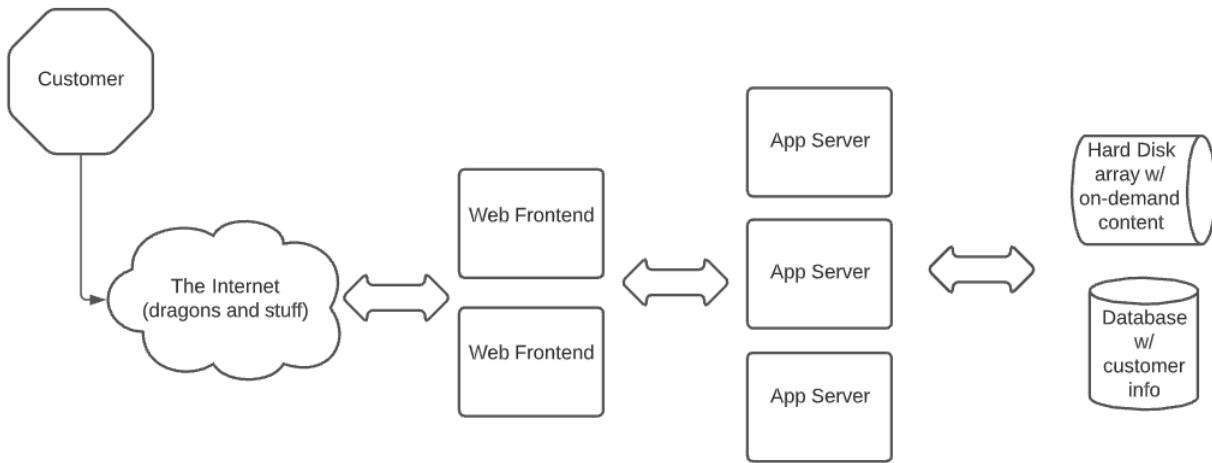


Final Project

Abstract → The document outlines a detailed solution to implement the technical plan (which was presented earlier) to migrate the architecture of CobraKai to AWS Cloud. The document provides strategies to resolve the prevalent issues in the on-premises architecture of cobra kai.

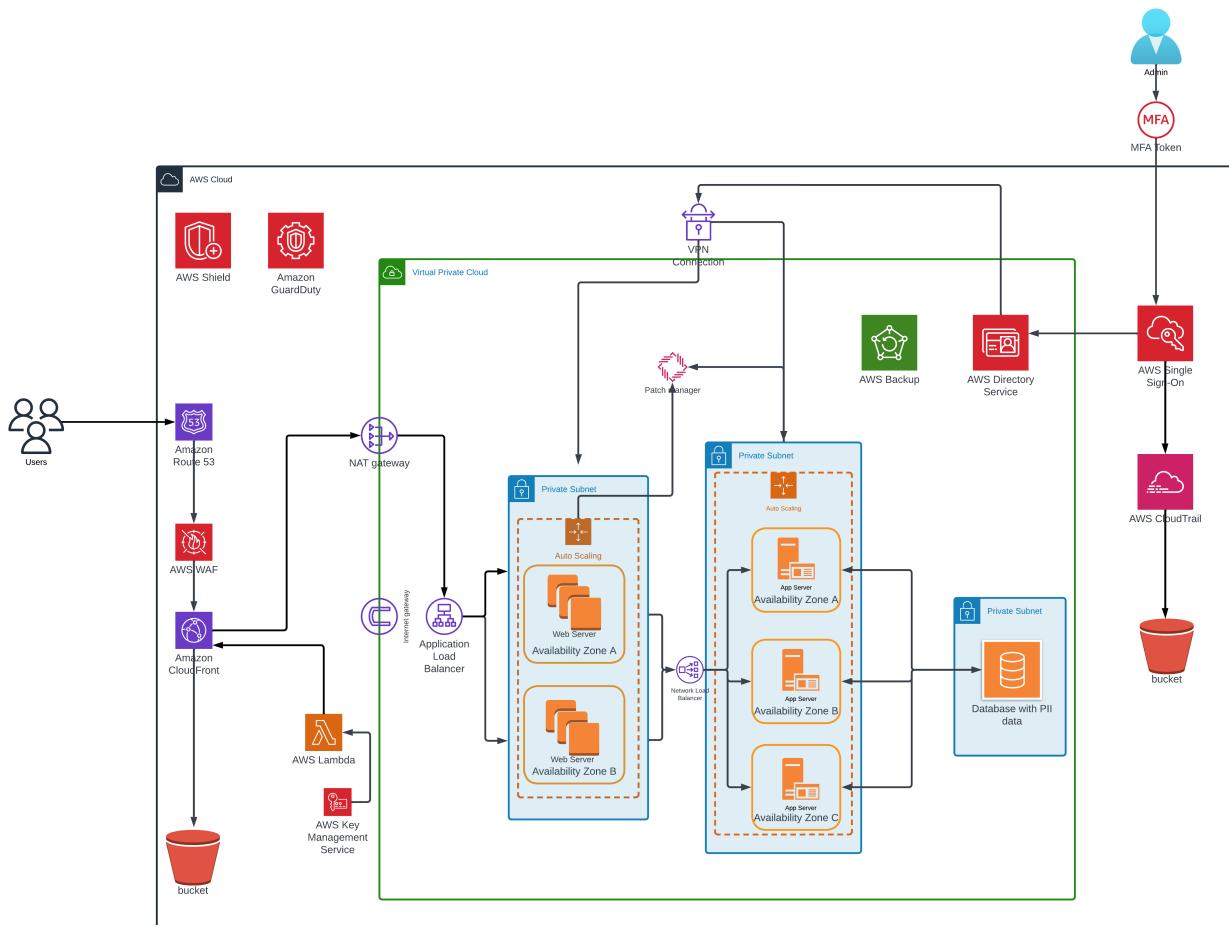
Current System →



Current Issues to Consider →

- Cobra Kai does not currently have a patching strategy
- Cobra Kai does not currently have a backup strategy
- Cobra Kai does not currently have an account permission strategy, every user has the ability to run privileged commands on the web server if they want to
- Their entire website infrastructure is highly vulnerable to DDoS, hardware failures, and human error. It runs in a closet for crying out loud
- The website has experienced DDoS attacks and compromise attempts they suspect comes from a rival dojo ran by Daniel LaRusso who with his deep pockets has become a persistent threat against Cobra Kai's IT operations
- Customers have complained about slow streaming, downloads, and order processing
- Cobra Kai's platform is processing credit card data and also stores customer PII (name, phone, email, address, and additional details about the customer)

