Mitchell Trow

NET702 Project Report

Introduction/Problem Statement

The company needs a reliable solution to allow scalability and reliability (low downtime) for their services.

They are hosting their services locally on their own hardware, which is not currently sufficient for their increasing needs.

This is leading to increased costs and some downtime which is not ideal for their services.

To reduce the downtime and increase reliability, I propose that the company move all their services to a cloud-based solution on the AWS platform. Using AWS platform services will lead to reduced cost and higher reliability (less downtime) for the company's services.

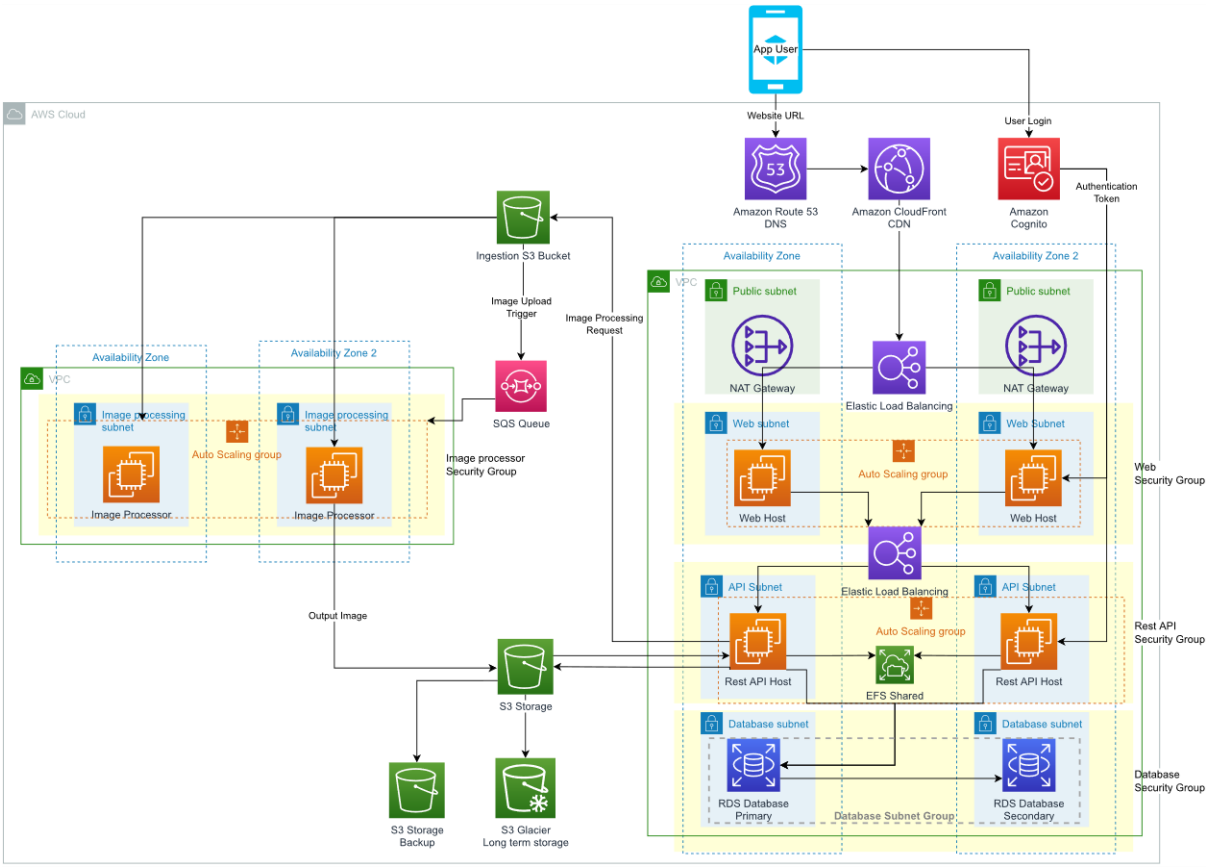
Company Requirements

- Web-based photo management application
- iOS and Android mobile app
- REST API for data access (website and app)
- User authentication (website and app)
- Photo upload processing and rendering services
- SAN Raid storage
- AWS CloudFront CDN services
- Tape backup

Cloud Solution Requirements

- Scalable, highly available and fault-tolerant architecture that conforms to AWS best practice.
- Security, monitoring that conforms to AWS best practice.
- User, group and role policy access permissions that conform to AWS best practice.
- Disaster recovery planning to meet the organisation's Recovery Time Objective (RTO) and Recovery Point Objective (RPO).
- Monthly usage cost estimation.

Proposed Cloud Infrastructure Diagram



Proposed Design/Services Explanation

- **Multiple AZs**

  To increase reliability, I propose that the vertical services be split in two or more so that each slice is in its own availability zone. Doing this will increase reliability such that if one zone becomes unavailable for whatever reason, then the other zone can continue operating normally.

