на якому працює більшість серверів

Що це таке

Linux — це вільне ядро операційної системи з родини Unix: програма, яка запускається першою, керує обладнанням, розподіляє процесорний час і пам'ять між усіма іншими програмами та вирішує, кому можна торкатися якого файлу й якого мережевого порту. У повсякденній мові цим словом називають і всю операційну систему, зібрану довкола ядра, а дехто пише GNU/Linux, щоб віддати належне інструментам GNU, які зазвичай ідуть разом із ним.

Почав його Лінус Торвальдс 1991 року як клон Unix, написаний з нуля, і відтоді ядро розвиває вільна мережа розробників, що співпрацюють через інтернет. Воно орієнтується на стандарти POSIX і Single UNIX Specification, тож програми з інших Unix-подібних систем зазвичай збираються під нього з невеликими змінами. Починалося ядро на 32-бітному процесорі Intel 386, а нині працює на довгому переліку 32- і 64-бітних архітектур. Код поширюється за ліцензією GPL, лише версії 2, з окремим застереженням: програми, які просто звертаються до ядра через системні виклики, під її дію не підпадають. Саме тому пропрієтарне програмне забезпечення працює на Linux без ліцензійних питань.

Саме ядро — ще не система, у яку можна увійти. Робочій системі потрібні також бібліотека C, оболонка, система ініціалізації, менеджер пакетів і тисячі програм, а проєкти, що збирають усе це в щось придатне до встановлення, називаються дистрибутивами. Debian, Ubuntu, Red Hat Enterprise Linux та його перезбірки, SUSE, Fedora й Alpine — це дистрибутиви. Кожен постачає власну збірку ядра, власні налаштування за замовчуванням і власну обіцянку, як довго виправлятиме вразливості.

З чим порівнюють Linux, залежить від завдання. На серверах — з Windows Server і з родиною BSD, тобто FreeBSD та OpenBSD, які розробляють як єдину цілісну систему, а не складають із частин. На ноутбуках — з Windows і macOS.

Застосуйте поняття до рішення

Застосуйте «Що таке Linux?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Щоденний догляд за серверами, поштою, обліковими записами й оновленнями, щоб те, на що всі покладаються, працювало тихо й без сюрпризів.

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. Підтримувала 10 серверів PostgreSQL і MS SQL — 10 серверів баз даних розгорнуто й підтримувано в справному стані

Підтримувала 10 серверів PostgreSQL і MS SQL Зв'язатися з нами →

Чому це важливо

Для компанії, що запускає програми на власних серверах чи в хмарі, операційну систему здебільшого обрано ще до того, як хтось поставить запитання. Вебсервери, бази даних, рушії контейнерів та інструменти моніторингу створюють насамперед для Linux і найретельніше тестують саме там. Тож рішення, яке лишається, вужче: який саме Linux і коли Windows Server усе-таки виправданий.

Вибір між Linux і Windows Server визначає програмне забезпечення, а не сама система. Бізнес, побудований на Active Directory, IIS чи старіших застосунках .NET, має вагомі причини лишатися на Windows. Майже все інше в сучасному вебстеку, від PostgreSQL і Nginx до Kubernetes, розраховане на Linux, і навіть Microsoft SQL Server має версію для Linux.

З вартістю все просто. Ядро та спільнотні дистрибутиви на кшталт Debian і Ubuntu безкоштовні незалежно від кількості машин, і під час аудиту не доводиться рахувати ліцензії; Windows Server продається за комерційною ліцензією. У випадку Linux гроші йдуть на людей і, за бажання, на підписку на підтримку від постачальника, як-от Canonical, Red Hat чи SUSE.

Контейнери Linux посилюють цей зв’язок. У кожному образі користувацький простір Linux упаковано у файл, а його запуск означає запуск звичайних процесів на ядрі Linux. Тому на Mac чи ноутбуці з Windows інструменти, які запускають ці контейнери, спершу запускають невелику віртуальну машину з

Linux; власний інструмент Apple, `container`, запускає кожен із них в окремій легкій віртуальній машині. Вибір контейнерів Linux додає Linux до програмного стека, хоч би що стояло на ноутбуках розробників.

Насправді два сервери з Linux різняться дистрибутивом і тим, як довго його підтримують. Проєкт ядра випускає нову основну (mainline) версію кожні дев'ять–десять тижнів і роками підтримує кілька довгострокових (longterm) гілок: 6.12, наприклад, за планом отримуватиме виправлення до грудня 2028 року. Дистрибутиви надбудовують над цим власну підтримку, і ядро дистрибутива підтримує сам дистрибутив, а не розробники ядра. Обирайте дистрибутив за строком підтримки й за тим, що команда вже вміє.

Одне правило проєкту ядра робить оновлення менш ризикованими, ніж можна подумати з такого темпу: зміна, яка ламає програму, що раніше працювала, вважається регресією, а регресії виправляють, часто просто відкочуючи зміну. Завдяки цьому давно зібрані програми й далі працюють на нових ядрах.

Зворотний бік цієї свободи в тому, що фраза «у нас Linux» мало що пояснює. Два дистрибутиви можуть мати різні менеджери пакетів, інструменти брандмауера, способи налаштування мережі й ритм випусків, і команді, яка знає один, доведеться частково переучуватися на інший.