Proposed Architecture



Technical Report

Configuring S3 bucket for image storage

- S3 is one of the building blocks of AWS, and is advertised as “infinitely scaling” storage
- In this scenario, we are using S3 to store the ova file which will be used to create the AMI (Amazon Machine Image), and consequently used to create the EC2 instances
- Following are some of the configurations of the S3 bucket →
 - Public access to the public has been blocked; this enables us to save the content of the bucket being compromised to any unauthorized user
 - Server Side Encryption has been enabled using the Amazon KMS (Amazon Key Management Service). This encrypts all the contents of the S3 bucket, when the data is at rest

Create bucket Info

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

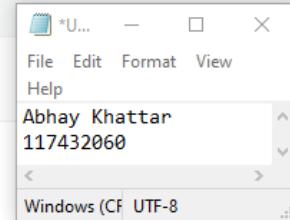
Bucket name

Bucket name must be globally unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

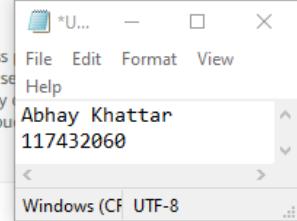
Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access points, and its access points. AWS recommends that you turn on Block all public access, but before applying any changes, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket, customize the individual settings below to suit your specific storage use cases. [Learn more](#)



Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

Block public access to buckets and objects granted through new public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

 Disable Enable

Default encryption

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

Disable
 Enable

Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

Amazon S3-managed keys (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#)

AWS KMS key

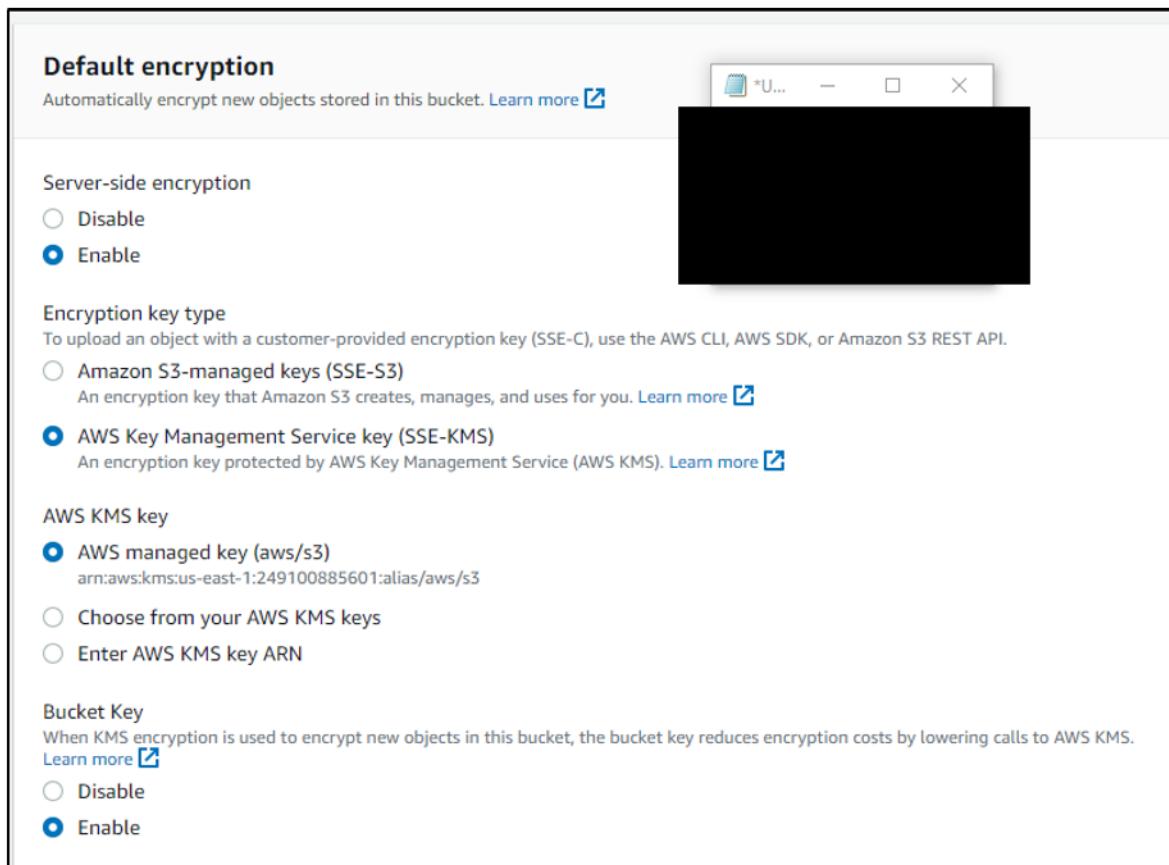
AWS managed key (aws/s3)
arn:aws:kms:us-east-1:249100885601:alias/aws/s3

Choose from your AWS KMS keys
 Enter AWS KMS key ARN

Bucket Key

When KMS encryption is used to encrypt new objects in this bucket, the bucket key reduces encryption costs by lowering calls to AWS KMS.
[Learn more](#)

