Final Project – AWS Cloud

Architecting

ITC 6480

Professor: Carmen Taglienti

David Gao

Yateeswar Chennamsetti

YuanYu Huang

# Introduction

In today's rapidly evolving digital landscape, businesses are increasingly turning to cloud solutions to address their technological needs. This shift is particularly evident in the healthcare sector, where the demand for innovative, scalable, and secure solutions is paramount. This paper explores the design of a cloud-based infrastructure for a startup SaaS company in the medical field, referred to as A Medical Company, aiming to provide online medical social networking and diagnosis assistance services to users across APAC, the US, and Europe.

The project at hand involves architecting an AWS infrastructure to meet A Medical Company's application needs. Currently, the company's development and test infrastructure are hosted by a third-party server hosting company, utilizing Microsoft Windows servers for the web and application tiers, and Microsoft SQL Server Standard Edition for backend databases. With an impending application launch and expectations of high user traffic, A Medical Company seeks to leverage cloud technologies to support its growth and host its development, test, and production environments on a new cloud platform.

This paper will delve into the communication challenges faced when applying technology as a solution to business problems, the translation of customer requirements into a proposed technical solution, and the presentation of this solution to the customer. By addressing these key aspects, this project aims to provide a comprehensive cloud solution that not only meets the immediate needs of A Medical Company but also sets the stage for its future scalability and success in the competitive healthcare market.

# Executive Summary

This project focuses on designing a cloud-based infrastructure for A Medical Company, a startup SaaS company, to support their online medical social networking and diagnosis assistance application. The project aimed to address the challenges faced by A Medical Company in managing their IT infrastructure, particularly with the impending launch of their application and the expected increase in user traffic.

To achieve this, the project team thoroughly reviewed the customer requirements and current environment of A Medical Company. Based on this analysis, a comprehensive cloud solution was designed using AWS services. The solution includes the identification of AWS services, user authentication mechanisms, network and security configurations, web and application tier architecture, business continuity strategies, and auditing processes.

The outcome of the project is a scalable, secure, and efficient cloud-based infrastructure tailored to the specific needs of A Medical Company. The solution not only meets the immediate requirements of the company but also positions them for future growth and success in the competitive healthcare market. The recommendation is to proceed with the implementation of the proposed cloud solution, ensuring proper training and support for A Medical Company's team to manage and maintain the infrastructure effectively.

# Overall Requirments and Assumptions

## Overall Requirements

Designing a cloud-based architecture for A Medical Company's online medical social networking and diagnosis aid application was the main requirement for this project. By facilitating online appointments, remote consultations, remote diagnostics, electronic prescription transfers, and payment services, this program links patients and physicians. The system required to enable the uploading of files, including the ability to convert picture formats and extract text from documents. In addition, the system had to offer a stable and expandable environment for development, testing, and production in addition to handling the anticipated surge in user traffic following the application's introduction. Other essential criteria included data security, high availability, and adherence to healthcare laws.

## Assumptions As the cloud solution for A Medical Company was being designed, the following crucial presumptions were made:

· Soon after its introduction, the program would see a large spike in user traffic, requiring a scalable infrastructure to handle this strain.

· Because they understood the fundamentals of cloud computing and AWS services, the company's IT personnel were able to handle the new infrastructure with ease.

· The new cloud architecture would improve the company's robust backup and recovery plan by introducing automated and centralized backup options.

· The business's security and privacy policies for data were in line with industry norms, especially in the healthcare field, and the suggested AWS services would abide by these rules to guarantee data security and privacy.

# Architecture

Amazon EC2 & Auto Scaling

Purpose: Amazon EC2 provides resizable compute capacity in the cloud, which is essential for hosting the application and web servers of the SaaS platform.

Connection: EC2 instances will be dynamically managed by Auto Scaling groups, which ensure that the infrastructure scales up or down based on actual demand. This is particularly useful for handling high load periods effectively without incurring unnecessary costs during off-peak times.

