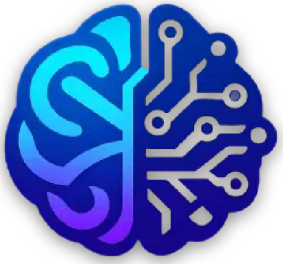
Vi driftar plattformarna **er mjukvara rullas ut på** — uppbyggda från kod, övervakade och nedskrivna.

Moln, infrastruktur som kod, CI/CD, containrar, övervakning, säkerhetskopior, nätverk och åtkomst — från Köpenhamn.



Engineer company

Plattform

Infrastruktur ni kan bygga upp igen från ett repositorium

+45 50 10 61 84 · Köpenhamn, Danmark

Omsorgsfullt skapad med kärlek av D.

Molninfrastruktur som kod, automatiserad driftsättning och återställningsplaner för systemen er verksamhet är beroende av. Arbetsbeskrivningarna visar utgångsläget, vad som gjordes och resultatet.

Börja med en driftsättningsprocess eller en driftrisk. Vi kan avtala en genomgång eller ett genomförande med en dokumenterad plan för driftsättning, återgång till föregående version och ansvar, så att teamet vet vad de ska driva och hur.

Infrastrukturarbete · Priser och samarbetsformer · Beskriv ert uppdrag

Börja med resultatet ni behöver

Berätta vad ert företag behöver uppnå. Vi kopplar samman programvara, data och infrastruktur kring människorna som använder dem.

Se våra tjänster Hör av er →

Se hur vi tar oss an arbetet

Hitta tjänsten som passar ert mål och se exempel på vad som levererades och hur.

Fallstudier Hör av er →

Läs om arbetet **i sitt sammanhang** — besluten, leveransen och de beskrivna resultaten.

Se hela katalogen

En partner för **10 tjänster** från den första servern till larmet klockan tre på natten.

Bläddra i katalogen

{ Moln, infrastruktur & DevOps }

Där mjukvaran körs

Molninfrastruktur och migrering

Servrar dimensionerade efter vad du faktiskt använder, flyttade till molnet eller tillbaka ut ur det, med månadskostnaden uträknad först.

Provisionerade en andra server från kod

Automatiserade provisioneringen av en andra server hos en andra molnleverantör och skapade brandväggen före maskinen så att den föds bakom en, med båda leverantörernas brandväggar skrivna direkt mot deras REST-API:er för att undvika en collection från tredje part.

Körde containrar på Podman och Quadlet

Driftsatte en containermiljö på Podman och Quadlet under systemd i stället för Docker, eftersom Dockers standardpublicering av portar i bridge-nätverk kan kringgå UFW-regler — och gav driftsättningarna en oprivilegierad användare med ett enda fast kommando i stället för root.

Offentliga tjänster på en liten server

Drev företagets offentliga tjänster på en värd med en kärna och 512 MB — en git-forge, en webbserver för sju domäner, Tor, två servrar för alternativa protokoll och intrångsblockering — med 464 MB användbart minne som designbegränsning; backupautomatik är en separat implementation.

Se alla 9 resultat

Så arbetar vi

Förstå först

Vi sätter oss in i problemet innan vi föreslår en lösning. Uppdraget ni kommer med är början på det samtalet, inte slutet.

Bygg för att hålla

Hållbart framför snabbt, klarhet framför finurlighet. Systemet måste fortfarande gå att arbeta i för dem som ärver det.

Stå för det

Vi avtalar eventuell löpande drift eller support som en del av uppdraget. Överlämningen tydliggör ansvar och fortsatt kontaktväg.

{ Moln, infrastruktur & DevOps }

Servrar du kan bygga upp igen från versionshanterad konfiguration

Infrastruktur som kod

Hela uppsättningen nedskriven som filer, så att den kan återskapas exakt och aldrig bara finns i en enda persons huvud.

Publicerade en läsbar onion-spegel i Tor

Publicerade webbplatsen som en onion-spegel i Tor på en läsbar adress som börjar med engineer, med nyckeln genererad utanför värden så att den aldrig nådde detta repo eller en bärbar dator, och lät den förbli den enda av fyra ytterdörrar som aldrig räknas.