Disable
 Enable



Upload the ova file to the S3 bucket

cobrakai-vm-117432060 [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

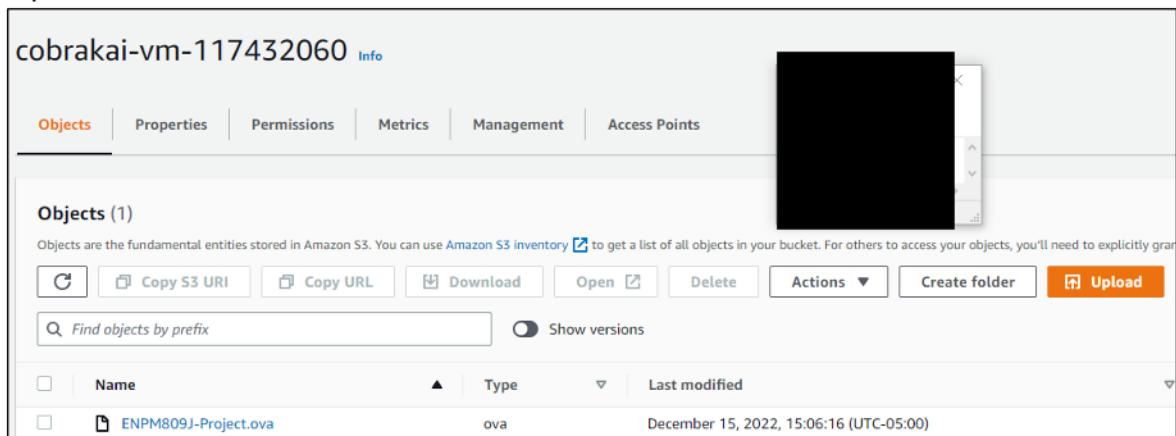
Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permission to do so.

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

[Find objects by prefix](#) Show versions

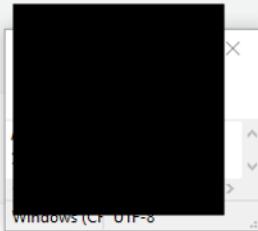
<input type="checkbox"/>	Name	Type	Last modified
<input type="checkbox"/>	ENPM809J-Project.ova	ova	December 15, 2022, 15:06:16 (UTC-05:00)



Creating AMI from ova file

To create the AMI from the image, we need to create the image using the ova file in the S3 bucket, and we can do this using the **EC2 Image Builder**

Import image



General

Enter a name and description for the base image that is being imported.

Name

Maximum of 128 characters. Letters, numbers, spaces, -, and _ are allowed

Version

Use the format: major.minor.patch

Description - optional

Custom description allowed. Maximum of 255 characters

We choose the base OS as *Ubuntu 18*. This version of ubuntu is stable and because ubuntu is open source, it has been thoroughly investigated by the community for the vulnerabilities

Base image operating system [Info](#)

Image Operating System (OS)
Image Builder supports Amazon Linux, Windows, Ubuntu, CentOS, RHEL, and SLES.

Amazon Linux
Amazon Linux 2


Windows
Windows Server 2012R2, 2016, 2019, 2004, 20H2, and 2022


Ubuntu
Ubuntu 18.04 LTS and 20.04 LTS


CentOS
CentOS 7 and 8


Red Hat Enterprise Linux (RHEL)
RHEL 7 and 8


SUSE Linux Enterprise Server (SLES)
SLES 12 and 15


OS version

Ubuntu 18

We now configure the VM configuration, by selecting the S3 bucket with the ova file as the source

VM import configuration

Specify the location of your import source, and optionally configure security, encryption, licensing, and other settings to transform your VM source into your base image.

Import source [Info](#)

Import disk container files (created when you export your VM from its virtual environment) as the source for your Image Builder image.

Disk container 1

Source: S3 bucket [Select S3 location of disk](#)

S3 bucket: s3://cobra-kai-vm-117432060/ENPM809J- [View](#) [Browse S3](#)

Description - optional: Enter description

Maximum of 255 characters.

[Add disk container](#)

IAM role [Info](#)

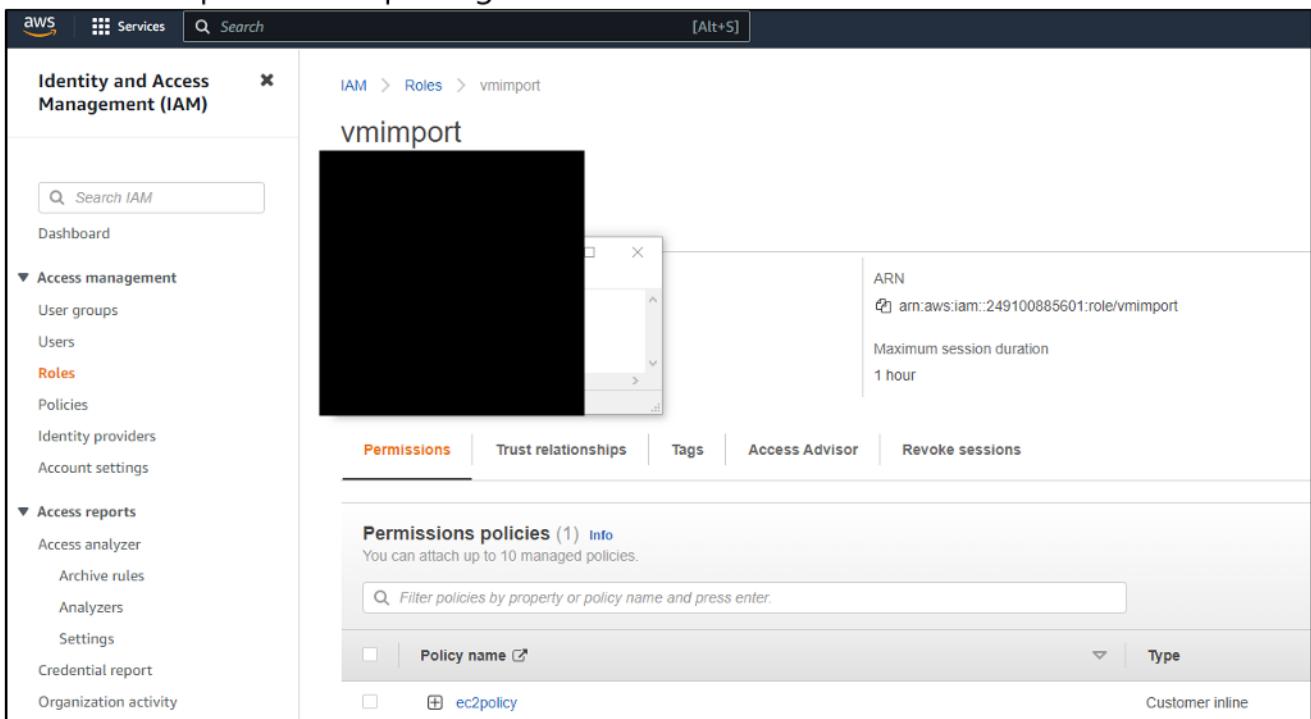
Specify an IAM role to use during the import process, or choose Create new role to create a new one.

