To implement a **High Availability (HA), zero-downtime Nexus IQ Server deployment** on **AWS EKS**, using **RDS Aurora PostgreSQL** with **Terraform**, **Helm**, and **automated RDS rollback**, here’s a complete, production-grade design including infrastructure provisioning, deployment, and disaster recovery with **RDS rollback automation**.

**🚀 Overview**

| **Component** | **Tech Stack / Strategy** |
| --- | --- |
| **App Platform** | AWS EKS (Multi-AZ) |
| **DB Backend** | Amazon Aurora PostgreSQL (Multi-AZ, provisioned) |
| **Deployment Tool** | Terraform + Helm |
| **Availability** | Blue-Green Deployment (via Helm + Ingress) |
| **RDS Rollback** | Automated restore from snapshots or point-in-time |
| **Secrets Mgmt** | AWS Secrets Manager + IRSA |

**🔁 Zero-Downtime HA with Blue-Green**

1. Deploy current version (nexus-iq-blue)
2. Deploy new version (nexus-iq-green) → health check
3. Cut traffic to green by updating Ingress
4. If issues → revert Ingress to blue

This allows rollback of app versions with **zero downtime**.

**🛡️ RDS Aurora: Automated Rollback**

**🧠 Concepts:**

* **Automated Snapshots** (daily backups with retention)
* **Manual Snapshots** (on demand, before upgrades)
* **Point-in-Time Recovery (PITR)** – restore to a timestamp (e.g., before a deployment)

**✅ Terraform Setup (with PITR Support)**

h

CopyEdit

module "aurora" {

source = "terraform-aws-modules/rds-aurora/aws"

name = "nexus-iq-db"

engine = "aurora-postgresql"

instances = 2

instance\_class = "db.r6g.large"

master\_username = "iqadmin"

master\_password = var.db\_password

backup\_retention\_period = 7

preferred\_backup\_window = "03:00-04:00"

deletion\_protection = true

storage\_encrypted = true

apply\_immediately = true

}

**🔄 RDS Rollback via Terraform (PITR)**

**Step 1: Determine last known good time (timestamp or snapshot ID)**

Example: "2025-05-21T14:00:00Z"

**Step 2: Restore cluster from snapshot or PITR**

hcl

CopyEdit

resource "aws\_rds\_cluster" "nexus\_rollback" {

cluster\_identifier = "nexus-iq-db-rollback"

engine = "aurora-postgresql"

restore\_to\_point\_in\_time {

source\_cluster\_identifier = "nexus-iq-db"

restore\_time = "2025-05-21T14:00:00Z"

use\_latest\_restorable\_time = false

}

}

**Step 3: Update Nexus IQ Helm deployment to point to new DB endpoint**

Update values.yaml:

yaml

CopyEdit

externalDatabase:

host: nexus-iq-db-rollback.cluster-xxxxx.us-east-1.rds.amazonaws.com

Or do it dynamically via:

bash

CopyEdit

helm upgrade nexus-iq-blue \

--set externalDatabase.host=nexus-iq-db-rollback.cluster-xxxxx.us-east-1.rds.amazonaws.com

**✅ Optional: CI/CD Workflow for RDS Snapshot + Restore**

**GitHub Actions / ArgoCD Hooks:**

1. **Pre-deployment**: Create snapshot
2. **Post-deployment test**: Health check
3. **On failure**: Restore snapshot, patch Helm release

Example:

bash

CopyEdit

aws rds create-db-cluster-snapshot \

--db-cluster-identifier nexus-iq-db \

--db-cluster-snapshot-identifier "nexus-iq-pre-deploy-$(date +%s)"

On failure:

bash

CopyEdit

aws rds restore-db-cluster-from-snapshot \

--db-cluster-identifier nexus-iq-db-rollback \

--snapshot-identifier nexus-iq-pre-deploy-<timestamp>

**🔐 Secrets Management**

Use [AWS Secrets Manager](https://docs.aws.amazon.com/secretsmanager/latest/userguide/intro.html) to store database credentials, pulled into Helm via External Secrets or IRSA:

yaml

CopyEdit

externalDatabase:

user: iqadmin

password: {{ .Values.secret.dbPassword }}

**🧪 Health Check & Rollback Trigger**

You can automate with:

bash

CopyEdit

curl -fs https://iq.example.com/healthcheck || {

echo "Nexus IQ unhealthy, triggering rollback..."

aws rds restore-db-cluster-from-snapshot ...

helm upgrade nexus-iq-blue --set externalDatabase.host=...

}

**🧭 Workflow Summary**

**Normal Deployment**

1. Create DB snapshot
2. Deploy green via Helm
3. Health check
4. Switch Ingress to green

**Rollback Scenario**

1. Green fails
2. Restore RDS from snapshot (PITR or manual)
3. Patch Helm to point to restored DB
4. Switch Ingress back to blue

**📌 Summary**

| **Capability** | **Status** |
| --- | --- |
| HA Nexus IQ on EKS | ✅ |
| Zero-downtime Deployments | ✅ |
| Blue-Green Switching | ✅ |
| RDS PITR + Snapshots | ✅ |
| Terraform Automation | ✅ |
| Helm Config Management | ✅ |
| CI/CD Rollback Trigger | ✅ (Scripted) |
| Secrets Best Practice | ✅ AWS Secrets Manager |