- **Auto scaling groups**

  To allow the services to scale depending on the workload, I propose that each compute instance in the vertical stack be in an auto scaling group which will allow it to scale horizontally. This will reduce the cost because you are only paying for the current workload since unactive instances are automatically terminated by the scaling group.

- **Elastic Load Balancing**

  To facilitate and ensure the scaling groups are optimized, you need to spread their workload across how many instances there is. To do this you use a Load Balancer which will split the traffic coming from the internet or other internal layers to cover all the instances that is available.

- **Amazon CloudFront**

  To optimize traffic and reduce load on the hosting instances, CloudFront will cache many static elements on the website and put them in many locations around the world so that

they are delivered to users quicker. This will reduce cost, since you will need less hosting instances in the case where there is higher traffic to the website.

- **Multiple VPCs**

  The main VPC is connected to the internet and subsequent layers inside it are connected to one another via networking connections. The other VPC is for image processing and is not connected to the other one via networking. It is completely isolated from incoming internet traffic and only interacts with other AWS services such as S3 buckets.

- **Ingestion S3 Bucket and SQS queue**

  When an image is placed in the ingestion bucket, it indicates that it is ready for processing. The Bucket has an event set up to add a request to an SQS queue. The queue is then accessed by the image processors in the auto scaling group. Once an image is processed, it is saved to the main S3 bucket where it is then removed from the queue.

- **Main S3 Buckets and its backups**

  The main S3 bucket is where all the processed images are stored to be accessed by the rest API layer. It has versioning enabled in the event that files are accidentally mass deleted. This bucket also has two backups occurring on it. The main backup bucket is stored in another AWS location and every object uploaded to main bucket is replicated to it. This greatly increases reliability in the case where the main bucket's host is lost. This backup bucket acts as the "SAN Raid" requirement.

  The second backup solution is where all the contents of the main bucket are stored in an S3 glacier drive. This can only be done once every 30 days otherwise you will incur an increased cost. As such, the bucket is set up to only backup itself to glacier every 30 days. This drive is for extra redundancy in the event that the other two buckets fail. Glacier is very cheap compared to the other S3 solutions. Glacier acts as the "tape backup" requirement.

- **Amazon RDS**

  The database solution is called RDS. I propose that the RDS service be available in multiple AZs through a subnet group. This will allow the different services in the AZs communicate with the database much quicker and also increase reliability in the event that one of the AZs becomes unavailable.

Security and Monitoring

- **Multiple Security groups**

  By giving each vertical layer its own security group, it allows the concerns for each layer to be isolated. Such that only the layers above it can communicate directly with it through specific interfaces as not to allow unwanted traffic through. This increases the security such that the group is a whitelist how the layer can function.

- **Multiple Subnets**

Same with the security groups, by giving each horizontal layer in each AZ its own subnet, it gives each layer its own subnet range to isolate it with. Only the top layer can access the internet and the rest must go through the layer above it. This makes each descending layer more secure from attacks from the internet. This can be used in conjunction with the security group to limit the traffic to only come from a specific subnet.

- **Amazon CloudWatch**

  To monitor the solution and ensure its operation, I propose to use the CloudWatch service to view and monitor the operating services. It logs when scaling groups are changing how many instances it uses and how much network traffic is being exchanged between the different layers.

- **Amazon Cognito**

  Amazon Cognito is a managed solution for managing user identities to interact with the services. This is for the user authentication in the Webapp and RestAPI. This will reduce complexity, increase security and reliability since the user does not have to manage it directly and concern over storage of the credentials since it is managed by Amazon.

Disaster recovery planning

Scenario: AZ Failure

- **RPO**

  The database is a multi-AZ database so if one AZ were to become unavailable, the other one should be able to continue as normal. The Amazon S3 buckets are not in any specific AZ so they are unaffected. The compute instances are also multi-AZ so they should scale accordingly if one AZ goes down.

- **RTO**

  The compute instances should be able to immediately recover, maybe 10-20 seconds for the new instances to spin up if the load triggers the auto scaling group in the remaining AZ. The website should still be available instantly but with less functionality from CloudFront edge locations.

Scenario: Region Failure

- **RPO**

  All data stored on the main S3 bucket is replicated to another S3 bucket in another region, so all data should be retained.

- **RTO**

  The compute instances will take 10-30 seconds to recover after the disaster, with all data being retained via the cross-region replication S3 bucket. The website should still be available instantly but with less functionality from CloudFront edge locations.