IAM role: vmimport [Create new role](#)

Advanced settings - optional

You can define the Base image architecture, Encryption, and License configuration settings for this Image.

We have set the IAM role as `vmimport` [1] which is a new rule which has been defined keeping in mind the principle of least-privilege. This role only has access to the services which are required for importing the VM



The screenshot shows the AWS IAM Roles page. The left sidebar is titled "Identity and Access Management (IAM)" and includes sections for "Access management" (User groups, Users, Roles, Policies, Identity providers, Account settings), "Access reports" (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity), and a search bar. The main content area shows the "vmimport" role under "Roles". The role's ARN is listed as `arn:aws:iam::249100885601:role/vmimport`. The "Permissions" tab is selected, showing a single policy named "ec2policy" attached. Other tabs include "Trust relationships", "Tags", "Access Advisor", and "Revoke sessions".

Under the Permission Policies, we create a custom policy →

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3>ListBucket",
        "s3>GetBucketLocation"
      ],
    }
  ]
}
```

```

    "Resource": [
        "arn:aws:s3:::cobrakai-vm-117432060"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "s3:GetObject"
    ],
    "Resource": [
        "arn:aws:s3:::cobrakai-vm-117432060/*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:ModifySnapshotAttribute",
        "ec2:CopySnapshot",
        "ec2:RegisterImage",
        "ec2:Describe*"
    ],
    "Resource": "*"
}
]
}

```

We also need to edit the *Trusted Relationships*, and enter the following policy in the *Trusted Entities*

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "vmie.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "sts:ExternalId": "vmimport"
                }
            }
        }
    ]
}

```

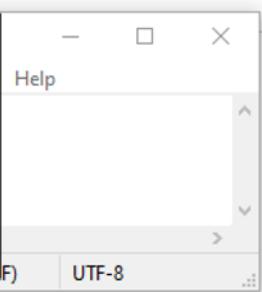
```
]  
}
```

IAM > Roles > vmimport

vmimport

IAM role for VM importing

Summary

ARN
 arn:aws:iam::249100885601:role/vmimport

Maximum session duration
1 hour

Permissions **Trust relationships** Tags Access Advisor Revoke sessions

Trusted entities

Entities that can assume this role under specified conditions.

```
1 [ {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Sid": "",  
6             "Effect": "Allow",  
7             "Principal": {  
8                 "Service": "vmie.amazonaws.com"  
9             },  
10            "Action": "sts:AssumeRole",  
11            "Condition": {  
12                "StringEquals": {  
13                    "sts:ExternalId": "vmimport"  
14                }  
15            }  
16        }  
17    ]  
18 }]
```

The image gets created, and can be used to create instances

Images (2)

This page lists Image Builder images created by you and your AWS accounts.

Find images by name

Image name	Type	Version	Image source	Platform
Cobra Kai Image 2	AMI	1.0.0	VMIE	Linux

Creating the Infrastructure

VPC

We start by creating the virtual private cloud where we will be placing all of our AWS infrastructures. This helps by creating a private network within the AWS Cloud to deploy our infrastructure.

The screenshot shows the 'Create VPC' wizard. In the 'VPC settings' section, 'VPC and more' is selected. The 'Preview' section displays a visual representation of the VPC structure:

- VPC:** Your AWS virtual network, labeled 'cobrakai-vpc'.
- Subnets (4):** Subnets within this VPC, divided into two availability zones:
 - us-east-1a:** Contains 'public-cobrakai-1' and 'private-cobrakai-1'.
 - us-east-1b:** Contains 'public-cobrakai-2' and 'private-cobrakai-2'.
- Route tables (3):** Route tables to resources, including 'Public route table without Name tag', 'Private route table without Name tag', and another 'Private route table without Name tag'.
- Network connections:** Connections to other networks, including 'cobrakai-ig' (Internet Gateway) and 'VPC endpoint without Name tag'.

The VPC contains a total of Four subnets, one Internet gateway, and route tables. We create one public and one private subnet each in two different availability zones (US-east-1a, and US-east-1b)

The following are the IPv4 CIDR for the subnets →

Name	IPv4 CIDR	Region
public-cobrakai-1	10.0.0.0/20	US-east-1a
private-cobrakai-1	10.0.16.0/20	US-east-1a
public-cobrakai-2	10.0.128.0/20	US-east-1b
private-cobrakai-2	10.0.144.0/20	US-east-1b

The screenshot shows the 'Subnets (4)' list view. The table displays the following information:

Check	Name	Subnet ID	IPv4 CIDR	Availability Zone
<input type="checkbox"/>	public-cobrakai-1	subnet-0919d9d8f111c6de6	10.0.0.0/20	us-east-1a
<input type="checkbox"/>	public-cobrakai-2	subnet-05ddf4f4468403ac6	10.0.16.0/20	us-east-1b
<input type="checkbox"/>	private-cobrakai-1	subnet-050384662611eb153	10.0.128.0/20	us-east-1a
<input type="checkbox"/>	private-cobrakai-2	subnet-001787ff04a70b343	10.0.144.0/20	us-east-1b

Route Tables →

Route tables (4) Info					
<input type="checkbox"/>	Name	▼	Route table ID	▼	Explicit subnet associat... Edge associati...
<input type="checkbox"/>	-		rtb-0deb3f8ae06c33c91	-	-
<input type="checkbox"/>	-		rtb-00634d9fc039b0eb2	4 subnets	-
<input type="checkbox"/>	-		rtb-09be99ae724fb1c67	-	-
<input type="checkbox"/>	-		rtb-06b6d0b4a6d3b84ac	-	-

Internet Gateway → Helps our instances connect to the internet

igw-0a0e5bdacb4e06621 / cobrakai-ig			
Details Info		View Help	
Internet gateway ID <input type="checkbox"/> igw-0a0e5bdacb4e06621	State  Attached		VPC ID vpc-0fb2022fb96aae5a cobrakai-vpc
Tags			
<input type="text"/> Search tags			
Key	Value		
Name	cobrakai-ig		

Elastic Load Balancer

We are using the Application Load Balancer (ALB) to help distribute the internet traffic to multiple servers (EC2 Instances). This will be the single point of access to the application (internet facing). The ALB works over Layer 7, as well as does regular health checks to the instances. This helps increase the availability of the application. We will integrate this with the *Auto Scaling Group*, and when the traffic increases the ASG would increase the number of instances, while the ALB will route the traffic automatically to help reduce the load on a single instance.

Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and AWS Lambda functions. You can route traffic to targets based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which target group to forward the request to. If no rule applies, or if none of the applicable rules specify a target group, the load balancer uses the default target group. If no default target group is specified, or if no targets are available in the default target group, the load balancer returns an error response. If no targets are available in any of the target groups specified in the rules, the load balancer returns an error response. If no rules are specified, the load balancer returns an error response.

► How Elastic Load balancing works



Basic configuration

Load balancer name

Name must be unique within your AWS account and cannot be changed after the load balancer is created.

Cobrakai-loadbalancer-lb1

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)

Scheme cannot be changed after the load balancer is created.

Internet-facing

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)

Select the type of IP addresses that your subnets use.

IPv4

Recommended for internal load balancers.

Dualstack

Includes IPv4 and IPv6 addresses.

Setting the network configurations for the ALB. By putting the load balancer in atleast two regions, we are able to increase the availability of the application

Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info

Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after you confirm the VPC for your targets, view your target groups [\[?\]](#).

cobrakai-vpc

vpc-0fb2022fb96aae5a

IPv4: 10.0.0.0/16



Mappings Info

Select at least two Availability Zones and one subnet per zone. The load balancer will automatically route traffic between the subnets in each zone. If no subnets are selected in an availability zone, the load balancer or the VPC are not available for selection.

us-east-1a (use1-az4)

Subnet

subnet-0919d9d8f111c6de6

public-cobrakai-1 ▾

IPv4 settings

Assigned by AWS

us-east-1b (use1-az6)

Subnet

subnet-05ddf4f4468403ac6

public-cobrakai-2 ▾

IPv4 settings

Assigned by AWS

Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups

Select up to 5 security groups

Create new security group [\[?\]](#)

Cobrakai-loadbalance-sg1 sg-0445f14766ca16372 X
VPC: vpc-0fb2022fb96aae5a



A **Security Group** acts as a **Firewall** to control the traffic to and from an EC2 instance. A SG has an *explicit Deny*, which means that if a rule does not allow for a certain traffic, it will automatically be denied.

Security Group for the Load Balancer → We allow traffic only on the HTTP port, and deny traffic to all the other ports

EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info
Cobrakai-loadbalance-sg1
Name cannot be edited after creation.

Description Info
Security Group for Load Balancer

VPC Info
Q_ vpc-0fb2022fb096aae5a

Inbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>
HTTP	TCP	80	Anywhere-IPv4	HTTP inbound to alb from internet 0.0.0.0/0

Add rule

Outbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Description - optional <small>Info</small>
HTTP	TCP	80	Custom	HTTP inbound to alb from internet sg-0be3aa4f5d8bf49bd

Security Group for the Web Server →

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info
cobrakai-webserver-sg1
Name cannot be edited after creation.

Description Info
Security Group for WebServers

VPC Info
Q_ vpc-0fb2022fb096aae5a

Inbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>
HTTP	TCP	80	Custom	HTTP inbound from alb on webserver sg-0445f14766ca16372

Add rule

Outbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Description - optional <small>Info</small>
HTTP	TCP	80	Custom	HTTP outbound to alb on webserver sg-0445f14766ca16372

In this application, the traffic is currently being sent as HTTP request, i.e. it is not being encrypted. This is a flaw in the application. We use the ALB to solve this vulnerability,

by having it list to port 443.

▶ Details
arn:aws:elasticloadbalancing:us-east-1:249100885601:loadbalancer/app/Cobrakai-loadbalancer-lb1/47479accf953f43d

Listener details
A listener is a process that checks for connection requests using the specified protocol and port, and then determine how the load balancer routes requests to its registered targets.

Protocol	Port
HTTPS ▾	: 443 1-65535

Default actions [Info](#)
Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

▼ 1. Forward to [Info](#) [Remove](#)

Target group	Weight (0-999)
cobrakai-tg1 Target type: Instance, IPv4	HTTP ▾ 1 Traffic distribution: 100%
Select a target group	0

[Create target group](#)

[Enable group-level stickiness](#) [Info](#)
If you enable stickiness for your target group, requests routed to it remain in the same group for the duration you specify.

Add action ▾

Secure listener settings [Info](#)

Security policy
Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections with clients.