Amazon RDS for SQL Server

Purpose: Manages relational database needs, offering automated backups, patch management, and failover support, crucial for the high availability and durability of database services.

Connection: Integrates seamlessly with EC2 instances, providing a reliable data source for the application, and supports Multi-AZ deployments for disaster recovery and high availability.

Amazon S3 & Glacier

Purpose: S3 provides a highly durable storage location for the static content and user uploads, which is scalable and secure. Glacier is used for cost-effective long-term archival storage, especially important for medical records and compliance data.

Connection: S3 can serve as a data lake for logs and unstructured data which can be analyzed using other AWS services. Glacier archives can be accessed by applications when needed, though less frequently, ensuring compliance with data retention policies.

Amazon VPC

Purpose: Offers a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. It’s crucial for security and network isolation.

Connection: Within the VPC, subnets can be used to isolate and control network design for public-facing application components and backend systems that should not be directly exposed to the Internet. Security groups and Network ACLs provide additional layers of security.

Elastic Load Balancing (ELB)

Purpose: Automatically distributes incoming application traffic across multiple EC2 instances, improving fault tolerance and increasing the application's fault tolerance.

Connection: It helps to ensure that no single instance bears too much load and that the application remains highly available. It works directly with Auto Scaling to distribute traffic as new instances are launched and terminated.

AWS IAM

Purpose: Manages access to AWS services and resources securely. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

Connection: IAM roles and policies are applied to all other services to ensure that they are accessed securely and according to the principle of least privilege.

AWS Direct Connect

Purpose: Establishes a dedicated network connection from an on-premises to AWS. It can reduce network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections.

Connection: Useful for the seamless transfer of critical data between the company’s data centers and AWS, ensuring consistent latency and secure data transfer.

AWS CloudWatch & AWS CloudTrail

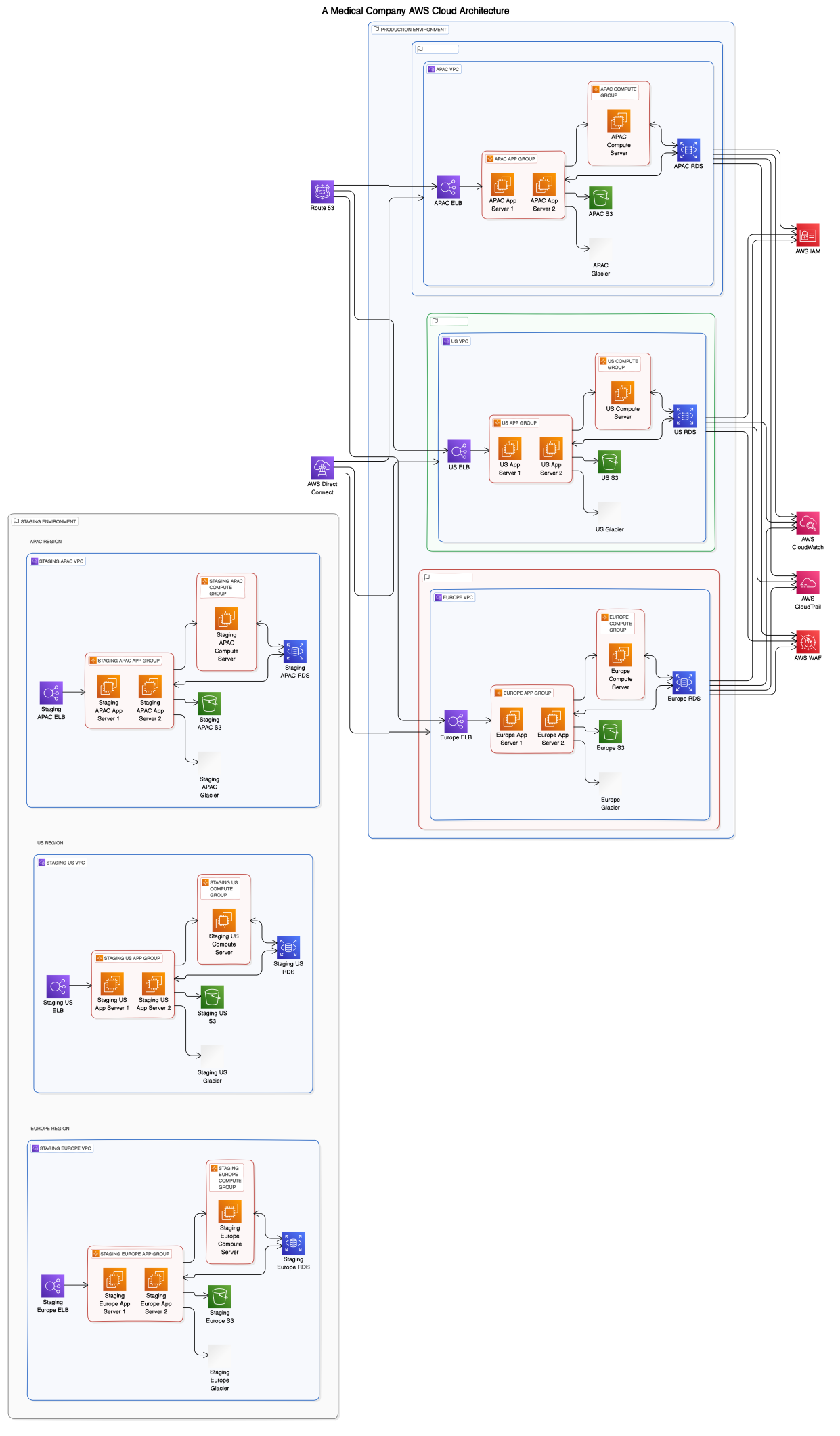
Purpose: CloudWatch provides monitoring for AWS cloud resources and the applications running on AWS, whereas CloudTrail logs every single API call, making it possible to audit changes.

