CCT College Dublin

Assessment Cover Page

Module Title:	Cloud Services
Assessment Title:	HW-10%
Lecturer Name:	Micheal Weiss
Student Full Name:	Muhammad Sajjad Haider
Student Number:	2021384
Assessment Due Date:	27th October 2024
Date of Submission:	27th October 2024

Table of Contents

Question no:1

As the virtual firewalls, the security groups regulate traffic to components of a three-tier application architecture. The dividing in layers means there are Web Tier, App Tier, and Data Tier available in the architecture. Flows is the response for the general rules which is applied to each of these tiers and security groups is the response for rules that regulates and controls the type of traffics that is allowed to get in.

Security Group for Web Tier Elastic Load Balancer (ELB):

Goal: The goal of this layer is to perform the first tasks related to web usage from the users who wish to engage the application online.

Rule: Any IP address outside the user's private address space is allowed to transfer Secure HTTP (S-HTTP) traffic on the port 443 thus enabling users to access the website through Secure HTTP.

That is why Only HTTPS-encrypted communications are allowed. It shields information crossing in between the user's browser and the load balancer.

Group for Web Tier Security:

Goal: Pages are served to the web servers and the traffic does go through the load balancer.

Rule: Outside of the load balancer, no direct internet traffic is allowed to connect through port 80 or HTTP.

That is why it is secure because only the traffic going to the web servers from the load balancer is allowed. Further protection is augmented by the fact that there is no direct physical access possible to most zones.

ELB Security Group for App Tier:

Goal: At the present time, traffic is being transferred between the web tier and the application tier, where the application is implemented.

Rule: It allows web tier traffic in port 8080 which is commonly used for app to communicate.

The reason it is safe This ensures that only traffic within the access point to the application layer can be from the web tier only. The application servers are not directly accessible by the users from the world outside the organization.

Security Group for App Tier:

Goal: This rule provides a check to the amount of traffic that is received by the app servers.

Rule: Facilitating only traffic from app tier load balancer on configured port 8080 with no other traffic From All Other Sources.

The reason it is safe Therefore a restricted path is formed so only certain traffic is permitted to get to the app servers. Due to the isolation of the app servers from other traffic, only the essential communication is allowed.

Group for Data Tier Security:

Goal: At this last level of the architecture, the data is stored in the database.

Rule: Only the communication originating from the app tier is allowed on port 3306 for MySQL database.

Why it is secure: No other user including those in the web tier can directly interface with the database. Due to the fact that the database can only be accessed by the app tier, this information kept here is safe.

Security Groups Offer Protection

The Way in Which These Security Groups Offer Protection:

Security Layers:

This is because traffic between the different tiers the Web, App, as well as Data tiers has to follow specific paths. This helps to ensure that even when an attacker gets into the system he cannot administer the whole system.

Access Controlled:

While external access is allowed only at the web layer, and through load balancer with secured HTTPS connections. When the public is exposed to any part of the system, it is not to the complete other component of the system.

Limited Access to Data:

The database is hidden from the web servers; it is also hidden from the general public but is available only for the app servers. This ensures that as soon as the database is developed it will not be vulnerable no matter the level of compromise of the web tier.

Reduced Surface of Attack:

To eliminate possible open doors for assaults, each security group is designed to allow traffic only on some ports mandatory for the application, which are 443, 80, 8080, and 3308.

In conclusion, for this configuration, security groups operate together to ensure that each system tier allows only the specific amount of traffic that it needs to, greatly increasing the application's security as well as the data stored in the database.

Question no:2

Decentralized application architecture exhibited in the diagram above has a glaring issue of single point failure at the database server level. This is problematic since failure in any database causes an application, especially when the desired availability is 5 nines, or 99.999%. Here are some suggestions to increase this architecture's availability and satisfy the uptime requirement:

Create multi-AZ databases

Use Amazon Aurora or multi-AZ deployment Amazon RDS. Multi-AZ enables the applications and the databases to have enhanced availability, and switch to a standby database in other Availability Zone during outages or maintenance.

Elastic Load Balancer (ELB) and Auto-Scaling

What does that mean? For both Web and Application servers, use auto-scaling. This ability also allows for managing the architectural infrastructures for traffic load on Web services by scaling up or down the number of instances.

Prerequisites: Place the application servers behind an Elastic Load Balancer. This distributes traffic equally to ensure none of the servers are congested with the traffic and also ensures that traffic is redirected to only active server.

Opt for a Dispersed database structure

Make use of database read replicas for workloads that involve many reads especially at this point that traffic is high. By doing this, much of the traffic is shifted away from the main database. If you feel the application will derive the type of value that comes with a

completely managed NoSQL service that guarantees high availability and scaling, then Amazon DynamoDB should be your choice.

Containers and Microservices

For Containerized applications you should apply Elastic Kubernetes Service (AKS) or Amazon ECS. Thus, these services broaden system scalability and reliability as long as distributed microservice operations are optimized.

To reduce the impact of the SPOF, each microservice might have a shared distributed database design, or own a database.

Automation and Monitoring

Monitor the condition of the infrastructure by using Amazon CloudWatch for this purpose. Build notifications for performance metrics, for instance, CPU, RAM, and database, Auto-scaling and Auto-recovery based on these signals should be created.

In order to achieve power your system with self-healing capabilities, utilize the AWS Auto-Healing mechanism through the utilization of the EC2 Auto-Recovery to recover failed instances.

Backup system

Backup of the system's replica for different regions which was considered essential in the case of the Centre for Excellence in Applied Arts and Health.

Use cross-region replication so that data of important databases is mirrored to a different region. It can be rolled back to the second area in case of a failure that involves an entire geographical region.