Як це виглядає на практиці

Найближче до заліза — один хост із 464 МБ пам'яті, на якому працюють платформа для коду (forge) і вебсервер. Сигнал про нестачу пам'яті чи місця на диску тут дає перевірка за принципом dead man's switch на таймері: кожні п'ятнадцять хвилин, із випадковим зсувом, невеликий скрипт вимірює доступну пам'ять і вільне місце на диску й пінгує зовнішній сервіс лише тоді, коли обидва показники вище порогів. Тиша і є тривогою. Скрипт від початку читав правильне число, доступну пам'ять, а шість імен змінних навколо нього казали «free». Оскільки ядро віддає незайняту пам'ять під кеш, вільна пам'ять на здоровій машині з Linux тримається біля нуля: з порогом у десять відсотків читач побачив би дванадцять мегабайтів вільними на машині з 464 МБ, а «виправлення» в один символ перетворило б робочу тривогу на таку, що спрацьовує постійно. Імена виправили. Отриманий сигнал підтверджує, що скрипт виконався, а перевірки доступної пам'яті й вільного місця на диску пройдено за встановленими порогами; він не засвідчує загальної

справності хоста чи того, що сайт обслуговує запити. Моніторинг може виявляти тривожні ознаки, хоча час їх появи залежить від навантаження й умов спостереження.

Під час попередньої роботи засновниця створила першу панель спостережуваності однієї організації, яка збирала по SSH дані про CPU, RAM, HDD, температуру та інші метрики з віддалених серверів на Linux і так само стежила за контейнерами Docker. Друга версія перейшла на Grafana і Prometheus. Панель висіла на великому телевізорі в офісі зі сповіщеннями про перевищення порогів, і команди почали виявляти й усувати інциденти на 40% швидше.

У масштабі всієї інфраструктури доступ став тим, що записано, а не тим, що хтось пам'ятає. Сервери Linux були однією із систем, на які поширювалися 1 000 правил RBAC, поряд із середовищами розробників, екземплярами застосунків та базами даних PostgreSQL і MS SQL, а Ansible послідовно застосовував правила в різних середовищах. Розробники могли звертатися лише до призначених їм серверів. Результатом роботи стали задокументований набір правил на основі ролей та повторюваний спосіб їх застосування й перевірки.

Де виникають проблеми

Моніторинг показує майже заповнену пам'ять на цілком здоровому сервері. Linux вважає незайняту пам'ять змарнованою і заповнює її кешем файлів, який віддає, щойно програма попросить місця. Показник «free» враховує лише пам'ять, що взагалі нічим не зайнята, тож на будь-якій навантаженій машині він сповзає до нуля. На запитання «чи витримає сервер більше роботи?» відповідає MemAvailable — власна оцінка ядра, скільки пам'яті можуть отримати нові програми без свопу. Ставте тривогу саме на нього:

```
grep -E '^(MemFree|MemAvailable):' /proc/meminfo
```

Сервіс зникає, і пояснення доводиться шукати в журналі ядра, а не в журналі самого сервісу. За замовчуванням Linux дозволяє програмам резервувати більше пам'яті, ніж є на машині, з розрахунком, що значної її частини так ніхто й не торкнеться. Коли розрахунок не справджується і пам'ять справді закінчується, OOM killer обирає процес, зазвичай великий, і

завершує його. Застосунок не встигає нічого записати; слід лишається в журналі ядра, який показує `journalctl -k` . Задайте кожному сервісу власну межу пам'яті через `MemoryMax=` у systemd або ліміт контейнера, щоб програму, яка вийшла з-під контролю, зупинили раніше, ніж ядру доведеться обирати, і стежте за доступною пам'яттю, а не чекайте на примусове завершення.

Звіт про помилку, надісланий розробникам ядра, лишається без відповіді. На машині працює ядро дистрибутива з його власними патчами, а розробники ядра підтримують лише власні випуски. `uname -r` показує версію, і все після дефіса, як у `6.8.0-45-generic` , означає збірку дистрибутива. Спершу повідомте про проблему дистрибутиву або відтворіть її на актуальному ядрі з kernel.org, і лише тоді несіть її розробникам ядра.

Оновлення безпеки перестають надходити, а сервер працює далі як завжди. Кожен випуск дистрибутива й кожне довгострокове ядро мають дату завершення підтримки, і сама машина про неї не попереджає. Довгострокові гілки 5.10 і 5.15, наприклад, за планом дійдуть до неї в грудні 2026 року. Тримайте дату завершення підтримки в тому самому переліку, що й назву сервера, і плануйте оновлення на один випуск наперед, а не в останній момент.

Що почитати далі

- About Linux Kernel — коротке пояснення від самого проєкту ядра: що таке Linux і чому новачкові потрібен дистрибутив.
- Releases — основні, стабільні й довгострокові ядра та запланована дата завершення кожної довгострокової гілки.
- The /proc Filesystem — де визначено MemFree, MemAvailable та інші лічильники пам'яті.
- Overcommit Accounting — три режими overcommit і що кожен із них обіцяє програмі.
- Reporting regressions — правило «жодних регресій» і як повідомити про поломку, щоб її врахували.
- Linux kernel licensing rules — лише GPL-2.0 і застереження про системні виклики, яке звільняє програми над ядром від ліцензії.

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 Linux?» 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/linux/