**Engineer** Платформа

{ Безпека та керування доступом }

# Визначити, хто до чого має доступ

Облікові записи, паролі й права налаштовані так, щоб доступ відповідав роботі, а звільнені втрачали його того ж дня.

## Ще визначаєтесь, що саме вам потрібно?

Розкажіть, що йде не так або що ви намагаєтесь побудувати. Ми скажемо, що для цього потрібно, скільки це приблизно коштуватиме і чи ми ті люди, які вам потрібні.

**Від 1 400 DKK за годину, без ПДВ — і дешевша година на довшому замовленні.**

**Зв'язатися з нами** Дивитися, скільки це коштує

Вибрані роботи, що демонструють цю послугу.

### Перетворіть послугу на результат

Нехай «Безпека та керування доступом» слугує меті бізнесу. Ми можемо пов'язати технічну роботу з користувачами, експлуатацією та рішеннями, які вона має підтримувати. Облікові записи, паролі й права налаштовані так, щоб доступ відповідав роботі, а звільнені втрачали його того ж дня.

**Зв'язатися з нами**

### Розгляньте відповідні роботи

Використайте роботи на цій сторінці, щоб оцінити наш підхід. Запитайте про компроміси та передумови кожного результату. Написала 1 000 правил RBAC для Linux-серверів — 1,000 правил доступу написано, щоб усунути вразливість

**Написала 1 000 правил RBAC для Linux-серверів** Зв'язатися з нами →

### Написала 1 000 правил RBAC для Linux-серверів

Посилила безпеку даних, впровадивши 1 000 правил RBAC для розробників, інстансів застосунків, PostgreSQL, MS SQL та інших Linux-серверів, запобігши несанкціонованому доступу; задокументувала їх як автоматизацію в Ansible.

### Керувала доступом у M365 і SSO

Запобігла порушенням безпеки, очоливши ініціативи з керування доступом, використовуючи M365, 1Password, Red Hat SSO та OKTA SSO.

### Керувала 8 з'єднаннями WireGuard VPN та IPsec VPN

Керувала 8 з'єднаннями WireGuard VPN та IPsec VPN й усувала в них несправності, забезпечивши безпечний зв'язок між системами Google Cloud та Linux.

### Покращила комунікацію та співпрацю в команді

Покращила комунікацію та співпрацю в команді, впровадивши Slack, Mattermost, 1Password та Jira, заощадивши 8 000 людино-годин.

### Адмініструвала мережеву інфраструктуру

Адмініструвала мережеву інфраструктуру для понад 1 000 серверів, забезпечивши оптимальне розгортання систем, безпеку та усунення несправностей.

### Автоматизувала SSL/TLS для 100 застосунків Docker

Автоматизувала створення сертифікатів SSL/TLS для 100 застосунків Docker, забезпечивши безпечні з'єднання на хостах Ubuntu Linux.

### Побудувала перемикання контексту організації

Спроєктувала систему перемикання контексту організації з клієнтським localStorage та серверним дзеркалюванням cookie, що дала змогу користувачам діяти від імені керованих організацій, забезпечуючи авторизацію за принципом найменших привілеїв.

### Побудувала пошту з failover і журналом доставки

Побудувала електронну пошту як можливість платформи — три провайдери з резервним перемиканням, webhooks доставки, логування надсилання й доставки, шаблонізацію та кампанії — з перевіркою під час запуску, яка не дасть застосунку стартувати без жодного налаштованого поштового провайдера.

### Розгорнула інфраструктуру Azure як код

Розгорнула інфраструктуру Azure як код за допомогою Bicep — Container Apps, PostgreSQL Flexible Server, Front Door/WAF та мережу — у середовищах development, staging і production.

### Посилила захист застосунку CSP на основі nonce

Посилила захист застосунку за допомогою CSP на основі nonce, HSTS, cookie SameSite, ролей бази даних за принципом найменших привілеїв та серверних повторних перевірок прав доступу.

### Автоматизувала командну співпрацю

Автоматизувала командну співпрацю, керування паролями, управління завданнями та часом, а також створила напівавтоматичну систему демонстрації проєктів, підвищивши продуктивність команди.

### Впровадила корпоративний менеджер паролів

Інтегрувала загальнокорпоративну систему керування паролями, посиливши безпеку та оптимізувавши керування доступом.

### Доступ через Stripe та покупки в застосунку

Побудувала рівень платежів і прав доступу — Stripe поряд із внутрішньозастосунковими покупками Apple і Google — обмеживши каталог, пошук та експорт моделлю доступу з 11 таблиць, яку перевіряє сервер.

### Побудувала ліміти запитів і захист від зловживань

Побудувала засоби протидії зловживанням за принципом fail-closed — 22 обмежувачі частоти на Redis, Cloudflare Turnstile, ідемпотентність запитів та прив'язку до origin — щоб платформа відсікала ботів і напливи, а не довіряла тим, хто її викликає.

### Побудувала процес заявок на володіння компанією

Побудувала наскрізний процес заявок на володіння компанією — користувач заявляє права на компанію, адміністратор ухвалює рішення, а схвалення переписує граф авторизації, який визначає, кому що дозволено редагувати.

### Запровадила безперервне сканування безпеки

Запровадила безперервну програму безпеки — сканування коду, DAST, перевірки залежностей і вразливостей, генерацію SBOM, пошук секретів та actions, закріплені за SHA, — поряд із 21 письмовим аудитом безпеки.