ELBSecurityPolicy-2016-08

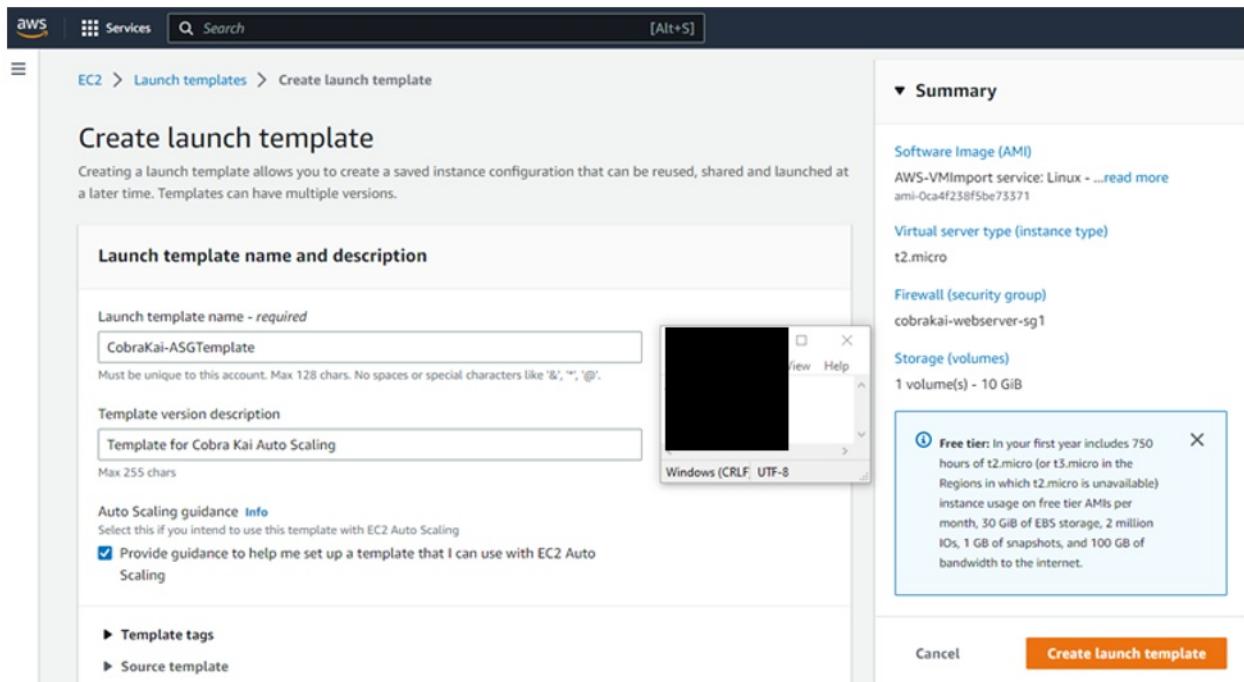
[Compare security policies](#)

Auto Scaling Groups

We need to create a Launch template to create the auto scaling group. We set the template with the following configurations →

- Firewall - Specify the security groups, so that only the ports which are required are open

- Health Check of the instances every 300 seconds. We can use the health dashboard to make sure the required number of instances are running
- We can configure the size for the group size and scaling policies. We have set them as the following -
 - Desired Capacity - 2
 - Minimum Capacity - 1
 - Maximum Capacity - 3
- These can be adjusted according to the traffic that might be expected
- We also set a notification system, which will notify the subscribers when an event occurs



▼ Network settings [Info](#)

Subnet [Info](#)

Don't include in launch template

 [Create new subnet](#) 

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

[Select existing security group](#)

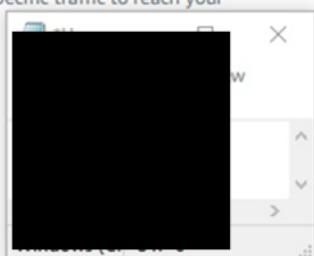
[Create security group](#)

Common security groups [Info](#)

Select security groups

cobrakai-webserver-sg1 sg-06e3aa4f5386f49bd 
VPC: vpc-0fb2022fb96aae5a

CobraKai-SSHAcess sg-02683a101efa62287 
VPC: vpc-0fb2022fb96aae5a



 [Compare security group rules](#)

[Hide all selected](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

► [Advanced network configuration](#)

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer

Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer

Choose from your existing load balancers.

Attach to a new load balancer

Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups

This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups

Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

cobrakai-tg1 | HTTP

Application Load Balancer: Cobrakai-loadbalancer-lb1



Health checks - optional

Health check type Info

EC2 Auto Scaling automatically replaces instances that fail health checks. If you enabled load balancing, you can enable ELB health checks in addition to the EC2 health checks that are always enabled.

EC2 ELB

Health check grace period

The amount of time until EC2 Auto Scaling performs the first health check on new instances after they are put into service.

300 seconds

Configure group size and scaling policies Info

Set the desired, minimum, and maximum capacity of your Auto Scaling group. You can optionally add a scaling policy to dynamically scale the number of instances in the group.

Group size - optional Info

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.

Desired capacity

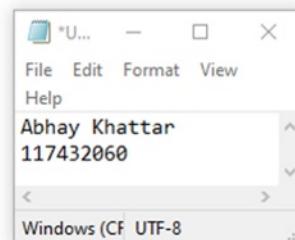
2

Minimum capacity

1

Maximum capacity

3



▼ Notification 1

Send a notification to

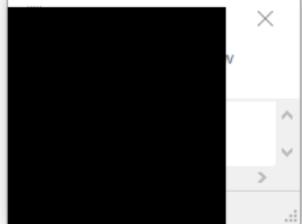
With these recipients

[Use existing topic](#)

Event types

Notify subscribers whenever instances

- Launch
- Terminate
- Fail to launch
- Fail to terminate



Choose launch template or configuration [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name

Auto Scaling group name
Enter a name to identify the group.

Must be unique to this account in the current Region and no more than 255 characters.

Launch template [Info](#) [Switch to launch configuration](#)

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

CobraKai-ASGTemplate

Create a launch template [Edit](#) [Create a launch template version](#)

Version
Default (1) [Edit](#)

Description	Launch template	Instance type
Template for Cobra Kai Auto Scaling	CobraKai-ASGTemplate Edit lt-0d0f257cc910a8910	t2.micro
AMI ID	Security groups	Request Spot Instances
ami-0ca4f238f5be73371	-	No
Key pair name	Security group IDs	
CloudSecurityFinal	-	

Setting up a Database

For the database, we are using the **Amazon Aurora**, which is a proprietary technology from AWS. Amazon Aurora^[2] is “cloud optimized” and claims 5x performance improvement over MySQL on RDS, and over 3x performance of the Postgres on RDS. The size for the database increases automatically in increments on 10GB. Although it costs more than RDS but is more efficient.

Note → Aurora is not present in the free tier, so I did not deploy it in my environment, but I have added screenshots for all the steps

Create database

Choose a database creation method Info

Standard create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type Info

Amazon Aurora



MySQL



MariaDB



PostgreSQL



Oracle



Microsoft SQL Server



Instance configuration

The DB instance configuration options below are limited to those supported by Aurora.