Repeterade forgens CI-hook och hittade 2 fel

Repeterade CI-hooken på forge-sidan och hittade två fel som inte gick att se genom att läsa filen: en reservväg som lade ett argument som inte gick att lösa upp i hookens indata, och en miljövariabel från git själv som följde med grinden in i utcheckningen och gjorde 19 tester röda.

Delade upp varje driftshemlighet per värd

Delade upp de fyra driftshemligheterna per värd efter att ha konstaterat att två värdar som delar ett dödmansgrepp larmar mindre än två separata grepp, inte mer, och att en delad lösenfras för säkerhetskopior kan ge tillgång till båda värdarnas kopior, även i separata repon, om någon får tag på den och kan läsa båda reponas krypterade data och nyckelfiler.

Se alla 17 resultat

{ Moln, infrastruktur & DevOps }

Leverera ändringar utan dramatik

DevOps och CI/CD-automatisering

Testade, släppta och möjliga att backa, så att ändringar når kunderna en tisdagseftermiddag i stället för sent en fredagskväll.

Lät 4 repon följa en och samma historikstandard

Lät fyra repon följa en och samma historikstandard — konventionella commits, fria från emojier, utan attributionsrader, upprätthållen av en hook för commit-meddelanden — vid sidan av 48 instruktionsdokument som styr hur arbetet utförs.

Känn till licensen för allt ni levererar

Granskade licenskompatibiliteten hos 644 Rust-crates vid varje bygge och bevisade att kontrollen slår till genom att skriva om licensen för en crate och genom att flytta den låsta kärnrevisionen utan att generera om.

Tog i bruk den oanvända halvan av kärnan

Tog i bruk den hälft av meddelandekärnan som applikationen aldrig hade använt — överföring av säkerhetskopior, försvinnande meddelanden, redigering och omsändning av meddelanden, verifierade inbjudningar, proxyer och krypteringspolicy — och körde varje test mot det riktiga biblioteket utan mockar.

Se alla 43 resultat

{ Moln, infrastruktur & DevOps }

Mjukvara som fungerar likadant överallt

Containerisering och orkestrering

Paketerad för att bete sig identiskt på en laptop, på en testserver och i produktion, och för att skala upp när en hektisk vecka kommer.

Körde containrar på Podman och Quadlet

Driftsatte en containermiljö på Podman och Quadlet under systemd i stället för Docker, eftersom Dockers standardpublicering av portar i bridge-nätverk kan kringgå UFW-regler — och gav driftsättningarna en oprivilegierad användare med ett enda fast kommando i stället för root.

Minskade systemd-exponeringen för varje enhet

Minskade exponeringen i systemd-sandlådan för varje enhet som plattformen installerar — en tjänst för dödmansgrepp från 9,6 UNSAFE till 1,5, en internetexponerad git-forge från 8,3 EXPOSED till 1,5 — och lade till en parserkontroll vid konvergering efter att ha hittat ett felstavat direktiv som tyst ignorerades i tre enhetsmallar.

Byggde CI/CD-pipelines i GitHub Actions

Byggde CI/CD-pipelines i GitHub Actions med en distroless-image för frontenden i produktion och uppflyttning mellan flera miljöer.

Se alla 5 resultat

{ Moln, infrastruktur & DevOps }

Vet före dina kunder

Tillförlitlig drift och övervakning

Övervakning och larm som säger till medan ett fel fortfarande är litet, och siffrorna som visar hur ofta det förblir så.

Fick sex insamlare att ange vad de hoppade över

Fick sex rapporterande insamlare att ange den del av sin indata som de aldrig läste — filer som förblev stängda, sonder som inte installerats, journalrader som inte kändes igen — och anpassade tröskelvärden till 116 mätningar, efter att ha konstaterat att en blind insamling och en värd i gott skick ger samma sida.

Mätte 13 adresser helt utan spårning

Mätte tretton adresser enbart utifrån webbserverns egen åtkomstlogg — ingen cookie, ingen spårare och ingen tjänst från tredje part — maskerade klientens nätverk när varje rad skrivs och förde in resultatet i ett permanent arkiv med 21 dagliga räkningar och 20 månatliga fasetter.

