

How a \$4 billion retail giant built an enterprise-grade Kubernetes platform powered by Linkerd

Elkjøp, the largest electronics retailer in the Nordics, built an internal Kubernetes platform that is now successfully hosting over 200 production-grade microservices to increase development speed—without compromising security or visibility. The Linkerd-based platform enabled the organization to reduce hosting costs by around 80%

Backdrop: accelerating retail modernization with microservices

With over 400 retail locations and 12,000 employees across Norway, Sweden, Finland, Denmark, and franchises in Iceland, Greenland, and the Faroe Islands, Elkjøp is the largest electronics retailer in the Nordics. It also has a large e-commerce presence in all these markets.

Although reliant on technology to power its point-of-sale (POS) systems, historically, the IT department was mostly focused on integrating third-party products and externally developed solutions. Five years ago this strategy changed and the team introduced microservices to provide shared functionality between systems and increase development velocity. These included an advanced payment API used by both the e-commerce platform and in-store POS systems.

Initially, Elkjøp hosted these microservices in individual Azure Web Apps, but as the environment grew a new approach was needed. “Azure Web Apps is a great platform for simple systems, but when you start having 70 or 100 copies of web apps it becomes hard to manage and expensive,” said Henry Hagnås, Elkjøp’s Cloud Solution Architect.

On top of this, Elkjøp was about to start an extensive project called “Next-Generation Retail” that would put even more pressure on microservices. Next-Generation Retail would replace Elkjøp’s 20-year-old POS system with a more flexible and scalable solution that allows sales associates to better serve customers by checking if an item was in stock, manage inventory, or performing a sale from a desktop or mobile device.

Elkjøp had an aggressive timeline for the initiative and needed a robust system that would work from day one. After a brief feasibility study, Hagnås’ team engaged Fredrik Klingenberg from [Aurum AS, a Norwegian IT services firm](#), to help them build a modern, enterprise-ready microservices hosting platform.

The imperative of a service mesh approach

The team started the migration by dockerizing and deploying applications onto Kubernetes. But they quickly realized that they lacked the metrics and insight needed to assess performance. Additionally, since they terminated TLS at the ingress controller, all communication between the applications was unencrypted. They needed to solve both problems—and quickly.

To gain visibility into service health and encrypt all service-to-service communication, Hagnås and his team chose [Linkerd, the lightweight, ultra-fast Cloud Native Computing Foundation \(CNCF\) service mesh](#).

Linkerd injects an ultra-lightweight “micro-proxy” as a sidecar for each application. The proxy can offload many cross-cutting concerns such as end-to-end encryption, provide valuable metrics, and give insight into service to service communication—precisely the problems the team needed to solve.

Bringing observability and reliability to Kubernetes, without the complexity

Linkerd was Elkjøp’s choice for several reasons.

Importantly, they wanted a project backed by the CNCF with all of its benefits including a rigorous maturity framework, a community-based commitment to high-quality projects, and technical excellence.

Also, a priority was ease of setup. Within a week, the team had run, tested, and was ready to move forward with Linkerd. “The initial setup was really quick,” said Fredrik Klingenberg. “Overall, it took very few hours to get it up and running and realize value.”

Achieving insight into service health and performance was also critical. Based on experience, Klingenberg knew that debugging a microservices-based app without a service mesh can be hard: “When something isn’t working, it’s hard to know if the problem is with the application, the client, or the underlying network. Sometimes, nothing beats looking at raw network data.”

Elkjøp’s existing approach was to provide that functionality through homegrown tools and libraries. Linkerd was a good fit because it made that functionality readily available across the entire platform. App teams were able to get all those benefits by simply deploying their apps. Actionable service metrics allowed the team to monitor critical performance indicators – success rate, request volume, and latency – for every service.

No more flying blind

This observability that Linkerd delivered was of paramount importance. This was clearly demonstrated by an early incident that occurred as they were preparing for the migration.

Weeks prior to the deployment of the initial POS system to 40 stores across Denmark, a simple load test caused the Kubernetes cluster to fail. “Nothing obvious was wrong with the environment. Something just broke,” said Hagnås.

“We desperately needed insights into what was happening in the cluster and the new microservices architecture. Without the observability that Linkerd gave us, it would have been difficult, if not impossible, to find the source of the problem.”

“WE WERE QUICKLY ABLE TO IDENTIFY IF THE ISSUE WAS WITH THE NETWORK OR NOT. LINKERD SPED UP THE TROUBLESHOOTING PROCESS BECAUSE IT NARROWED DOWN THE OPTIONS AND PREVENTED US FROM FLYING BLIND,” SAID HAGNÅS. USING LINKERD’S OBSERVABILITY TOOLS ELKJØP WAS ABLE TO QUICKLY DIAGNOSE THE PROBLEM, GET A FIX IMPLEMENTED, AND THE PROJECT STAYED ON TRACK.

An interesting side benefit was that when new incidents arise, Hagnås’ team can definitely show application developers if the problem lies in the network or not. “We’ve put an end to the typical cycle of engineers reflexively placing blame on the network whenever there was a problem,” he said.

Security as standard

The security Linkerd brings was also a driving factor behind Hagnås and Klingenberg choice of service mesh.

The team needed a way to provide developers with a base set of functionality and security by just deploying the application onto the platform. By default, Linkerd automatically enables mutual TLS for TCP traffic between meshed pods, by establishing and authenticating secure, private TLS connections between Linkerd proxies. Developers simply add their services to Linkerd, and Linkerd will take care of the rest.

“We wanted to embrace a more aspect-oriented model. We needed to provide individual teams with full autonomy yet ensure there were boundaries in place to protect against mistakes,” Hagnås explained.

Supported by a responsive community

The team also appreciated Linkerd’s community of maintainers and contributors. As Klingenberg explains: “Once we started implementing and using Linkerd, the few times we needed help we found that the community was super friendly, inclusive, and responsive. And the documentation was second to none.”

Linkerd as a critical infrastructure component

Today, Elkjøp’s platform hosts over two hundred microservices, all tuned for the increased requirements of a ²/₂ modernized and seamless purchasing experience for Elkjøp’s customers, whatever channel they’re on—online, mobile, or in-store.

What’s next for Elkjøp? After the successful migration in Denmark, the retailer will roll out the new POS system across its 400 stores in Norway, Sweden, and Finland in a little over six months, right in time for the next Black Friday busy season.

“That’s a really aggressive rollout. But since we already validated that this works well, we are confident we can move fast,” said Hagnås. By the time the project is complete, every sale – 40 billion NOK (\$4.7 billion) per year – will be processed through Linkerd and the new Kubernetes environment.

“We are trusting Linkerd to help us keep the company running and help our customers enjoy amazing technology.”