DevSecOps Toolkit Interview Outline

Objective

Build and demonstrate a secure-by-default DevOps pipeline that integrates security at every stage: from infrastructure to deployment, secrets, and monitoring.

Infrastructure as Code (IaC)

- Tool: Terraform
- Security: Scanned with tfsec, Checkov
- Policy Enforcement: OPA (Rego rules for IAM, S3, secrets)
- Best Practices: Modular code, versioned state, secure defaults

CI/CD Pipeline

- Tool: GitHub Actions
- Stages:
  - Lint Test Security Scan Policy Check Plan Apply
  - Manual approval required for production deploys
- Security Tools: Trivy, tfsec, Conftest

Secrets Management

- Tools:
  - HashiCorp Vault (HCL policy)
  - AWS Secrets Manager (Terraform-managed)
  - Azure Key Vault (ARM + role assignment)
- Approach: Environment injection in pipelines; no hardcoding

Container Security

- Base Image: Distroless static non-root
- Tools: Trivy, Grype
- Practices: Minimal surface area, signed images, version pinning

Azure Monitoring

- Tools: Azure Monitor, Log Analytics
- Assets:
  - Workbooks (dashboards)
  - KQL queries for CPU, restarts, login failures
  - ARM-deployed alert rules with RBAC

Documentation

- architecture.md system overview
- contributing.md contribution guidelines
- glossary.md DevSecOps terms
- README.md structured onboarding for every folder

Interview Tips

- Emphasize the full lifecycle security coverage
- Mention specific tools you've integrated hands-on
- Talk about CI/CD approvals, scanning, and policy blocking
- Bonus: Show a working repo or diagram if asked