Connection: CloudWatch can trigger alarms based on metrics which can initiate Auto Scaling; CloudTrail logs are used for security audits and real-time monitoring of API calls to AWS resources.

AWS WAF

Purpose: Helps to protect the web applications from common web exploits that could affect application availability, compromise security, or consume excessive resources.

Connection: Directly attached to ELB (or the more specific Application Load Balancer) to filter traffic before it reaches the application, ensuring that only legitimate traffic is processed.



# Solution

## Identify AWS Services

To build a robust, scalable, and secure infrastructure for the Medical Company's application, we need to carefully select AWS services that align with the current and future needs of the business. Below are the AWS services identified to meet the requirements of each tier and component of the application.

1. **Compute Resources:**
   * **Amazon EC2 (Elastic Compute Cloud):** EC2 instances will be used for the web and application tiers. This service provides resizable compute capacity in the cloud, allowing the Medical Company to handle variable traffic and scale up or down as needed. Instances with similar specifications to the current environment will be selected (e.g., t3.medium for the web tier and m5.xlarge for the application tier).
   * **Auto Scaling:** EC2 Auto Scaling will be implemented to automatically adjust the number of EC2 instances during demand spikes, ensuring high availability and cost efficiency.
2. **Load Balancing:**
   * **Amazon Elastic Load Balancer (ELB):** ELB will be used to distribute incoming application traffic across multiple EC2 instances in the web and application tiers. This ensures fault tolerance and improves the overall performance of the application.
3. **Database:**
   * **Amazon RDS (Relational Database Service):** Amazon RDS for SQL Server will be used to manage the database tier. RDS offers automated backups, software patching, and scaling without requiring DBA intervention, enhancing reliability and availability. A Multi-AZ deployment will be configured for disaster recovery.
4. **Storage:**
   * **Amazon S3 (Simple Storage Service):** S3 will be used for storing documents and images uploaded by customers. S3 provides durable and highly available object storage, ensuring that data is securely stored and easily accessible.
   * **Amazon S3 Glacier:** For long-term archival storage, S3 Glacier will be used to store documents and images that are not frequently accessed but need to be retained for compliance or historical purposes.
5. **Networking:**
   * **Amazon VPC (Virtual Private Cloud):** A VPC will be created to provide network isolation and secure communication between different components of the application. Subnets, security groups, and network ACLs will be configured to enhance security.
   * **AWS Direct Connect:** If needed, AWS Direct Connect can be used to establish a dedicated network connection between the Medical Company’s on-premises environment and AWS, ensuring low latency and secure connectivity.
6. **Security:**
   * **AWS IAM (Identity and Access Management):** IAM will be used to manage user access and permissions. Fine-grained access control policies will be defined to ensure that only authorized users have access to specific resources.
   * **AWS WAF (Web Application Firewall):** WAF will be implemented to protect the application from common web exploits and vulnerabilities.

