DAVID LIHOR

AWS Certified Solutions Architect – Associate

credly.com/badges/b8e38d5f-607f-4328-a3f5-f55254b81cab

linkedin.com/in/david-lihor | **0760-816-277**

github.com/davidlihor | **davidlihor.dev@gmail.com** | **Cluj-Napoca**

Cloud & Platform Engineer with AWS Solutions Architect – Associate certification. Hands-on experience deploying EKS infrastructure with Terraform IaC, Istio mTLS enforcement, ArgoCD GitOps, and multi-stage DevSecOps pipelines. Built 4 production-ready cloud projects demonstrating serverless orchestration, service mesh architectures, and secrets management. Eager to contribute to DevOps teams building scalable, secure cloud-native systems.

Technical Skills

Cloud & Infrastructure: AWS (EKS, Lambda, API Gateway, DynamoDB, S3, CloudFront, RDS, ElastiCache, VPC, KMS, IAM, EventBridge, SQS, Step Functions, Secrets Manager, Route53, ACM), Terraform, HashiCorp Vault

Container Orchestration & Service Mesh: Kubernetes, EKS, Helm, kubectl, Istio (mTLS, Egress Gateway, AuthorizationPolicy), External Secrets Operator, cert-manager

CI/CD & Automation: GitLab CI, GitHub Actions, ArgoCD, Argo Rollouts, GitOps workflows, Trivy, Checkov, Gitleaks, Semgrep

Languages & Tooling: Python, Boto3, C#, .NET, Git, YAML, HCL

Observability: LGTM Stack (Loki, Grafana, Tempo, Mimir), Kiali, CloudWatch

Projects

Microservices Platform with Service Mesh & GitOps | github.com/davidlihor/Eshop-Distributed-CloudNative-Architecture

Microservices architecture deployed on EKS, utilizing Istio for traffic management and ArgoCD for GitOps with Istio

- Provisioned EKS cluster with Terraform deploying 6 microservices (Catalog, Basket, Discount gRPC, Ordering, YARP Gateway, Web Client) through ArgoCD ApplicationSets with Helm templating and drift self-healing
- Configured Istio service mesh with mTLS enforcement, REGISTRY_ONLY egress mode using ServiceEntry allow-lists (RDS/ElastiCache/AmazonMQ), and L7 AuthorizationPolicy for HTTP method control per security zone
- Integrated Argo Rollouts for progressive delivery strategies and External Secrets Operator synchronizing AWS Secrets Manager credentials to Kubernetes runtime environments

Serverless Task Platform with Event-Driven Orchestration | github.com/davidlihor/Secure-Serverless-Application

Multi-service AWS architecture with Lambda, Step Functions, and EventBridge Pipes for parallel resource cleanup

- Architected serverless application with API Gateway REST API, 7 Lambda functions in VPC private subnets, DynamoDB storage, and Cognito authentication deployed via Terraform modules (network, security, storage, compute, frontend)
- Implemented Step Functions parallel workflows triggered by EventBridge Pipes from SQS queues — simultaneous DynamoDB deletion and S3 object cleanup with independent error handling and dead-letter routing
- Deployed CloudFront distribution with WAF rate limiting, signed URLs for authenticated S3 access using KMS-protected private keys, and VPC endpoints eliminating NAT Gateway costs for DynamoDB/S3/Secrets Manager traffic

EKS Infrastructure with Multi-Stage DevSecOps Pipeline | github.com/davidlihor/Eshop-Distributed-IaC

Terraform-based AWS foundation with GitLab CI security gates and OIDC authentication

- Provisioned multi-AZ EKS cluster (1.34) with Terraform deploying RDS PostgreSQL 17, ElastiCache Valkey 8, Amazon MQ RabbitMQ, and private VPC architecture with Gateway/Interface endpoints for cost-optimized AWS service access
- Built GitLab CI pipeline with OIDC Web Identity Federation (keyless authentication) executing 6-stage workflow: secret detection (Gitleaks), IaC scanning (Trivy + Checkov), Terraform plan/apply with manual approval gates, and configuration layer on private runner
- Deployed LGTM observability stack (Loki, Grafana, Tempo, Mimir) with S3 backend storage and IAM Pod Identity associations for EKS workloads accessing logs/metrics/traces

Vault Secrets Management on EKS with IRSA | github.com/davidlihor/Terraform-Vault-EKS-Bootstrap

Centralized secrets platform integrating HashiCorp Vault with Kubernetes authentication and KMS encryption

- Deployed Vault in HA mode on EKS with cert-manager issuing TLS certificates, Raft consensus backend, and KMS auto-unseal via IRSA eliminating static credentials
- Configured Vault Kubernetes auth backend with service account bindings, KV-v2 secrets engine, and policy-based access control for application workloads retrieving dynamic credentials
- Integrated External Secrets Operator with Vault provider synchronizing secrets to Kubernetes namespaces and AWS Load Balancer Controller managing ingress with ALB provisioning

Education

High School Student - Grade 11 | Expected to finish: 2027 Cluj-Napoca, Romania

Self-taught in cloud architecture, container orchestration, and infrastructure automation through AWS documentation, HashiCorp tutorials, and hands-on project work.