Ett begränsat händelseflöde i en macOS-app

Begränsade händelsebuffertar och loggning efter en dokumenterad macOS-incident med en tillväxt på 41 MB i sekunden och 111 GB komprimerade sidor på en maskin med 36 GB; en dokumenterad fullständig testkörning minskade kärnans loggrader från 610 996 till 1 411.

Se alla 23 resultat

{ Moln, infrastruktur & DevOps }

Få tillbaka allt efter en dålig dag

Säkerhetskopiering och återställning

Säkerhetskopior som testas i stället för att tas för givna, och en skriftlig plan för den morgon då en server inte startar igen.

Delade upp varje driftshemlighet per värd

Delade upp de fyra driftshemligheterna per värd efter att ha konstaterat att två värdar som delar ett dödmansgrepp larmar mindre än två separata grepp, inte mer, och att en delad lösenfras för säkerhetskopior kan ge tillgång till båda värdarnas kopior, även i separata repon, om någon får tag på den och kan läsa båda reponas krypterade data och nyckelfiler.

Backupautomatik med en tydlig återläsningsmetod

Implementerade restic-backupautomatik med lagringspolicy, integritetskontroll och ett flöde för månatliga återläsningstest; dokumenterade aktivering av lagring och kontroll av en verklig återläsning som separata driftsteg.

Byggde säkerhetskopior och återställning som kod

Konfigurerade lagringstiden för databasens säkerhetskopior som infrastruktur som kod, granskade sedan återställningsförmågan och dokumenterade rutinen för återläsning — och pekade ut de återstående luckorna i stället för att låta dem upptäckas under en incident.

Se alla 5 resultat

{ Moln, infrastruktur & DevOps }

Håller allt igång

Systemadministration

Daglig skötsel av servrar, e-post, konton och uppdateringar, så att det som alla förlitar sig på förblir odramatiskt.

Publicerade en läsbar onion-spegel i Tor

Publicerade webbplatsen som en onion-spegel i Tor på en läsbar adress som börjar med engineer, med nyckeln genererad utanför värden så att den aldrig nådde detta repo eller en bärbar dator, och lät den förbli den enda av fyra ytterdörrar som aldrig räknas.

Fick sex insamlare att ange vad de hoppade över

Fick sex rapporterande insamlare att ange den del av sin indata som de aldrig läste — filer som förblev stängda, sonder som inte installerats, journalrader som inte kändes igen — och anpassade tröskelvärden till 116 mätningar, efter att ha konstaterat att en blind insamling och en värd i gott skick ger samma sida.

Repeterade forgens CI-hook och hittade 2 fel

Repeterade CI-hooken på forge-sidan och hittade två fel som inte gick att se genom att läsa filen: en reservväg som lade ett argument som inte gick att lösa upp i hookens indata, och en miljövariabel från git själv som följde med grinden in i utcheckningen och gjorde 19 tester röda.

Se alla 28 resultat

{ Moln, infrastruktur & DevOps }

Kopplar ihop dina kontor och dina medarbetare

Nätverk och VPN-uppsättning

Wi-Fi, brandväggar och fjärråtkomst uppsatta så att personalen kan arbeta varifrån som helst och bara rätt personer kommer in.

Provisionerade en andra server från kod

Automatiserade provisioneringen av en andra server hos en andra molnleverantör och skapade brandväggen före maskinen så att den föds bakom en, med båda leverantörernas brandväggar skrivna direkt mot deras REST-API:er för att undvika en collection från tredje part.

Stämde av en DNS-zon med 20 poster som kod

Stämde av en DNS-zon med 20 poster deklarativt mot Cloudflare-API:t med separata ingångar för granskning och BIND-export, och stängde av CDN-proxyn igen av integritetsskäl efter att ha byggt den.

Verifierade brandväggsregler efter position

Verifierade brandväggsregler efter position snarare än förekomst och läste den numrerade regellistan och den aktiva paketfilterkedjan, eftersom en regel som finns inte är detsamma som en regel som något paket når.

Se alla 11 resultat

{ Moln, infrastruktur & DevOps }

Bestämmer vem som når vad

Säkerhet och behörighetshantering

Konton, lösenord och behörigheter ordnade så att alla har exakt den åtkomst deras arbete kräver, och de som slutar förlorar den samma dag.