|  |
| --- |
| **Amazon EC2**: To provide scalable compute capacity for the web and application tiers. |
| **EC2 Auto Scaling**: To automatically adjust the number of EC2 instances based on traffic demand. |
| **Amazon Elastic Load Balancer (ELB)**: To distribute incoming application traffic across multiple EC2 instances, ensuring fault tolerance and high performance. |
| **Amazon RDS for SQL Server**: To manage the database tier with automated backups, patching, and scaling capabilities. |
| **Amazon S3**: To store documents and images uploaded by customers, providing durable and highly available object storage. |
| **Amazon S3 Glacier**: For long-term archival storage of documents and images that need to be retained for compliance. |
| **Amazon VPC**: To create a logically isolated network for secure communication between application components. |
| **AWS Direct Connect**: For dedicated network connections between on-premises environments and AWS, ensuring low latency and secure connectivity. |
| **AWS IAM**: To manage user access and permissions securely, with fine-grained access control and Multi-Factor Authentication (MFA). |
| **AWS WAF**: To protect the application from common web exploits and vulnerabilities. |
| **AWS CloudTrail**: To log all API calls within the AWS environment for compliance and security analysis. |
| **AWS CloudWatch**: To monitor and collect metrics from all AWS resources, set up alarms, and create custom dashboards for system health. |
| **AWS Config**: To track and record configurations of AWS resources, ensuring compliance with best practices. |
| **AWS GuardDuty**: To continuously monitor for malicious activity and unauthorized behavior, integrating with AWS Lambda for automated responses. |

## User Authentication

#### **AWS Identity and Access Management (IAM)**

**AWS IAM** plays a critical role in managing access to AWS services and resources securely. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources. IAM is crucial for any application, especially in healthcare, where data sensitivity is paramount.

**Key Features:**

* **Fine-Grained Access Control:** IAM enables you to define policies with precision, ensuring users and systems have only the permissions necessary to perform their tasks. For instance, developers can have differing access levels to resources like EC2 instances, RDS databases, and S3 buckets.
* **Multi-Factor Authentication (MFA):** MFA adds an extra layer of security by requiring users to provide two forms of identification before being granted access. This is vital for administrators accessing sensitive environments.
* **Integration with Corporate Directories:** IAM can integrate with your existing corporate directories using AWS Directory Service, making user management seamless and secure.
* **Identity Federation:** This allows users to authenticate with external identity providers (IdPs) — consolidating and simplifying access management across AWS and on-premises systems.

A Medical Company Account

Role:

Group:

Group:

