Krzysztof Dariusz **Nazarewski**

Kubernetes/Cloud Engineer · DevOps/Automation/DevEx Engineer · Python/Go/IaC Developer

Warsaw, mazowieckie, Poland

☎ (+48) 530-019-621 | ✉ cv@kdn.im | nazarewk | nazarewk | nazarewk

Summary

Striving to build highly automated, scalable and cloud-agnostic platform covering the whole software development lifecycle: starting with ergonomic local development through efficient test environments and CI/CD leading to secure, observable and highly available production.

Skills

Programming	Primary: Python, Go, shell/bash, Nix; Basics: JavaScript, Lua, Java, CUE; Learning: Rust, Nickel
CI/CD	Argo CD/Workflows, Gitlab, CircleCI, Bitbucket, Luigi
Security	Vault, Netbird/Wireguard, authenticated/private infrastucture
IaC	Terraform (incl. writing providers), NixOS, GitOps
Containers	docker, podman, buildkit, skopeo, buildah
Kubernetes	Rancher, k3s, EKS, AWS integrations, CNIs, autoscaling
Observability	Elasticsearch/Opensearch, OpenTelemetry, Fluentd, Prometheus, Datadog
AWS	SDK (Python, Go), Route53, VPC, EC2, SSM, ASG, IAM, S3, Lambda, EKS and other EKS-assisting services
Backend	Python (Django REST Framework, asyncio/trio, Celery), PostgreSQL, Redis
Languages	Polish, English, basic mandarin

Education

Warsaw University of Technology

Warsaw, Poland

B.S. in Computer Science and Software Engineering

2011 - 2015, grad.2018

- Bachelor's Thesis - *Implementing Kubernetes cluster on diskless machines*

Experience

Incredevo

Warsaw, Poland

DevOps | Cloud Engineer

Oct. 2022 - Aug. 2023

- Outsourced part-time (100h/mo) as the only cloud engineer in client's company. Was primarily responsible for implementation from scratch and upkeep of their EKS-based AWS infrastructure,
- Started by automating and forcing the build process to be reproducible for 15 Java applications without versioning packaged into 8 containers,
- Researched, evaluated, debugged and sometimes fixed various open-source tools written primarily in Go,
- Built up AWS infrastructure from zero using Terraform/Terragrunt,
- Set up infrastructure to be deployed to private VPC and accessible through Netbird VPN,
- Set up Bottlerocket based EKS cluster configured using kustomization Terraform provider with all the usual cloud controllers,
- Set up observability stack using Opensearch, OpenTelemetry and Prometheus,
- Set up NLB-based APISIX Ingress Controller with automatic certificates using cert-manager,
- Set up Argo Workflows for CI/CD,
- Set up rootless container builds in Argo using buildkit or devenv.sh (Nix),
- Set up devenv.sh based local development environment,
- Rewritten the build process from Bitbucket Pipelines to Argo Workflows,
- Set up cluster (Karpenter) and workflows (VPA) autoscalers for cost optimization,
- Rewritten the build process from Argo Workflows to spotify/luigi for a local execution and easier handover,
- Prepared a NixOS-based EC2 instance capable of running all the tooling to be used as a last-resort when Mac setup breaks,

Fresha

Warsaw, Poland

DevOps | DevEx

Nov. 2021 - Oct. 2022

- Member of a team focusing on increasing development velocity through CI/CD process and developer's experience improvements,
- Successfully cut Datadog monitoring costs by a third through refining gathered data,
- Worked primarily on complete redesign and simplification of existing - organically acquired over the years - complex processes,
- The job involved identifying, measuring and debugging performance and operational issues on very large (30-60 envs, 20+ apps each, multiple deployments per hour) EKS cluster dedicated to staging environments,
- Set up new deployment and secrets management flow using: ArgoCD, External Secrets Operator and a set of Python scripts for the GitOps repository management & optimization,

BEC Poland
Warsaw, Poland

DevOps | Cloud Engineer *July 2020 - Oct. 2021*

- Introduced 5 man team to: Terraform, pytest and Terraform SDK,
- Introduced team to and provided guidance for custom Terraform providers implementations,
- Participated in design and implementation of company-wide Azure based cloud platform,
- Worked with Azure cloud using Terraform and Python SDK,
- Tested infrastructure with pytest and pytest-bdd (Python implementation of Behaviour Driven Development/Gherkin),
- Designed and implemented shared testing library to be used by the team and customers,
- Designed and implemented developer's environments using: Packer, Vagrant, VirtualBox, Azure DevTest Lab,

