Engineer Software

Sikkerhed

15 resultater

Forbind delene af jeres system

Arbejdet under »Sikkerhed« viser en del af et fungerende system. Vi kan hjælpe med at forbinde den del med resten af jeres drift. Den del, brugerne aldrig ser: hvor data ligger, hvilke regler der gælder, og hvordan andre systemer må tale med den.

Se vores ydelser Kontakt os →

Find en sammenlignelig udfordring

Åbn et eksempel, der ligner jeres situation. Gennemgangen beskriver beslutningerne bag resultatet, ud over teknologiens navn. Har designet og drevet 6 ETL/ELT-pipelines — 6 ETL-pipelines på tre databasemotorer

Har designet og drevet 6 ETL/ELT-pipelines Kontakt os →

Har forbedret GIS-kortydeevne 10x med PostgreSQL

Har gjort en geografisk kortapplikation 10 gange hurtigere gennem en strategisk databaseovergang fra MSSQL til PostgreSQL, herunder selve omlægningen af data, og optimeret behandlingen og datasikkerheden.

Har bygget skift af organisationskontekst

Har designet et system til skift af organisationskontekst med localStorage i klienten og en spejlet cookie på serveren, så brugere kan optræde som de organisationer, de administrerer, mens autorisationen følger mindste privilegium.

Har bygget e-mail med failover og leveringslog

Har bygget e-mail som en funktion i selve platformen — tre udbydere med failover, webhooks for levering, logning af afsendelse og levering, skabeloner og kampagner — sikret af et opstartstjek, så platformen ikke starter uden mindst én udbyder.

Har designet et funktionsbaseret Postgres-datalag

Har designet et funktionsbaseret datalag i PostgreSQL — 1.275 lagrede funktioner fordelt på 34 skemaer — så al læsning og skrivning går gennem en funktion, databasen kan tildele rettigheder til, og ikke direkte gennem en tabel.

Har hærdet applikationen med nonce-baseret CSP

Har hærdet applikationen med nonce-baseret CSP, HSTS, SameSite-cookies, least-privilege-databaseroller og fornyet kontrol af rettigheder på serveren.

Har fastlagt platformens grundbeslutninger

Har fastlagt platformens grundlæggende beslutninger i ugerne efter, at repositoryet blev oprettet i november 2025 — lagdelingen, database-first-dataadgangen og standarden med nul advarsler — og de holder stadig ni måneder senere.

Har bygget hele kerneinfrastrukturen fra bunden

Har som en af de første medarbejdere designet og bygget hele kerneinfrastrukturen og de understøttende processer fra bunden for en SaaS-startup inden for grøn energi og lagt fundamentet for hurtig vækst.

Har bygget adgangsret via Stripe og in-app-køb

Har bygget betalings- og rettighedslaget — Stripe side om side med app-køb via Apple og Google — så adgangen til kataloget, søgningen og eksporten styres af en adgangsmodel på 11 tabeller, der kontrolleres på serveren.

Har bygget egen fejlovervågning og tracing

Har bygget egen fejlovervågning og OpenTelemetry-tracing i stedet for at købe dem — sanitering af payloads, detektion af spikes og regressioner, symbolication og et syntetisk heartbeat — bag 11 operatørvisninger.

Har bygget fail-closed rate limiting mod misbrug

Har bygget fail-closed misbrugskontroller — 22 Redis-baserede rate limiters, Cloudflare Turnstile, idempotens på requests og en origin-lås — så platformen afviser bots og flood-angreb i stedet for at stole på dem, der kalder den.

Har bygget ejerskabskrav og deres afgørelse

Har bygget ejerskabskrav på virksomheder fra ende til anden — en bruger gør krav på en virksomhed, en administrator afgør sagen, og en godkendelse omskriver den autorisationsgraf, der afgør, hvem der må redigere hvad.

Har skrevet en parser, der tjekker hvert FFI-kald

Har skrevet en parser, der læser den rigtige C-header på 7.308 linjer og verificerer hvert kaldssted, hver enum-konstant, og at hver klasse, der ejer en pointer, er final, efter at en håndskrevet pladsholder-header lod kald til tre fjernede funktioner kompilere, linke og crashe.

Har taget den ubrugte halvdel af kernen i brug

Har taget den halvdel af beskedkernen i brug, som applikationen aldrig havde brugt — backupoverførsel, forsvindende beskeder, redigering og gensendelse af beskeder, verificerede invitationer, proxyer og krypteringspolitik — og kørt hver test mod det rigtige bibliotek uden mocks.

Har renset fejlbeskederne for bebrejdelser

Har omskrevet applikationens fejlbeskeder efter en nedskrevet standard for tone, efter at et afvist login gav brugeren skylden for at have tastet forkert, mens udbyderen i virkeligheden krævede en applikationsspecifik adgangskode, og dækket det med en test, der nævner udbyderen.

Har udgivet et læsbart Tor-onion-spejl

Har udgivet websitet som et Tor-onion-spejl på en læsbar adresse, der begynder med engineer, med nøglen genereret uden for værten, så den aldrig nåede dette repository eller en laptop, og ladet den være den ene af fire hoveddøre, der aldrig tælles.

Sæt fokus på den tekniske indsats

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Sæt emnet ind i jeres sammenhæng

Fortæl os, hvordan »Sikkerhed« indgår i jeres system. Vi kan hjælpe med at vælge den relevante ydelse og afgrænse arbejdet.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://software.engineer.company/da/categories/security/