DB instance class [Info](#)

- Memory optimized classes (includes r classes)
- Burstable classes (includes t classes)



- Include previous generation classes

Availability & durability

Multi-AZ deployment [Info](#)

- Create an Aurora Replica or Reader node in a different AZ (recommended for scaled availability)
Creates an Aurora Replica for fast failover and high availability.
- Don't create an Aurora Replica

Aurora has the capability of creating replicas in different AZ to increase the availability of the database. We also enable encryption using the AWS KMS, to encrypt the data at rest. Aurora has in-built capability of auto scaling and multi region availability can be enabled as a functionality

Encryption

Enable encryption

Choose to encrypt the given instance. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service console. [Info](#)

AWS KMS key [Info](#)

(default) aws/rds ▾

Account

249100885601

KMS key ID

alias/aws/rds

Backtrack

Backtrack lets you quickly rewind the DB cluster to a specific point in time, without having to create another DB cluster. [Info](#)

Enable Backtrack

Enabling Backtrack will charge you for storing the changes you make for backtracking.

Log exports

Select the log types to publish to Amazon CloudWatch Logs

- Audit log
- Error log
- General log
- Slow query log

We are collecting all the logs and exporting it to the Amazon CloudWatch.

IAM

Note → It is better to use AWS Organization to manage the users, resources etc. for the organization. I was not able to implement it for my account as, it required buying a domain.

In accordance with the principles of Identity and access management, we set up different user groups which will help us give access to the users. We can add the users to their respective user groups, and give them the permissions accordingly. If we have multiple people who require the same access, instead of adding permissions individually, we can club them in a group and give all the users, the access that they need. We have the following three user groups -

1. CobraKai-Administrators → They maintain the entire application, and have access similar to that of the root
2. CobraKai-Developers → They develop and test new utilities, before adding them to the main application

3. CobraKai-Guest

These user groups have permissions in accordance with the **Principle of Least Privilege**, i.e. the minimum permissions which are required to complete the task

User groups (4) [Info](#)
A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

Filter User groups by property or group name and press enter

<input type="checkbox"/>	Group name	Users
<input type="checkbox"/>	admin_udemy	[REDACTED]
<input type="checkbox"/>	CobraKai-Guest	[REDACTED]
<input type="checkbox"/>	CobraKai-Developers	[REDACTED]
<input type="checkbox"/>	CobraKai-Administrators	[REDACTED]

Some other security measures which have been put in place are the following →

4. Password Policy → We have set up a custom password policy, which can help maintain the security standard of the organization

Edit password policy

Password policy

IAM default
Default password requirements for IAM users.

Custom
Use a customized password policy.

Password minimum length.
Enforce a minimum length of characters.
14 characters
Needs to be between 6 and 128.

Password strength

Require at least one uppercase letter from the Latin alphabet (A-Z)
 Require at least one lowercase letter from the Latin alphabet (a-z)
 Require at least one number
 Require at least one non-alphanumeric character (! @ # \$ % ^ & * () _ + - = [] { } | ')

Other requirements

Turn on password expiration
Expire password in 60 day(s)
Needs to be between 1 and 1095 days.

Password expiration requires administrator reset

Allow users to change their own password

Prevent password reuse
Remember 3 password(s)
Needs to be between 1 and 24.

5. MultiFactor Authentication → We have enforced MultiFactor authentication. This helps by adding an extra layer of security when the user tries to login

Configure multi-factor authentication

Choose how often workforce users are prompted for multi-factor authentication (MFA) and which types of devices they can use to sign in to the AWS access portal. [Learn more](#)

MFA Settings

Prompt users for MFA

Only when their sign-in context changes (recommended)
Users with a registered MFA device are only prompted if they sign in from a new device or browser, or from an unknown IP address. Users can register a new MFA device.

Every time they sign in (always-on)
Users with a registered MFA device are prompted every time they sign in.

Never (disabled)
All users sign in with their standard user name and password only. Choosing this option disables MFA.

Users can authenticate with these MFA types

Security keys and built-in authenticators
Users can verify their identity by using any FIDO2 or U2F capable device such as an external physical security key (for example, YubiKey or Feitian devices) or a built-in authenticator (for example, Apple TouchID or Windows Hello).

Authenticator apps
Users can verify their identity by entering a code generated from a time-based one-time password authenticator app (for example, Authy, Google Authenticator, Microsoft Authenticator).

If a user does not yet have a registered MFA device

Require them to register an MFA device at sign in

Require them to provide a one-time password sent by email to sign in

Block their sign-in

Allow them to sign in

Who can manage MFA devices

Users can add and manage their own MFA devices

[Cancel](#) [Save changes](#)

6. User access keys for the root user is deactivated, and the root console access is also disabled
7. Credentials Reports can be a good way to check the status of credentials for the users in the organization

IAM > Credential Report

Credentials report of IAM users in this account [Info](#)

The credentials report lists all your IAM users in this account and the status of their various credentials. After a report is created, it is stored for up to four hours.

[Download credentials report](#)

No report created in the past 4 hours. A new report will be created.

[Create report](#)

Creating a Backup Plan

We use AWS Backup to create a backup plan according to our needs. We make use of tags to identify the resources that we need to backup. We can customize the backup plan according to our need and according to the compliance requirement (if any). In this scenario, we create a plan to backup all the resources in the account which will be retained for 1 year.

Create backup plan Info

Start options

Backup plan options Info

Start with a template

Create a Backup plan based on a template provided by AWS Backup.

Build a new plan

Configure a new Backup plan from scratch.

Define a plan using JSON

Modify the JSON expression of an existing backup plan or create a new expression.

Templates

Choose a template plan with existing rules.

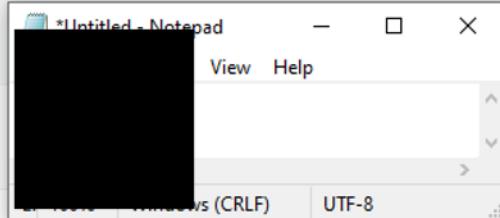
Daily-Monthly-1yr-Retention

Backup plan name

CobraKai

Backup plan name is case sensitive. Must contain from 1 to 50 alphanumeric or '-' characters.