Publicerade en läsbar onion-spegel i Tor

Publicerade webbplatsen som en onion-spegel i Tor på en läsbar adress som börjar med engineer, med nyckeln genererad utanför värden så att den aldrig nådde detta repo eller en bärbar dator, och lät den förbli den enda av fyra ytterdörrar som aldrig räknas.

Mätte 13 adresser helt utan spårning

Mätte tretton adresser enbart utifrån webbserverns egen åtkomstlogg — ingen cookie, ingen spårare och ingen tjänst från tredje part — maskerade klientens nätverk när varje rad skrivs och förde in resultatet i ett permanent arkiv med 21 dagliga räkningar och 20 månatliga fasetter.

Känn till licensen för allt ni levererar

Granskade licenskompatibiliteten hos 644 Rust-crates vid varje bygge och bevisade att kontrollen slår till genom att skriva om licensen för en crate och genom att flytta den låsta kärnrevisionen utan att generera om.

Se alla 31 resultat

{ Moln, infrastruktur & DevOps }

Hålla AI:n borta från det den inte ska röra

Säkerhet för AI-agenter

Varje agents behörigheter kartlagda och begränsade till det nödvändiga, varje handling loggad där agenten inte kan ändra loggen, och varje skydd prövat i stället för taget för givet.

Delade upp varje driftshemlighet per värd

Delade upp de fyra driftshemligheterna per värd efter att ha konstaterat att två värdar som delar ett dödmansgrepp larmar mindre än två separata grepp, inte mer, och att en delad lösenfras för säkerhetskopior kan ge tillgång till båda värdarnas kopior, även i separata repon, om någon får tag på den och kan läsa båda reponas krypterade data och nyckelfiler.

Skrev en avgränsningsregel efter läckta uppgifter

Skrev in en avgränsningsregel i repot efter att en omstrukturering hade fört in ett annat företags värdförteckning, brandväggsundantag och text i det — och höll de karantänsatta resterna kvar under skanningen efter hemligheter i stället för att undanta dem.

Kontrollerade hemligheter i förväg på en sekund

Lade till en play för förhandskontroll som kör samma kod som konvergeringen mot operatörens lokala hemligheter på ungefär en sekund, efter att en halvt tillämpad produktionskörning kraschade vid sin nionde uppgift med swapinställningarna redan skrivna till värden i drift.

Se alla 9 resultat

Vi arbetar med **plattformarna** som er verksamhet redan bygger på.



Apple Microsoft DigitalOcean AWS Google

Se alla tekniker

Säkerhetsgenomgång av AI-agenterna som ingår i uppdraget

Kartlagd

En avtalad förteckning över agenterna, inloggningsuppgifterna och dataåtkomsten som ingår i uppdraget, sparad med er konfiguration.

Låst

Varje agent med egen identitet och minsta behörighet, och texten den läser behandlad som data, aldrig som order.

Loggad

Varje handling loggad med personen den gjordes åt, där agenten inte kan ändra den, och larm vid det ovanliga.

Se hur agenter låses Fråga om dina agenter →

Berätta vad ni håller på att bygga.

Inget formulär, ingen tratt, ingen säljsekvens — bara ett e-postmeddelande till en människa som läser det.

Inled samtalet

Den här adressen visar bara infrastrukturarbetet. Företaget bygger också mjukvara, arbetar med data och ger team råd — den sidan av det ligger på engineer.company.
Se allt Engineer ApS gör →

Börja med en avgränsad förbättring

Ett arbetsflöde, ett dataflöde eller ett system som behöver fungera bättre kan vara en bra första uppgift. Vi kan komma överens om omfattningen tillsammans.

Se vad det kostar Hör av er →

Förbered det första samtalet

Berätta vilket resultat ni behöver, vad som redan finns och när ni vill ha det klart. Då kan vi diskutera nästa steg.

Se våra tjänster

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

På servern arbetar vi med **Linux med öppen källkod** — inga licenser att betala och inget dolt.

Ubuntu Debian

Så driftar vi servrar

Ta med dig den här sidan Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://platform.engineer.company/sv/