Security Report – DevSecOps Assignment

Security Controls Added

- **Application & Code Security**
    - Secure Dockerfiles (multi-stage, non-root user, dropped capabilities).
    - SAST with **Semgrep**.
    - SCA with **Trivy fs** (dependency scanning).
    - Image scanning with **Trivy image**.
- **Infrastructure as Code**
    - Terraform provisioning of S3 bucket with encryption enabled.
    - IaC scanning with **Checkov**.
- **CI/CD Pipeline Security**
    - Jenkins pipeline stages for scanning, testing, building, pushing.
    - Secrets stored in Jenkins credentials (GHCR creds).
    - Pipeline gates → fail build if CRITICAL findings > 25.
- **Deployment & Runtime Security**
    - Kubernetes manifests with securityContext enforcing non-root execution.
    - Kyverno admission policies: enforce CPU/memory defaults, prevent privileged escalation.
    - Strict hardening of Dockerfiles of backend and frontend.

Gaps & Risks

- Jenkins is assumed to run on an AWS EC2 instance with IAM roles attached → **not portable**.
- Assumes AWS CLI and `kubectl` pre-installed and configured on Jenkins node.
- GHCR credentials are pre-synced in EKS cluster (manual step).
- No dedicated Kubernetes namespaces per environment (frontend/backend/test) or per stack (be/fe/db).
- No default LimitRanges or ResourceQuotas (per-namespace resource governance missing).
- Static manifests → not parameterized (e.g., via Helm/Kustomize) for easier updates.
- Health probes (`readinessProbe`, `livenessProbe`) missing in deployments.
- Monitoring & alerting (Prometheus, Grafana, Falco) not integrated.
- No image signing .

Next Steps for Improving Security Posture

1. **Secrets Management & Rotation**
   - Automate AWS secret rotation every 30 days via Lambda functions.
2. **Observability & Alerts**
   - Add email/Slack notifications on Jenkins stage failures.
   - Integrate monitoring with Prometheus/Grafana.
   - Add Falco for runtime anomaly detection.
3. **Kubernetes Hardening**
   - Introduce namespaces for frontend, backend and environments - dev/test/prod.
   - Apply **LimitRanges** and **ResourceQuotas**.
   - Add readiness/liveness probes to deployments.
4. **Pipeline Improvements**
   - Sign container images (Cosign/Sigstore).
   - Generate SBOM (Trivy, Syft) and store in artifact registry.
   - Enable policy-as-code (OPA Gatekeeper/Kyverno advanced policies).
5. **Compliance & Governance**
   - Regular IaC scanning in pipeline (Checkov in Jenkins).
   - Move towards Zero Trust (fine-grained IAM roles, RBAC).