

Success Story

Mixed-OS Kubernetes Platform

Boris Djurdjevic

17.07.2021
microfast GmbH

A Swiss software development company for resource management systems asked us to support them in building a Kubernetes platform which can help to accelerate the development of their new product and host their older primary product as SaaS. The latter is also a .NET software that is containerizable, but can't be run on Linux while the new product is made to run on Linux. This is the reason why a mixed-OS Kubernetes cluster was required. Full integration in their development procedure and a modern monitoring setup were also part of the project.

IaaS and PaaS

Their infrastructure is running on Hetzner Cloud, which was provisioned with HashiCorp Terraform and automated using Ansible Playbooks to bootstrap Kubernetes clusters on it. This setup allows to rebuild the whole environment in less than 20 minutes, from no-VMs to a running multinode Kubernetes cluster.

Together with the customer we chose to use vanilla Kubernetes, running on Debian nodes. As their primary and older product is not runnable under Linux, we also had to use Windows kubelets. These are considered as Ready-To-Use/GA in the Kubernetes project, but contained some pitfalls, especially in the networking and service management area, which we were able to resolve. Windows kubelet support is still evolving and we could already see some improvements with newer Windows Server versions and more adapted Kubernetes components.

For the user experience, the authentication to Kubernetes itself was done using OpenID Connect, connected to Azure AD. This allows the customer to easily manage permissions and HR processes. Using authentication proxies, we were able to secure additional applications with Single-Sign-On that usually require either application specific passwords or have no authentication at all.

CI/CD and monitoring

To automate the deployment process, ArgoCD was configured to speed up their time-to-market and to provide a seamless integration from their existing Gitlab setup. It allows them to have different development versions deployed at the same time and to implement a git-push-to-productions, possibly with a 4-eyes principle. The ArgoCD WebUI is also used by their operations team to have a simple overview what is running.

To monitor the cluster and application lifecycle, we leveraged on the `kube-prometheus-stack`, providing metric based observability and alerting. Alerts are also configured to be pushed over a bridge to their Microsoft Teams channel. This allows their operations to be notified about problems, in the software they already use. It eases the coordination and minimizes the overhead of having to observe different dashboards or channels.

For better insights in the applications, the ELK stack together with fluentbit was configured to gather logs and Jaeger Tracing was integrated in the application for tracing. This allows good observability where analytics can be made on and simplifies debugging in case of errors. Such insights are critical for running microservices.

The SSO integration covers also all components of the CI/CD and monitoring pipeline, removing the need for application specific passwords.

Key points

The business outcome of the new environment is:

- Faster time-to-market
- Easier scaling for number or services they offer
- Easier scaling of computing resources for the services
- Less errors through End-to-End automation
- Single-Sign-On of the infrastructure components increases the user experience and security