### Три замки на дверях, і жоден не замкнений

Знайшла й виправила три механізми захисту SSH від перебору паролів, які ніколи не працювали: jail блокування, що стежив за портом 22, тоді як демон слухав порт 1986; обмеження частоти, яке перекривало ширше правило, розташоване вище; і дію блокування, бінарний файл якої так і не знаходився, тож жодне блокування жодного разу не спрацювало.

### Посилила SSH до 24 перевірених директив

Посилила захист SSH: 24 директиви, кожну з яких перевіряє assert, і триетапна валідація — файл-кандидат, зібрана конфігурація, а потім зворотне зчитування самим демоном, — після того як таке зчитування виявило, що запущений сервер мовчки перевизначав дві з двадцяти чотирьох.

### Довела, що блокування вторгнень працює щоразу

Довела працездатність усього ланцюжка блокування вторгнень під час кожного запуску посилення захисту: блокувала зарезервовану тестову адресу, зчитувала створене правило ядра й знімала блокування в гарантованому блоці очищення, тож jail, який перестав працювати, обриває запуск помилкою, а не звітує про справність.

### Перевірила правила фаєрвола за позицією

Перевірила правила фаєрвола за позицією, а не лише за наявністю, зчитуючи пронумерований перелік правил і чинний ланцюжок фільтрації пакетів, бо наявне правило ще не означає, що до нього доходить хоч один пакет.

### Додала секундну попередню перевірку секретів

Додала попередній плей (preflight), який приблизно за секунду проганяє той самий код, що й converge, на локальних секретах оператора, — після того як продакшн-запуск обірвався на дев'ятому завданні, застосувавшись лише наполовину, а налаштування swap уже було записано на робочий хост.

### Описала зону DNS із 20 записів як код

Налаштувала декларативне узгодження зони DNS із 20 записів з API Cloudflare, з окремими точками входу для аудиту та експорту в BIND, а побудувавши проксі CDN, знову вимкнула його з міркувань приватності.

### Посилила ізоляцію systemd для кожного юніта

Знизила оцінку відкритості (exposure) у пісочниці systemd для кожного юніта, який встановлює платформа: служба dead man's switch — з 9,6 UNSAFE до 1,5, відкритий в інтернет git-сервер — з 8,3 EXPOSED до 1,5, — і додала перевірку парсером під час converge, після того як знайшла у трьох шаблонах юнітів директиву з орфографічною помилкою, яку systemd мовчки ігнорував.

### Запустила контейнери на Podman і Quadlet

Розгорнула контейнерну платформу на Podman і Quadlet під systemd, а не на Docker, бо типове публікування портів Docker у мережі bridge може обходити правила UFW, — а для розгортань виділила непривілейованого користувача з однією фіксованою командою замість root.

### Підготувала другий сервер із коду

Автоматизувала підготовку (provisioning) другого сервера в другого хмарного провайдера, створюючи фаєрвол раніше за машину, щоб вона від самого початку була за ним, а фаєрволи обох провайдерів налаштовуються безпосередньо через їхні REST API, без сторонньої колекції.

### Записала правило меж після витоку облікових даних

Записала в репозиторій правило про межі його вмісту після того, як під час реструктуризації туди потрапили інвентар, дозволи фаєрвола та тексти іншої компанії, — і залишила відокремлений у карантин залишок під наглядом сканера секретів, а не виключила його.

### Розділила всі операційні секрети за хостами

Розділила чотири операційні секрети так, щоб кожен хост мав власні, з'ясувавши, що два хости зі спільним dead man's switch сповіщають про збої рідше, ніж два окремі перемикачі, а не частіше, і що спільна парольна фраза резервних копій може відкрити доступ до копій обох хостів навіть в окремих репозиторіях, якщо хтось отримає її та зможе прочитати зашифровані дані й файли ключів обох репозиторіїв.

### Задіяла невикористану половину ядра

Задіяла ту половину ядра обміну повідомленнями, якою застосунок ніколи не користувався, — передавання резервної копії, зникомі повідомлення, редагування та повторне надсилання, підтверджені запрошення, проксі та політику шифрування, — проганяючи кожен тест на справжній бібліотеці без заглушок.

### Знайте ліцензію всього, що випускаєте

Запровадила перевірку сумісності ліцензій 644 крейтів Rust під час кожної збірки й довела, що вона спрацьовує, переписавши ліцензію одного крейта та змістивши зафіксовану ревізію ядра без перегенерації.

### Виміряла 13 адрес без трекінгу

Виміряла тринадцять адрес лише за власним журналом доступу вебсервера — без кукі, без трекерів і без сторонніх сервісів — маскуючи мережу клієнта в момент запису рядка та згортаючи результат у постійний архів із 21 денного показника та 20 місячних фасетів.

### Опублікувала читабельне onion-дзеркало в Tor

Опублікувала сайт як onion-дзеркало в Tor за читабельною адресою, що починається з engineer, з ключем, згенерованим поза хостом, тож він ніколи не потрапив ні до цього репозиторію, ні на ноутбук, і залишила цей вхід єдиним із чотирьох, відвідування якого ніколи не рахуються.

## Потрібна послуга: Безпека та керування доступом?

Оцініть відповідний досвід за пов’язаними роботами. Розкажіть, що потрібно створити; обговоримо обсяг, підхід до виконання та передачу результатів.

**Зв'язатися з нами** Повне портфоліо

Визначте обсяг до замовлення

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 →

# Обговоріть свої вимоги

Розкажіть, що вам потрібно за напрямом «Безпека та керування доступом», зокрема про строки та команду, яка підтримуватиме рішення. Обговоримо практичний шлях уперед.

**Зв'язатися з нами**

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/services/security-access/**