Database Administrator

Audit and Compliance Group

administrators group

Network Administrator

developers group

operations group

|  |  |  |
| --- | --- | --- |
| Group/Role # | Group/Role Name | Permissions |
| Group | Administrators Group | Full access to all AWS resources.  Ability to manage IAM roles and policies.  Permissions to configure and manage all aspects of the AWS account. |
| Group | Developers Group | Access to AWS development tools like AWS CodeCommit, CodeBuild, CodeDeploy, Elastic Beanstalk, and limited access to Amazon EC2, Lambda, and other compute resources necessary for application development. |
| Group | Operations Group | Read and write access to Amazon EC2, Elastic Load Balancing, Auto Scaling Groups, Amazon CloudWatch, AWS Systems Manager, and limited access to AWS networking services. |
| Role | Audit and Compliance Group | Read-only access to AWS Config, AWS CloudTrail, and access to Amazon S3 buckets where logs and audit trails are stored. |

|  |  |
| --- | --- |
| Requirement | Solution |
| Should be at least 8 characters and have 1 uppercase, 1 lowercase, 1 special character, and a number. | Configure AWS IAM password policy to enforce these requirements. This can be done under the account settings in the IAM dashboard where you can set minimum password length and require specific character types. |
| Change passwords every 90 days and ensure that the previous three passwords can’t be re-used. | IAM password policy settings, enable password expiration after 90 days and password reuse prevention set to the last three passwords. |
| All administrators require programmatic access | Provide each administrator with AWS IAM users credentials including access keys that can be used to interact with AWS services through the API, CLI, or SDK tools. |
| Administrator sign-in to the AWS Management Console requires the use of Virtual MFA. | Enforce MFA for all IAM users with console access, particularly administrators. This can be configured in the IAM dashboard where you can set up and manage MFA devices. |

## Network and Security

#### **Amazon Virtual Private Cloud (VPC)**

**Amazon VPC** allows you to provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

**Key Features:**

* **Subnets:** Allows you to segment your network, and control the availability and reliability by spreading your architecture across multiple availability zones.
* **Security Groups and Network ACLs:** These act as a virtual firewall for your instances to control inbound and outbound traffic at the instance and subnet level, respectively.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| VPC | Region | Purpose | subnets | AZs | CIDR range |
| Production Environment | us-east-1 | Host the production environment for the SaaS application, ensuring security, performance, and high availability | Public Subnet for Web Servers (us-east-1a)  Private Subnet for Application Servers (us-east-1b)  Private Subnet for Database Servers (Multi-AZ: us-east-1b, us-east-1c) | us-east-1a, us-east-1b, us-east-1c | 10.0.0.0/16 (change as required) |
| Staging Environment | (Assuming "us-east-1", as consistency in the same region as production for staging is typical but could be different for isolation) | Support testing and staging of the application separate from production to mimic the production environment without impacting the live user base. | Public Subnet for Web Servers (us-east-1a)  Private Subnet for Application Servers (us-east-1b) | us-east-1a, us-east-1b | 10.1.0.0/16 (change as required) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Subnet Name | VPC | Subnet Type (Public/private) | AZ | Subnet Address |
| Dev-Web-Public | #1 | Public | us-east-1a | 10.0.1.0/24 |
| Dev-App-Private | #1 | Private | us-east-1b | 10.0.2.0/24 |
| Dev-DB-Private | #1 | Private | us-east-1c | 10.0.3.0/24 |
| Dev-Backup-Private | #1 | Private | us-east-1a | 10.0.4.0/24 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Subnet Name | VPC | Subnet Type (Public/private) | AZ | Subnet Address |
| Test-Web-Public | #2 | Public | us-east-1b | 10.0.5.0/24 |
| Test-App-Private | #2 | Private | us-east-1c | 10.0.6.0/24 |
| Test-DB-Private | #2 | Private | us-east-1a | 10.0.7.0/24 |
| Test-Backup-Private | #2 | Private | us-east-1b | 10.0.8.0/24 |

