“Design for Failure”
High Availability Architectures using AWS

Harish Ganesan
Co founder & CTO
8K Miles

www.twitter.com/harish11g
http://www.linkedin.com/in/harishganesan
Sample Use Case

- Multi-tiered LAMP/LAMJ Web site on AWS
- Data base tier using MySQL
- Data Exchange (DEX) layer containing HTTP pull, Queues, Storage, BG programs and Alerts
A simple LAM(X) Architecture

1. Web/App Server interacts with MySQL for Queries and Transactions

2. Data Exchange Web Services are pushed with information through HTTP

3. Entire Setup is Monitored and Secured using AWS
A simple LAM(X) Architecture

AWS Security Groups

US-EAST-1a

Web/App Server

Data Exchange Server

MySQL DB

CloudWatch

Single Point of Failure at many layers

Not a Highly Available Architecture
High Availability in Web / App / DEX layer
1. Add Multiple EC2 instances in Web/App layer
2. Add Multiple EC2 instances in DEX Layer
1. Add Elastic Load Balancer

2. Round Robin with Session Sticky policy

3. ELB is a Highly Available Service with No SPOF
• ELB vs HAProxy
• AWS ELB provides load balancing service with thousands of EC2 servers behind them
• AWS ELB will automatically Scale up /down the load balancing servers in backend
• The theoretical maximum response rate of AWS ELB is limitless
• It can handle 20000+ concurrent requests easily
High Availability @ Web/App/DEX layer

1. Add AWS Auto Scaling to Web and DEX layer

2. Tie AWS Auto Scaling with AWS ELB
AWS Auto Scaling will manage un Healthy EC2 instances

AWS Auto Scaling will ensure minimum number Web/App /DEX EC2 instances are always running

In event of failure, new instances will be launched between 30-120 seconds automatically

ELB traffic is seamlessly attached to the Auto Scaled EC2 instances
What happens when an AWS AZ in US-East itself fails?
High Availability @ Web/App/DEX layer

AWS Elastic Load balancer

US-EAST-1a

AWS Security Groups

Web/App Server

Data Exchange Server

Auto Scaling

MySQL DB

CloudWatch

US-EAST-1b

AWS Security Groups

Web/App Server

Data Exchange Server

Auto Scaling

MySQL DB

CloudWatch
• AZ’s are connected by Low Latency network
• AZ’s are insulated from failures in other Availability Zones *
• AWS Auto Scaling can manage EC2 instances across AZ’s
• AWS ELB can direct load to EC2 instances across AZ’s
• AWS CloudWatch can monitor the EC2 instance availability across AZ
High Availability in DB layer
High Availability @ DB layer

1. Add Read Replica’s to the Master DB

2. Add RDS Stand by
1. Read Replica’s launched in Multiple AZ’s for HA

2. RDS Standby will be launched on different AZ from the RDS master for HA

3. Web/App hosted on Amazon EC2 will transact with RDS master and read from Read replica’s
High Availability @ DB Layer

- RDS Master and RDS Standby in Multiple AZ for HA
- Read Replica’s in Multiple AZ for HA
- Offers No SPOF on AZ level
- Read Replica’s can be launched/terminated without affecting the RDS Master availability
- In event of RDS master failure, RDS Standby will be automatically promoted
- Promotion <180 seconds and no changes in the application
• DB snapshots and MySQL Dumps facility available
• Automatic full backups at configured maintenance windows
• Point in time recovery till last minute
• Recovery might require App layer configuration changes
High Availability in DEX layer
High Availability @ DEX Layer

• DEX layer is designed with AWS building blocks like
  • S3, SQS, SES, EMR, CloudWatch

• AWS blocks are in built with fault tolerance and HA
High Availability @ DEX Layer

1. DEX layer puts the request messages in SQS and XML in S3
2. Auto Scaled BG programs process the same from the respective systems
3. SQS is a Highly available messaging system with in built fault tolerance
4. S3 is a Highly available storage system
1. BG programs send external emails using SES

2. SES is a Highly available email service

3. DEX layer input endpoint URLs are configured with AWS Route 53
High Availability @ DEX Layer

1. Elastic MapReduce jobs of DEX layer process the files present in S3
What happens if the Entire USA East region is affected?

