# Infrastructure

## AWS Zones

us-east-2a","us-east-2b","us-east-2c

"us-west-1a", "us-west-1c"

## Servers and Clusters

Three Ubuntu Web Servers in each availability region

Single 2-node cluster in each availability region

Single 2 node RDS cluster in each availability region

### Table 1.1 Summary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Asset | Purpose | Size | Qty | DR |
| Asset name | Brief description | AWS size eg. t3.micro | Number of nodes/replicas or just how many of a particular asset | Identify if this asset is deployed to DR, replicated, created in multiple locations, or just stored elsewhere |
| App server/instance | Runs the application | t3.micro | 3 instances per region | Deployed to West 1 |
| EKS cluster | Deployed and run application | t3.medium | 2 node clusters | clusters are created in 2 regions |
| Virtual IP | Direct traffic to AZ’s | NA | VIP in each availability zone | VIPs for clusters in each region |
| Load balancer | Directs traffic in each region | NA | 1 in each region | Manual failover with duplicate configuration |
| SQL Cluster | Regional cluster - Allows for replication in another region (west1) | na | Cluster with a writer and reader from each region. | Allows for replication and restoration with 5-day backup/retention |
| SQL instances | Writer and reader for data resiliency. | db.t2.small (2gb) | 2 | A writer and reader instance in West 1 for replication and restoration as needed. |
|  |  |  |  |  |

### Descriptions

The application infrastructure consists of a two-node web server cluster with nodes in 2 regions and virtual machine instances in multiple availability zones.

Virtual IP address exist in each availability zone and a load balancer is configured in each region to route traffic.

SQL is deployed using a two-node cluster in two availability zones. A HA primary read/write copy runs in one region with replicated data going to a read only cluster in the other region with a 5 day back/restore availability.

## DR Plan

### Pre-Steps:

List steps you would perform to setup the infrastructure in the other region. It doesn't have to be super detailed, but high-level should suffice.

1. PreConfigure the web server/cluster in Terraform in zone 1.   
Duplicate the configuration into a zone2 and update the appropriate availability zones.

2. PreConfigure the SQL server primary and secondary instances in terraform zone1 directory.

3. Run terraform apply from each zone folder to build the infrastructure in both regions

## Steps:

You won't actually perform these steps but write out what you would do to "fail-over" your application and database cluster to the other region. Think about all the pieces that were setup and how you would use those in the other region

1) Using Route53 direct web server traffic from the public DNS to the load balancer in the alternate region (west 1)

2) Run the code and bring up the application in the EKS cluster in West 1.

3) Point the application to the RDS cluster/ writer instance in secondary region (west1).