## Web and Application Tier

Amazon EC2 (Elastic Compute Cloud) instances will be used for the web and application tier. This solution gives the business access to cloud computing capacity that can be scaled up or down based on demand, enabling it to manage fluctuating traffic. Examples of instances that match the specifications of the existing environment, such as t3.medium for the web tier and m5.xlarge for the application tier, will be chosen. The number of EC2 instances will be dynamically adjusted during demand spikes thanks to the implementation of EC2 Auto Scaling, which will guarantee high availability and economical operation. at order to provide fault tolerance and boost overall performance, inbound application traffic will be split among several EC2 instances at the web and application tiers using Amazon Elastic Load Balancer (ELB).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tier | Tag | OS | Type | Size | Justification | #of instances | User Data? |
| Web | Key = Name  Value = web-tier | Amazon Linux 2 | t3.medium | 2 vCPU, 4 GB RAM | Chosen for its balance of compute, memory, and network resources, suitable for handling web server workloads. Provides cost-effectiveness for the anticipated web traffic. | 2 (for high availability across two Availability Zones) | Yes (to bootstrap instances with web server configuration scripts) |
| App | Key = Name  Value = app-tier | Amazon Linux 2 | m5.large | 2 vCPU, 8 GB RAM | Selected for its increased memory size, catering to the application layer's need to run in-memory operations and more demanding workloads efficiently. | 2 (for high availability and load balancing) | Yes (for application deployment and configuration) |
| DB | Key = Name  Value = db-tier | Not applicable (Managed Database Service) | db.m5.large (Amazon RDS) | 2 vCPU, 8 GB RAM | The db.m5.large instance type is chosen for its balance of compute and memory resources, which is ideal for moderate database workloads. It provides enough performance for transactional database operations and supports growth in database demand without being overly costly. Opting for Amazon RDS (Relational Database Service) allows for managed database services, which include automated backups, patching, and scalability options. This choice supports high availability and disaster recovery requirements through Multi-AZ deployments. | 1 (Multi-AZ for high availability) | Not applicable (Managed Database Service) |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Load Balancer | Name\* | External/Internal | Subnets | SG Name\* | Rule | Source |
| For Web Tier | web-elb | External (to accept traffic from the internet) | Spread across multiple Availability Zones for high availability | web-elb-sg | Allow all inbound HTTP (80) and HTTPS (443) traffic | 0.0.0.0/0 (to accept requests from all IP addresses) |
| For App Tier | app-elb | Internal (to only accept traffic from within the VPC) | Spread across multiple Availability Zones for high availability | app-elb-sg | Allow inbound HTTP (80) traffic from web-elb-sg | web-elb-sg (to only accept requests from the web tier load balancer) |

|  |  |  |  |
| --- | --- | --- | --- |
| Instance Tier | SG Name\* | Rule | Source |
| Web Tier | web-tier-sg | Allow inbound HTTP (80) and HTTPS (443) traffic | web-elb-sg (to only accept requests from the web tier load balancer) |
| App Tier | app-tier-sg | Allow inbound HTTP (80) traffic from app-elb-sg | app-elb-sg (to only accept requests from the application tier load balancer) |
| Database Tier | db-tier-sg | Allow inbound MySQL (3306) traffic | app-tier-sg (to only accept requests from the application tier security group) |

## Business Continuity

We will make use of multiple AWS services to guarantee business continuity:

· AWS CloudTrail: Keeps track of every API call made inside the AWS environment and offers a thorough history of resource modifications.For compliance audits, resource change tracking, and security analysis, this is essential.

· Metrics are gathered and tracked from every AWS resource via AWS CloudWatch. To see system health and critical performance indicators, develop custom dashboards. Notifications can be set up in CloudWatch Alarms to occur when particular thresholds are crossed.

· AWS Config: Maintains track of and logs AWS resource configurations, making sure resources adhere to guidelines and offering thorough compliance reports.