High availability can be easily implemented through having application servers in several regions available within AWS and Global DNS (if using Amazon's Route 53). This will allow one to have low latency retrieve and geo-based failover.

These principles, when implemented, will enable you to get an uptime of 99.999 %, eliminate single points of control, and scale your resources flexibly.

Question no3:

A:

In order to enhance the AWS architecture's availability, you need concentrate on multiple crucial tactics:

Multi-Availability Zone (AZ) Deployment:

It is recommended to ensure that some elements like databases – RDS or EC2 instances, are located on various AZs. This is important in an ASG to make certain that there is uninterrupted service even in the event that one AZ faces a failure and therefore minimizes the quality of an outage in any one AZ.

Auto-Scaling:

Choose auto-scaling for the EC2 instances to deal with fluctuating traffic conditions For more on this, you can check out this link. One of the things that the application can help to reduce costs during periods of low demand and ensure maximum availability during periods of increased load is to add or remove cases on its own.

Elastic Load Balancing (ELB):

With that you ready, if you have not already, use Elastic Load Balancers to help distribute incoming traffic through several different EC2 instances. This in a way increases availability because the load in the groups splits the load to healthy instances where one instance in a group may have failed.

Cross-Region Replication:

Consider spreading resources throughout numerous AWS regions in the event the application must be available everywhere around the world. This reduces the duration in which critical services fail by making sure that even an entire region can be serviced by another region. It also offers geographical diversity as well;

Data Backups and Snapshots:

Remember to stimulate backups for your data, and also taking the EC2 instances and databases snapshots. This ensures that in a failure or some form of data corruption, recovery is very fast to reduce on large loses.

Amazon Route 53 with Health Checks:

Combine Amazon Route 53 for fail over routing and DNS service with health checks. Route 53 can automatically shift traffic to a healthy instance in other region or Arizona if one region or instance became unhealthy.

RDS Read Replicas or Multi-AZ RDS:

For the improvement of read accessibility and performance, use read replicas or RDS Multi-AZ to support failover in databases.

Serverless Services:

At every opportunity, leverage ‘server-less’ services that don’t require infrastructure management and are inherently available due to services such as AWS Lambda or DynamoDB.

The effectiveness of the above best practices will go a long way in enhancing the availability of AWS architecture.

B:

The online store should take into consideration the following three inherently high-availability (HA) services in order to manage a notable and prolonged rise in web traffic over the Christmas period:

Justification:

Amazon Elastic Load Balancing (ELB) Inbound application traffic is by default distributed to several targets, for instance, EC2 instances. It ensures traffic is channeled to the healthy instances and does help prevent compromising any instance to be overwhelmed with traffic. When multiple application servers are deployed behind a load balancer, resilience and system accessibility enhance when going through intensive traffic at particular moments. If, for instance, some server is not running, then ELB has the capacity to route the traffic it receives to other instances that are already running, thus retaining the high quality of experience for the consumers.

Auto Scaling by Amazon:

Through auto scaling the number of instances that are actually running can be adjusted based on traffic load as needed. In the case when site traffic increases during the Christmas season, Auto Scaling is capable of starting additional EC2 instances to handle

the extra load. After the traffic density lessens, it can cut the number of instances, which contributes to better control of costs. This ensures that even though traffic maybe low at some points the application is still up and running in case the traffic increase.

Amazon CloudFront (Content Delivery Network):

CloudFront is a system that decreases stress and delay of origin servers preparing at edge sites all over the world to cache material. To reduce load on application server during traffic surge, cached objects (pictures, CSS, JavaScript, etc.) should be served from edge locations. Therefore, they do not risk overloading the origin servers and overall increases application availability to the users and also the user loading times. Also, there is integration with DDoS defense in CloudFront and it makes the work of CloudFront more secure and less depending on the third-party resources.

The online store may make sure their application is available and responsive even during the busy Christmas season by utilizing these HA services.

References:

AWS (2024). Control traffic to resources using security groups - Amazon Virtual Private Cloud. [online] docs.aws.amazon.com. Available at: https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security-groups.html.

Watt, A. and Eng, N. (2019). Chapter 9 Integrity Rules and Constraints – Database Design – 2nd Edition. [online] Opentextbc.ca. Available at: https://opentextbc.ca/dbdesign01/chapter/chapter-9-integrity-rules-and-constraints/.

exabeam (2022). Information Security: Goals, Types and Applications. [online] Exabeam. Available at: https://www.exabeam.com/explainers/information-security/information-security-goals-types-and-applications/.

Examtopics.com. (2020). Exam Associate Cloud Engineer topic 1 question 54 discussion - ExamTopics. [online] Available at:

https://www.examtopics.com/discussions/google/view/18614-exam-associate-cloud-engineer-topic-1-question-54-discussion/
[Accessed 20 Oct. 2024].

ResearchGate. (n.d.). (PDF) Cloud Computing of E-commerce. [online] Available at: https://www.researchgate.net/publication/329417336_Cloud_Computing_of_E-commerce.

www.linkedin.com. (n.d.). Impact of COVID-19 on Cloud Computing. [online] Available at: https://www.linkedin.com/pulse/impact-covid-19-cloud-computing-harsh-siriah.

Amazon Web Services, Inc. (n.d.). Reference Architecture Examples and Best Practices. [online] Available at: https://aws.amazon.com/architecture/?cards-all.sort-by=item.additionalFields.sortDate&cards-all.sort-order=desc&awsf.content-type=.

Martyniuk, Y. (2023). AWS Technology: Architecting (Questions) - Yurii Martyniuk - Medium. [online] Medium. Available at: https://ethtool.medium.com/aws-technology-architecting-q-a-a196300c6750
[Accessed 20 Oct. 2024].