Engineer Платформа

Ваш код на вашому власному сервері

Розгорнула власний git-сервер компанії на Soft Serve, приватний за замовчуванням і без вебпанелі, з портом SSH, прив'язаним до loopback за проміжним хостом (jump host), і зробила цільову сторінку перед ним артефактом збірки основного сайту, а не копією, яку підтримують вручну.

липень 2026

приватний git-сервер із портом, прив'язаним до loopback

Ситуація. Вихідний код компанії зберігався в сторонньому хостинг-сервісі — розумне для нього місце, але погано пасує бізнесу, головний аргумент якого для клієнтів — те, що він не передає їхніх даних посередникам. А мати власний git-сервер означає справді його тримати: автентифікація, контроль доступу, сховище, резервні копії й публічне обличчя для всього цього.

Завдання. На наявному хості треба було розгорнути канонічний git-сервер із якнайменшою поверхнею атаки й без вебпанелі адміністрування, а перед ним — цільову сторінку, яка не псуватиметься з часом.

Дія. Git-сервер — це один бінарний файл, який запускає й контролює операційна система, встановлений зі сховища пакунків постачальника; його свідомо обрано за те, що він працює насамперед через SSH і не має адміністративного вебінтерфейсу: що менше інтерфейсу, то менше треба захищати. Він приватний за замовчуванням: анонімний доступ заборонено, доступ без ключа заборонено, і кожен задекларований репозиторій позначено як приватний, а не покладаються на те, що про нього ніхто не знає. Його SSH-сервер слухає лише loopback-інтерфейс на порту 23231, а ззовні до нього потрапляють через проміжний хост, тож задекларована поверхня брандмауера не зростає. Мультиплексор протоколів, який розводив би HTTPS і git через SSH на одному публічному порту за першими байтами з’єднання, написано й підготовлено за головним вимикачем, і цей вимикач вимкнено: багатокористувацький git через публічний порт поки не потрібен, а порт, яким ніхто не користується, — це теж поверхня атаки. Цікавішою задачею виявилася цільова сторінка перед

ним. Вона була копією дизайну головного сайту, яку вручну підтримували в окремому репозиторії, і кожна відмінність між ними, будь-коли знайдена, виявилася випадковістю, а не рішенням: типографічна шкала, через яку текстовий логотип виходив приблизно на дев'ять відсотків завеликим, оголошення шрифту, яке на будь-якій машині зі встановленою гарнітурою зводило дві насиченості до однієї, орнаменти, приховані нижче певної ширини, тож на жодному телефоні їх не було, насиченість, для якої не постачався шрифт, і жодна сторінка не мала ні орієнтира main, ні заголовка верхнього рівня. П'ять із п'яти, і жодної не видно на знімку екрана. Копію видалено; тепер цільову сторінку збирають шаблони самого головного сайту, а постачається вона як артефакт.

Результат. Компанія тримає власний код на власному обладнанні, а сторінка перед ним успадковує всі перевірки, які проходить головний сайт, а не розходиться з ним так, що це видно лише вимірюванням. Розійтися з ним досі можна, але тепер таке рішення треба записати.

Застосуйте цей досвід у своєму проєкті

Використайте «Ваш код на вашому власному сервері» як відправну точку для схожого завдання у вашій організації. Послуги, що стоять за цією роботою, доступні й для вашого проєкту. Уся конфігурація записана у файлах, тож її можна відтворити точно, і вона не живе лише в чиїйсь голові.

Переглянути наші послуги Зв'язатися з нами →

Дізнайтеся, що привело до результату

Прочитайте про ситуацію, рішення та результат цього прикладу. Ми можемо обговорити, що відповідає вашим умовам, а що потребує іншого підходу. Ваш код на вашому власному сервері — приватний git-сервер із портом, прив'язаним до loopback

Переглянути наші послуги Зв'язатися з нами →

Хочете тримати свій код на власній інфраструктурі?

Обговоріть свій проєкт з інженером, відповідальним за виконання. Обсяг, основу розрахунку вартості та передачу результатів узгоджуємо письмово до початку робіт.

Роль у цій роботі: Системний адміністратор Linux. Тут задіяні Веброзробка, DevOps, Інфраструктура, Автоматизація та CI/CD, Linux та сервери та Системне адміністрування. І, до речі, ми добре вміємо Розробка вебсайтів та CMS, Infrastructure as Code, DevOps та автоматизація CI/CD та Системне адміністрування.

Зв'язатися з нами Ознайомтеся з нашими послугами

Одна частина **довшого переліку** — і він увесь на цьому сайті.

Кожен запис написано однаково: ситуація, завдання, що ми зробили і що змінилося.

Переглянути весь перелік

Визначте потрібне покращення

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 →

Хочете тримати свій код на власній інфраструктурі?

Запитайте про рішення в роботі «Ваш код на вашому власному сервері». Ви говоритимете з людьми, які можуть пояснити роботу й оцінити ваш проєкт.

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://platform.engineer.company/uk/portfolio/your-code-on-your-own-hardware/