With the help of analysis of account activity, network traffic, and data access patterns, AWS GuardDuty offers continuous monitoring for unauthorized activity and malicious activity. This helps to identify potential threats. Responses to attacks can be automated by integrating with AWS Lambda.​

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tier | OS | Type | Size | Configuration Name\* | Role | Security Group |
| Web | Amazon Linux 2 | t3.medium | 2 vCPUs, 4 GiB Memory | WebTier | Serve HTTP/HTTPS requests | web-tier-sg |
| App | Ubuntu Server 20.04 LTS | r5.large | 2 vCPUs, 16 GiB Memory | AppTier | Business Logic Processing | app-tier-sg |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tier | Launch Configuration\* | Group Name\* | Group Size | VPC | Subnets | ELB | Tags |
| Web | WebTier | WebTier | Min: 2, Desired: 4, Max: 8 | vpc-123abc | subnet-01abc, subnet-02abc, subnet-03abc | WebTierELB | Environment:Production, Tier:Web |
| App | AppTier | AppTier | Min: 2, Desired: 3, Max: 6 | vpc-123abc | subnet-04abc, subnet-05abc, subnet-06abc | AppTierELB | Environment:Production, Tier:App |

## Auditing

To ensure compliance, security, and operational efficiency, the Medical Company must have robust auditing mechanisms in place. AWS provides several services to facilitate comprehensive auditing and monitoring of cloud resources.

1. **AWS CloudTrail:**
   * **Logging API Calls:** AWS CloudTrail will be enabled to log all API calls made within the AWS environment. This includes actions taken through the AWS Management Console, AWS SDKs, command-line tools, and other AWS services.
   * **Security and Compliance:** CloudTrail logs provide a detailed history of changes made to resources, which is crucial for security analysis, resource change tracking, and compliance auditing. Logs can be stored in Amazon S3 and analyzed using Amazon Athena or third-party tools.
2. **AWS CloudWatch:**
   * **Monitoring Metrics:** CloudWatch will be used to collect and monitor metrics from all AWS resources. Custom dashboards will be created to visualize key performance indicators and system health.
   * **Alarms and Notifications:** CloudWatch Alarms will be configured to send notifications via Amazon SNS (Simple Notification Service) when specific thresholds are breached, allowing for proactive incident management.
3. **AWS Config:**
   * **Resource Configuration Tracking:** AWS Config will be used to track and record configurations of AWS resources. Config rules will be defined to ensure resources comply with specified configurations.
   * **Compliance Reporting:** AWS Config provides detailed compliance reports, making it easier to audit and ensure that all resources are configured according to best practices and regulatory requirements.
4. **AWS GuardDuty:**
   * **Threat Detection:** Amazon GuardDuty will be used for continuous monitoring of malicious activity and unauthorized behavior. GuardDuty analyzes network activity, account behavior, and data access patterns to identify potential threats.
   * **Automated Responses:** Integration with AWS Lambda can automate responses to detected threats, such as isolating compromised instances or notifying the security team.

By leveraging these AWS services, the Medical Company can build a secure, scalable, and compliant infrastructure that meets its business needs while providing the flexibility to grow and adapt as the application gains traction.

|  |
| --- |
| **AWS Services Used:** AWS CloudTrail AWS CloudWatch AWS S3 AWS IAM AWS Config |
| **CloudTrail Configuration:** Enable CloudTrail for all regions. Log management events and data events. |
| **S3 Bucket Setup:** Create an S3 bucket for log storage. Enable encryption and implement bucket policies. |
| **CloudWatch Setup:** Create CloudWatch logs and alarms. Configure SNS notifications. |
| **User Groups Management:** Create three user groups with IAM. Assign least privilege policies. Enable IAM Access Analyzer. |
| **Config Rules:** Enable AWS Config. Establish Config rules for compliance. |
| **Audit Trail:** Ensure all API calls are logged. Secure log storage in S3. |
| **Log Retention:** Implement S3 lifecycle policies. |
| **Monitoring:** Review logs and metrics regularly. Conduct periodic audits. |

# Next Steps and Conclusion

To move forward, we need to create a detailed project plan with timelines, milestones, and resource allocations. This will ensure that all stakeholders are informed and aligned. Additionally, defining a clear migration strategy is essential to transition smoothly from the current physical servers to the AWS environment. This strategy should address data migration, application deployment, and cutover plans to minimize downtime.

Next, we must set up the necessary AWS accounts with appropriate billing and security configurations. Defining IAM roles and policies will manage access securely, ensuring that only authorized personnel can access specific resources. This step lays the foundation for secure and efficient operations within AWS.