Scenario: Main S3 bucket and backup bucket deletion

- **RPO**

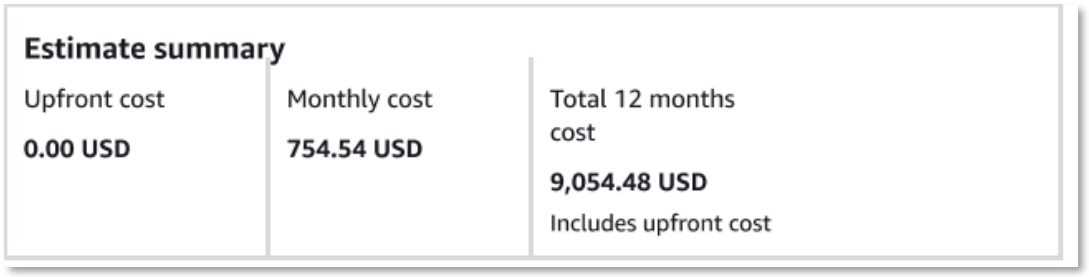
  All data stored on the main S3 bucket is replicated to an S3 glacier bucket every 30 days. So, the RPO is 1-30 days.

- **RTO**

  The S3 glacier bucket is on a glacier flexible retrieval plan which should restore data within 1-5 minutes, depending on how much data is stored.

Monthly usage cost estimation

Amazon Calculator Report with estimated workload



	Name	Group	Region	Upfront Cost	Monthly Cost
Service	Amazon CloudFront	No group	US East (N. Virginia)	0.00 USD	107.57 USD
Config Summary	Number of requests (HTTPS) (50000 per month), Data transfer out to origin (1 TB per month), Data transfer out to internet (1 TB per month)				
Discussion	This is with an overestimation of 50000 users per month, each uploading/downloading 200mb of data.				
Service	Amazon EC2	No group	US East (N. Virginia)	0.00USD	92.00USD
Config Summary	Operating system (Linux), Storage amount (30 GB), DT Inbound: Not selected (0 TB per month), DT Outbound: Not selected (0 TB per month), DT Intra-Region: (0 TB per month), Workload (Daily, (Workload days: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Baseline: 1, Peak: 20, Duration of peak: 8 Hr 30 Min)), Snapshot Frequency (2x Daily), Amount changed per snapshot (3 GB), Advance EC2 instance (t4g.nano), Pricing strategy (EC2 Instance Savings Plans 1 Year None upfront)				
Discussion	This is an estimation of load times throughout the week, cyclical with daylight hours and estimating that 20 EC2 instances are needed in peak times.				
Service	Amazon RDS for MySQL	No group	US East (N. Virginia)	0.00 USD	87.20USD

Config Summary	Storage for each RDS instance (General Purpose SSD (gp2)), Storage amount (30 GB), Quantity (1), Instance type (db.m1.small), Utilization (On-Demand only) (100 %Utilized/Month), Deployment option (Multi-AZ), Pricing strategy (OnDemand)				
Discussion	Estimating that there are 500,000 registered users with the site, each with 60kb of stored information.				
Service	Elastic Load Balancing	No group	US East (N. Virginia)	0.00 USD	79.57USD
Config Summary	Number of Application Load Balancers (2)				
Discussion	Load balancers for the rest api and website, there are no configuration options.				
Service	Amazon Simple Queue Service (SQS)	No group	US East (N. Virginia)	0.00 USD	0.00 USD
Config Summary	DT Inbound: Internet (1 TB per month), DT Outbound: Amazon CloudFront (1 TB per month), Data transfer cost (0), Standard queue requests (0.5 million per month), FIFO queue requests (0.05 million per month)				
Discussion	SQS is free, granted that it is used internally in the cloud.				
Service	Amazon Simple Storage Service (S3)	No group	US East (N. Virginia)	0.00 USD	23.10 USD
Config Summary	S3 Standard storage (100 GB per month) DT Inbound: All other regions (1 TB per month), DT Outbound: All other regions (1 TB per month)				
Discussion	Main S3 storage, estimating that 1tb of traffic is used per month.				
Service	Amazon Simple Storage Service (S3)	No group	US East (N. Virginia)	0.00 USD	23.10 USD
Config Summary	S3 Standard storage (100 GB per month) DT Inbound: All other regions (1 TB per month), DT Outbound: All other regions (1 TB per month)				
Discussion	Backup replicated S3 storage, estimating that 1tb of traffic is used per month.				
Service	Amazon Simple Storage Service (S3)	No group	US East (N. Virginia)	0.00 USD	24.15 USD
Config Summary	S3 Glacier Average Object Size (16 MB), S3 Glacier storage (1000 GB per month) DT Inbound: All other regions (1 TB per month), DT Outbound: All other regions (1 TB per month)				
Discussion	Glacier S3 storage, with 1 backup per month				

Service	Amazon Cognito	No group	US East (N. Virginia)	0.00USD	256.75USD
Config Summary	Advanced security features (Enabled), Number of monthly active users (MAU) (5000)				
Discussion	Estimation of 50000 active monthly users.				
Service	Amazon Elastic File System (EFS)	No group	US East (N. Virginia)	0.00USD	61.10 USD
Config Summary	Desired Storage Capacity (50 GB per month), Infrequent Access requests (1 0 GB per month), Provisioned Throughput (1 0 MB/s per month)				
Discussion	EFS for internal data transfer of 50gb per month, with 10MB/s bandwidth.				