GFT Poland
Warsaw, Poland

Kubernetes Administrator | DevOps | Cloud Engineer *Jan. 2019 - June 2020*

- Introduced 6-man team to Terraform and Kubernetes,
- Participated in fast-paced infrastructure development in a large team,
- Participated in development, automation and coordination of workflow across many programming teams and projects,
- Was the only person responsible for deploying, configuring and managing pre-production Kubernetes clusters,
- Provided initial deployment and guidance for Hashicorp Vault, ELK stack and Prometheus,
- Deployment and management of Kubernetes cluster backed by Rancher (Custom Nodes) on AWS EC2,
- Implemented AWS Lambda function overseeing Kubernetes cluster nodes with assistance of AutoScaling Groups,
- Implemented infrastructure development workflow including mass-execution, validation and automation for Terraform,
- Implemented in-cluster network monitoring solution using Calico (iptables) logs and Prometheus,
- Implemented Kubernetes cluster extensions and validation using Webhooks and Custom Resources,
- Used/integrated with: Gatekeeper, Istio, Okta, Datadog, Artifactory, CircleCI, GitHub, Spinnaker,

Daftcode
Warsaw, Poland

department CTO | Software Architect | Python Developer | Linux Administrator | DevOps *Jun. 2016 - Dec. 2018*

- Started as the most experienced backend developer on the team with some sysadmin skills, then gradually moved to full-time infrastructure administration and devops role,
- Developed application's prototype (Python 2, Django) into a dynamically growing project,
- Designed from ground up then migrated existing codebase to the new implementation based on Python 3, Django REST Framework and Celery,
- Migrated manually managed on-premise environment to semi-automated (Ansible and Supervisord) deployment onto AWS EC2-based infrastructure split into production and development envrionment,
- Unified error reporting with Sentry and streamlined deployment with BuildBot + Ansible,
- Implemented SSH based infrastructure access management with quick access revocation and automated setup of developers's machines,
- Configured data analysis and monitoring with ELK stack,
- Achieved zero-interruption deployments with uWSGI zerg mode,
- Configured live performance monitoring and alerting with ELK stack and Prometheus (mostly providing data to Kiosk-style dashboard running 24/7 in the office),
- Created terraform-provider-scripted,
- Implemented single-command Kubernetes cluster deployment and management using Terraform, kops and Python scripts,
- Implemented Kubernetes cluster hardening: private network, NetworkPolicy, PodSecurityPolicy, kube2iam, in-cluster SSL authentication, etc.,
- Deployed on Kubernetes: Docker Registry, Helm, S3-backed Vault and Gitlab, cert-manager, Prometheus stack, metrics-server, efs-provider, etc.
- Implemented application infrastructure deployment on top of Kubernetes based on Gitlab CI and Helm.

10Clouds
Warsaw, Poland

Django Developer *Aug. 2015 - May 2016*

- Tested then migrated an existing Django REST Framework & PostgreSQL tenants based SaaS from Python 2 to Python 3,
- Developed and maintained above mentioned backend and occasionaly frontend: Angular 1.3, CoffeeScript, ES6, SASS
- Designed and implemented Django REST Framework based backend in another project,

JMR Technologies
Warsaw, Poland

Lead Python Developer *Nov. 2014 - Jun. 2015*

- Was primary Python developer in a company newly adopting Python for selected projects,
- Designed and implemented of Django based CRM,
- Gave occasional assistance to Odoo team.

Freelance project
Poland

Full-stack Software Developer *Apr. 2014 - Jun. 2015*

- Implemented and deployed Django REST Framework backed CMS/knowledge base,
- Implemented JavaScript (Dojo Toolkit 1) based desktop application (TideSDK, NW.js, then Electron) backed by aforementioned backend.

Open Source contributions

- Go: prometheus/prom2json: implement reading metrics from stdin or file (only URLs were supported),
- Go: netbirdio/netbird VPN: fixed and tested errors with network routing,
- Go+k8s: kubectl drain improvements helping with cleanly scaling down cluster to zero,
- Go+k8s: gitops-engine/Argo CD add proper handling when etcd hiccups (eg: GC) occur,
- Go+k8s: helm/chart-testing: implementing passing of additional flags to helm command,
- Go+k8s: rook: enabled support for running on NixOS by adding NixOS-specific PATH,
- Go+tf+k8s: terraform-provider-kustomization added handling of remaining kustomize flags,
- Go+TF: terraform-provider-vault: improved handling of groups and policies assignments,
- Go+TF: created daftcode/terraform-provider-scripted as a first touch with Terraform and Go,
- Go+TF: created nazarewk-iac/terraform-provider-custom refined and properly designed implementation of above,
- Python+AWS: Nike-Inc/gimme-aws-creds: large refactor to enable using the tool in CI and as a library,
- Nix: various contributions to and packaging fixes for NixOS/nixpkgs,
- Nix+containers: devenv: enable building containers in CI using Nix's Flakes,