AWS ElastiCache for Redis — Hands-On Lab Manual

1. Introduction

Amazon ElastiCache for Redis is a fully managed, in-memory key-value data store and caching service by AWS. It improves application performance by retrieving data from high throughput, low latency, in-memory caches instead of relying entirely on slower disk-based databases.

2. Architecture Overview

ElastiCache for Redis supports three deployment types: - Single-node: Basic caching for dev/testing - Multi-AZ Replication Group: HA with automatic failover - Cluster Mode: Sharded architecture for scalability

3. Prerequisites

• AWS account with ElastiCache permissions • Basic knowledge of VPC, Subnets, EC2 • AWS CLI & Redis CLI installed

4. VPC Configuration

Step 1: Create a custom VPC (10.0.0.0/16) Step 2: Create two private subnets (10.0.1.0/24 & 10.0.2.0/24) Step 3: Create security groups for Redis and EC2, allowing port 6379 for Redis and port 22 for SSH.

5. Create Subnet Groups & Parameter Groups

Subnet groups ensure Redis nodes span multiple AZs. Parameter groups allow tuning Redis settings like maxmemory-policy, timeout, etc.

6. Create an ElastiCache Replication Group

• Go to AWS Console → ElastiCache → Redis → Create. • Engine: Redis 7.x • Deployment: Multi-AZ • Configure replication, enable encryption, set AUTH token.

7. Enable Cluster Mode

• Choose 'Cluster Mode Enabled' during creation. • Configure number of shards and replicas per shard. • Data is auto-sharded for high performance.

8. Connecting from EC2

Launch an EC2 instance in the same VPC and install Redis CLI: sudo yum install redis -y Connect to Redis: redis-cli -h -p 6379 -a

9. Failover Testing

Reboot the primary node from the AWS Console and verify automatic failover: redis-cli -h INFO replication

10. Performance Benchmarking

Run redis-benchmark from EC2: redis-benchmark -h -p 6379 -n 100000 -c 100 -a This tests throughput and latency.

11. Monitoring & Troubleshooting

• Use CloudWatch for CPU, memory, cache hit ratio. • Redis CLI commands: redis-cli INFO memory redis-cli SLOWLOG GET redis-cli MONITOR

12. Security Best Practices

• Place Redis in private subnets • Enable AUTH & KMS encryption • Restrict access via SGs • Enable TLS for in-transit encryption

13. Real-Time Use Case

E-commerce stock availability system: • Redis stores live inventory data • Checkout updates Redis instantly • Asynchronous sync to RDS/DynamoDB

14. Interview Questions

• Difference between ElastiCache & MemoryDB • Redis Cluster Mode Enabled vs Disabled • How does AWS handle failover? • How do you secure Redis in AWS? • Redis vs DynamoDB Accelerator (DAX)?

End of Lab Manual