Solution: Design High Availability across Regions
High Availability across AWS Regions

Web site is hosted in AWS USA East

Web Site is hosted in AWS USA West
HA across AWS Regions

Main Website in AWS region 1

AWS USA East Region

AWS West/Europe/APAC Region

Main Website in AWS region 2
Website in Multiple AWS regions (using MySQL)

1. Dynamic/Managed/Directional DNS Servers
   Directional DNS Servers directs the user requests to Main site in AWS USA east region. In case of outage in USA East region, the web requests are directed to same website hosted in USA West region.

Main Site - AWS USA East Region

AWS Elastic Load Balancer

2. AWS ELB balances the requests between the Auto scaled EC2 launched in multiple AZ’s inside the EAST region.

   USA-East-1A
   USA-East-1C

Auto Scaling

MySQL Master

MySQL is launched in Multiple AZ’s inside the EAST region with M-M replication mode.

Main Site- AWS USA West Region

AWS Elastic Load Balancer

3. USA-WEST-1A
   USA-WEST-1B

Auto Scaling

MySQL Master

MySQL Master replication between USA EAST and WEST regions are setup.
Website in Multiple AWS regions (using RDS)

Directional DNS Servers directs the user requests to Main site in AWS USA east region. In case of outage in USA East region, the web requests are directed to same website hosted in USA West region.

Dynamic/Managed/Directional DNS Servers

Main Site - AWS USA East Region

AWS Elastic Load Balancer

1. USA-East-1A
   - AWS ELB balances the requests between the Auto scaled EC2 launched in multiple AZ’s inside the EAST region

2. USA-East-1C
   - Auto Scaling
   - RDS Master
   - MySQL is launched in Multiple AZ’s inside the EAST region with HA replication mode

Main Site- AWS USA West Region

AWS Elastic Load Balancer

3. USA-West-1A
   - Auto Scaling
   - RDS Master standby

4. USA-West-1B
   - Auto Scaling
   - RDS Master
   - RDS Master replication between USA EAST and WEST regions is done programmatically

Programmatic Replication
HA across AWS regions

• Leverages AWS Inter Region application hosting
• Website is hosted on multiple Regions on AWS (example USA east – west, USA –EUR etc)
• GEO traffic distribution and HA across continents is possible in this Architecture blueprint
• Directional DNS combined with Route 53
• Suitable for companies which demand high level of Scalability, load balancing and Availability across the globe
• Managed DNS server will provide automatic failover at DNS level in case of a outage at the primary website location
• Transparent switch between websites hosted in AWS East and AWS West/Europe within <60 seconds during outage
• Automatic Traffic diversion to nearest site location
• Managed/Directional DNS servers are globally distributed and Highly Available Service
• Both AWS regions have RDS Master/Stand by Setup
• Programmatic replication of data between RDS Masters/Read in different AWS regions is needed
• HA **inside** a Region and **across** the Regions
• Master - Master MySQL is configured in two different AZ’s (offering **HA inside Region**)
• Configure Asynchronous Read slaves in Multiple AZ’s
• Master – Master replication is configured between MySQL of different regions (**HA across regions**)
• Elastic IP & health check based elevation within 60 seconds during failure
• Asynchronous data replication
• Scalable and Highly available Architecture
• Inter Regional High Availability in AWS
• In event of failure at USA east region, the traffic can be directed to USA west/Europe in seconds
• Website deployed in both regions can scale and shrink according to load
• Cost effective for large server farm deployments
• Low latency achieved through traffic direction
• No customers are lost because of load or availability problems. Ops are happy !!!
Negatives

- Complete Dependency on AWS cloud
- Technically complex and intricate setup
- Costlier to build and operate (Sophistication comes at a cost)
- No Unified Infra Management currently for this architecture
  - Example: Directional DNS and AWS are two separate management consoles
• Understood some AWS Building blocks for HA and fault tolerance
• Applied AWS HA techniques for sample use case
• How to achieve High Availability across AWS Availability Zones (AZ’s)?
• How to achieve High Availability across AWS regions?
How do I leverage High Availability architecture on AWS?
Leave it to the experts, we will handle this.

“Let's get the job done”
“All you need is an idea and the cloud will execute it for you.” *(Structure 2010 event)*
- Dr Werner Vogels, CTO of Amazon on 8KMiles

Contact:
http://cloud.8Kmiles.com
cloud@8Kmiles.com
harish@8Kmiles.com