

Erik Thorsell

DevOps Transformation Leader | Team Lead | Platform Developer

+46 709 86 60 48
erik@requestforcoffee.dev
thorsell.io

Overview

Erik är en lösningssorienterad DevOps-entusiast som tror att bästa sättet att generera värde i ett mjukvaruföretag är att riva murar och hjälpa utvecklarna att leverera ny funktionalitet fortare. Erik har arbetat som Team Lead och linjechef, med ansvar för fyra DevOps Team utspridda över världen, och har också varit ansvarig för att utveckla och underhålla organisationsspännande CI/CD-system. Genom sin tekniska expertis och ledarskapserfarenheter kan Erik hjälpa till med både implementering av utvecklingsplattformar, arkitekturella frågor, och övergripande förändringsarbete i er organisation.

Efter att ha arbetat i flera olika domäner, däribland: automotive, telekom och tull, har Erik en bred kunskapsbas. Han har implementerat cloud-baserade arbetsflöden som nyttjat gitops för att leverera Kubernetes-applikationer till såväl managerade som egen-hostade kluster; men också jobbat med on-premise-system utan Internet-åtkomst där det varit av högsta vikt att leverablerna kan stå på egna ben.

Erik är duktig på att bryta ned komplexa problem i hanterliga bestandsdelar. Han trivs bra i uppdrag med otydliga och föränderliga krav och har ofamnat det agila tankesättet med iterativt förbättringsarbete. Erik sätter sig snabbt in i nya verktyg, miljöer och metoder och letar alltid efter nästa sätt att korta ned feedbackloopen.



Erfarenhet

Dec '22–Dec '23
TOYOTA

DevOps Lead och Platform Developer, Toyota Material Handling Logistics Solutions, Göteborg.

TMHLS utvecklar T-ONE, mjukvaran som gör att Toyotas truckar kör sig själva. T-ONE är en container-iserad .NET applikation med en egenutvecklad regelmotor som bl.a. kan integrera med lagersystem samt kommunicera med OPC UA och HMI.

ROLL

Erik jobbade i ett trepersonsteam som ansvarade för "utvecklarstöd". Han underhöll och förbättrade den befintliga utvecklingsplattformen, som innefattade både molnresurser och fysiska servrar. Han utförde också en migrering av T-ONE från docker compose till Kubernetes, samt tog fram en plan för en ny release-process för TMHLS.

NYTTA

Genom att migrera T-ONE till Kubernetes var Erik en nyckelspelare som möjliggjorde en stor affär för TMHLS. Han hjälpte också till med flera kostnadsreducerande åtgärder, främst genom ett mer effektivt användande av köpta och egen-hostade tjänster.

TEKNIKER

Azure, Azure DevOps, Docker, Kubernetes, Ansible, Terraform, Linux, Windows Server, Python, Bash, Powershell

Nov '21–Dec '22



DevOps Transformation Leader, Maersk Customs Services, Göteborg.

Maersk (tidigare KGH) Customs Services tillhandahåller tull- och frakttjänster över hela världen. De utvecklar och driftar flera SaaS-lösningar som är tillgängliga dygnet runt för att säkerställa att deras kunder kan transportera sitt gods till korrekt destination.

ROLL

När Maersk köpte KGH anlätades Erik för att hjälpa KGH utveckla sin produktkatalog till att kunna köras i Kubernetes. Detta innefattade både tekniskt och organisatoriskt förändringsarbete, där Erik kunde kombinera sina ledarskapserfarenheter och sin tekniska expertis för att ge råd och stöd till KGHs CTO och arkitekter.

NYTTA

Vid uppdragets slut hade Erik lett flera lyckade service-migreringar samt varit med och formulerat en plan för hur migreringen av återstående produkter skulle ske.

TEKNIKER

Azure, Azure DevOps, Terraform, Octopus, Docker, Kubernetes, IIS, .NET, HA-Proxy

May '19–Nov '21



Linjechef, Team Facilitator, Software & CI/CD Engineer, Ericsson (Packet Core), Göteborg.

Avdelningen *Packet Core* ansvarar för Ericssons 5G-utveckling där SWDP är en förutsättning för att kunna utföra "zero downtime"-uppgräderingar av Ericssons produkter.

Jun '21–Nov '21 **Linjechef (Ställföreträdande)**

Som ställföreträdande linjechef hade Erik möjligheten att arbeta med rekrytering, förändringsarbete, och coachning av sina medarbetare i dess karriärer. Han ansvarade för ca 20 medarbetare från olika länder och rapporterade till Packet Core's avdelningschef.

NYTTA

Erik hjälpte teamens medlemmar att prestera bättre genom att tillgodose deras behov. Med hans stöd kunde teamen leverera i enlighet med sina åtaganden och få ett förbättrat arbetssätt gentemot sina intressenter.

Jul '20–Jun '21 **Team Facilitator för DevOps Teams**

Som *Team Facilitator* ansvarade Erik för fyra DevOps Team. Han fokuserade främst på att hjälpa teamen förstå koncept och arbetsmetoder från *Agile* och *DevOps* samt att utmana teamen till att hitta nya sätt att arbeta.

NYTTA

Erik lyckades fostra fram självgående DevOps-teams som tog ägandeskap för sina leverabler. Han var även en nyckelperson i SWDPs arkitektur och produktens CI/CD.

May '19–Jul '20 **Software & CI/CD Engineer**

Erik var en av de första medlemmarna i Packet Core's Continuous Delivery & Deployment-projekt: *SWDP*, en Python-applikation som automatiserar uppgräderingsprocessen för Packet Cores' utan att användarna får någon nedtid.