The infrastructure setup involves provisioning a VPC and subnets to establish a secure network architecture. Launching EC2 instances for the web and application tiers with Auto Scaling and ELB configurations will ensure high availability and fault tolerance. Additionally, setting up Amazon RDS for SQL Server, with Multi-AZ configuration for high availability and automated backups, is crucial for maintaining database reliability and performance.

Implementing security measures is critical. Setting up AWS WAF and GuardDuty will protect the application from threats. Configuring logging and monitoring services like CloudTrail, CloudWatch, and AWS Config will track activities and ensure compliance with industry standards. These measures will provide a robust security framework to protect sensitive data and maintain system integrity.

Data migration and application deployment follow. Utilizing AWS Data Migration Service (DMS) or other tools will enable smooth data migration from the on-premises database to Amazon RDS. Deploying the application on the newly provisioned EC2 instances requires careful configuration to ensure all dependencies are correctly set up. Thorough testing, including load testing, security testing, and user acceptance testing (UAT), will validate the new environment and address any issues.

Finally, the go-live phase involves executing the cutover plan and transitioning to the AWS infrastructure. Continuous monitoring using CloudWatch and other tools will maintain optimal performance and security post-migration. This ongoing monitoring will quickly identify and resolve any issues, ensuring a seamless user experience.

## Conclusion

The proposed AWS infrastructure for the Medical Company offers a scalable, secure, and high-performance solution to support its online medical social networking and diagnosis assistance application. By leveraging AWS services such as EC2, RDS, S3, and VPC, the company can ensure high availability, fault tolerance, and efficient resource utilization. Additionally, robust security measures and auditing mechanisms provided by AWS services like IAM, CloudTrail, and GuardDuty will help protect sensitive patient data and maintain compliance with industry standards.

Implementing this solution will not only meet the immediate needs of the Medical Company but also position it for future growth and expansion. The flexibility and scalability of the AWS cloud platform will enable the company to adapt to increasing user demand and evolving business requirements. As the application gains traction in APAC, the US, and Europe, the AWS infrastructure will provide the necessary foundation to support a growing user base and deliver a seamless, reliable user experience.

The next steps outlined will guide the implementation process, ensuring a smooth transition to the cloud and a successful deployment of the application. With careful planning, thorough testing, and continuous monitoring, the Medical Company can achieve its goals and establish itself as a leading provider of online medical services.

Reference:  
Amazon Web Services. Amazon EC2. [https://aws.amazon.com/ec2/](https://aws.amazon.com/ec2/" \t "_new)  
Amazon Web Services. AWS Identity and Access Management (IAM). [https://aws.amazon.com/iam/](https://aws.amazon.com/iam/" \t "_new)  
Amazon Web Services. Amazon S3. [https://aws.amazon.com/s3/](https://aws.amazon.com/s3/" \t "_new)  
Amazon Web Services. AWS CloudTrail. [https://aws.amazon.com/cloudtrail/](https://aws.amazon.com/cloudtrail/" \t "_new)  
Amazon Web Services. Amazon RDS. [https://aws.amazon.com/rds/](https://aws.amazon.com/rds/" \t "_new)  
Amazon Web Services. Amazon VPC. [https://aws.amazon.com/vpc/](https://aws.amazon.com/vpc/" \t "_new)  
Amazon Web Services. AWS CloudWatch. [https://aws.amazon.com/cloudwatch/](https://aws.amazon.com/cloudwatch/" \t "_new)  
Amazon Web Services. AWS GuardDuty. [https://aws.amazon.com/guardduty/](https://aws.amazon.com/guardduty/" \t "_new)  
Amazon Web Services. AWS WAF. [https://aws.amazon.com/waf/](https://aws.amazon.com/waf/" \t "_new)  
Amazon Web Services. AWS Config. [https://aws.amazon.com/config/](https://aws.amazon.com/config/" \t "_new)