NYTTA

Initialt bidrog Erik till *SWDPs* utveckling som Python-utvecklare men tog snart ansvar för att drifta och utveckla projektets CI/CD.

TEKNIKER

Kubernetes, Python, Jenkins, GitLab, Gerrit, Spinnaker, Confluence, Jira

Jun '18–May '19



Software & CI/CD Developer, Volvo Autonomous Solutions, Göteborg.

Volvo Autonomous Solutions eftersträvar att transformera hur gods fraktas, genom effektiva, hållbara och säkra autonoma lösningar. I deras produktportfölj finns ett "kontrolltorn" som kommunicerar med alla självkörande fordon.

ROLL

Under sitt uppdrag jobbade Erik initialt som Java-utvecklare och snart även som Scrum Master. Han var del i ett av flera team som jobbade med kontrolltornsmjukvaran. Han tog dock snart ett övergripande ansvar för avdelningens CI/CD-system och jobbade under största del av uppdraget uteslutande med utveckling och drift av detta.

NYTTA

CI/CD-plattformen som Erik utvecklade hjälpa alla utvecklare att hitta integrationsproblem mellan sina komponenter.

TEKNIKER

Jenkins, Java SpringBoot, AMQ, BitBucket, SonarQube, Maven, Confluence, Jira, Mattermost

Sep '17–Dec '17



Machine Learning Engineer, Machine Intelligence Sweden, Göteborg.

MIS grundades av två chalmester med målet att brygga gapet mellan AI/ML-forskning och industrin.

ROLL

Erik jobbade primärt med *Science Router*. Hans roll i projektet var att ta fram verktyg för datainsamling samt strukturera den insamlade datan.

NYTTA

Erik skrev flertalet Python-program som samlade in forskningsdata från olika databaser samt formaterade all insamlad data på ett enhetligt och för domänen lämpligt sätt.

TEKNIKER

AWS, Python

Verktyg & Tekniker



Azure

Av de tre stora molnleverantörerna har Erik doppat tårna i samtliga, men det är Azure han har jobbat mest med. Erik har satt upp IPsec-tunnlar, driftat bygga-genter mha scale-sets, jobbat med IAM och RBAC, AKS, databaser och virtuella maskiner; mestadels mha Terraform, men även genom Microsofts egna verktyg.

Versionshantering



Versionshantering är en självklar del av mjukvaruutveckling och Erik har alltid varit en stor förespråkare av detta. Han har mångårig erfarenhet av att arbeta med olika verktyg för att versionera källkod och binärer – däribland GitHub, GitLab, BitBucket, Azure DevOps och Gerrit – men också av att ta fram strategier för *hur* versioneringen görs på bästa sätt.

Terraform



På samma sätt som det är viktigt att ha sina pipelines och konfigurationer som kod, för att kunna återskapa det man gjort, är det lika viktigt när det kommer till ens infrastruktur. Erik har framförallt använt Terraform.

Boundary



Boundary är ett utmärkt verktyg för att hantera access till en organisations resurser. Erik har implementerat och driftat både Boundary OSS och HCP Boundary i en kontext där resurserna fanns både i Azure och on-prem.

Packer



I en miljö där man kontinuerligt behöver ha uppdaterade virtuella maskiner är Packer ett utmärkt verktyg för att reducera ledtider och få ökad reproducerbarhet. Erik har jobbat med Packer för att bygga olika Linux-distributioner samt Windows Server-versioner, med hjälp av Ansible och Azure.

Ansible



Ansible är för mjukvara vad Terraform är för infrastruktur. Ett utmärkt verktyg för att säkerställa att system hålls uppdaterade och kan återskapas snabbt vid behov. Erik har skrivit Ansible-roller för Ubuntu, RHEL, SLES och Windows Server.

Docker



En stor del av DevOps handlar om självstyre och att ha tydliga avgränsningar mellan team. Docker är ett utmärkt verktyg för att ge team möjligheten att sätta dessa gränser (interface) och arbeta effektivt. Erik har mångårig erfarenhet av Docker, både för att leverera produkter men också som användare.

Kubernetes



Kubernetes ger organisationer möjligheten att bygga och drifta extremt kraftfulla – men komplexa – applikationer. Eriks kuberneteserfarenheter sträcker sig från att ha kunskap om hur man skriver manifest och på ett effektivt sätt driftar sin applikation i ett kluster, till att sätta upp kluster från grunden både i molnet och on-prem.

Python



Balansen mellan att Python lämpar sig väldigt väl för att skriva små script och att det går att skriva fullfjädrade applikationer är svårslagen. Erik använder främst Python i automationssyften för att sy ihop olika delar av ett flöde.

Bash



Det finns tillfällen då Python inte är bästa valet, då vänder sig Erik helst till Bash. Med mångårig Linux-erfarenhet, där Bash många gånger är standard-skalet, ser Erik terminalen som en god vän.

Utbildning

Aug '13–Jun '18



BSc, MSc, Civ.Ing., Datateknik, Chalmers University of Technology, Göteborg.

Erik har en kandidat- och en civilingenjörsexamen i Datateknik. Han har därtill en masterexamen i *Datavetenskap - algoritmer, programspråk och logik* och koncentrerade sina studier till maskininläring, artificiell intelligens och optimering.

Aug '11–Jun '12



Teamträningsskolan Väst (TTS), Dalkarlså Folkhögskola, Dalkarlså.

En teologisk utbildning med fokus på ledarskap. Utbildningen gav en ökad förståelse för komplexiteten som kommer med att tjäna och leda en stor församling.

Nyckelintressen

DevOps | Improvement of Daily Work | Automation | FOSS/Open Source | Transparency
Continuous Integration and Deployment | Agile | Continual Learning | 🏊️ 🚴 🏃