Edit Backup rule



Backup rule configuration Info

Backup rule name

DailyBackups

Backup rule name is case sensitive. Must contain from 1 to 50 alphanumeric or '-' characters.

Backup vault Info

Default

[Create new Backup vault](#)

Backup frequency Info

Daily

Choose an IAM role

Role name

AWSServiceRoleForBackup

Resource selection Info

Assign resources to this Backup plan using tags and resource IDs.

1. Define resource selection Info

Protect all resources or specify resources by type or ID.

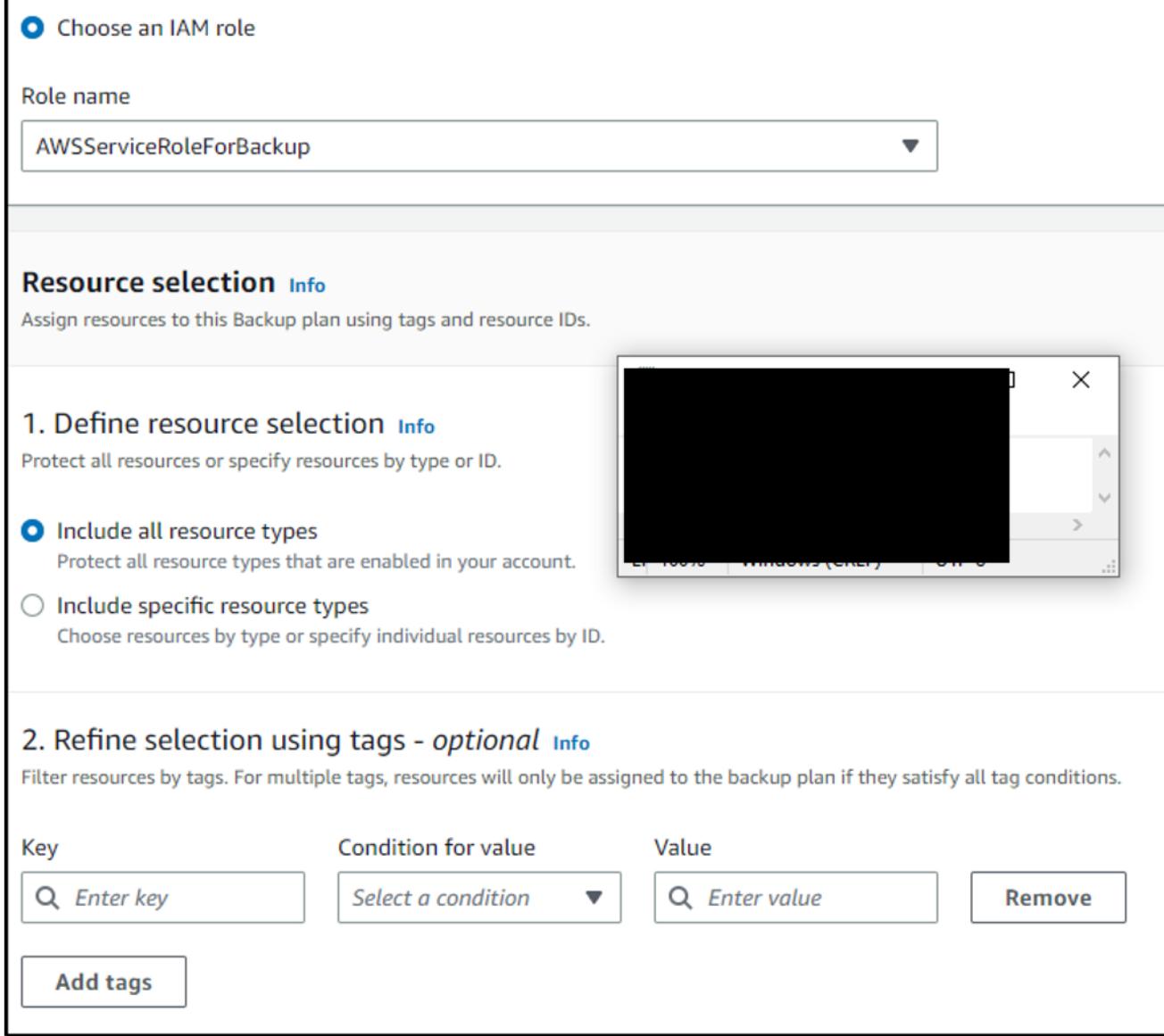
Include all resource types
Protect all resource types that are enabled in your account.

Include specific resource types
Choose resources by type or specify individual resources by ID.

2. Refine selection using tags - optional Info

Filter resources by tags. For multiple tags, resources will only be assigned to the backup plan if they satisfy all tag conditions.

Key	Condition for value	Value	Remove
<input type="text"/> <small>Enter key</small>	<input type="button" value="Select a condition"/> <small>▼</small>	<input type="text"/> <small>Enter value</small>	<input type="button" value="Remove"/>



Other Services

CloudTrail

CloudTrail can be used to provide governance, compliance and audit for the AWS account. CloudTrail is enabled by default, and we have all the logs saved to an S3 bucket. We cannot edit or change the logs, and hence helps with governance and compliance during the audits. These logs also play a crucial role during the investigation of an incident.

CloudTrail > Quick trail create

Quick trail create

Trail details

Start logging management events by creating a trail with simplified settings. Logs are sent to an S3 bucket we create on your behalf. To choose a different bucket or additional events, go to the full [Create trail](#) workflow.

A trail created in the console is a multi-region trail. [Learn more](#)

Trail name
Enter a display name for your trail.

3-128 characters. Only letters, numbers, periods, underscores, and dashes are allowed.

Trail log bucket and folder

Logs will be stored in aws-cloudtrail-logs-249100885601-af1304dc/AWSLogs/249100885601

Info Though there is no cost to log these events, you incur charges for the S3 bucket that we create to store your logs.

[Cancel](#) [Create trail](#)

CloudTrail > Trails

Trails

Name	Home region	Multi-region trail	Insights	Organization trail	S3 bucket
management-events	US East (N. Virginia)	Yes	Disabled	No	aws-cloudtrail-logs-249100885601-af1304dc

Buckets (3) [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

Name	AWS Region	Access
*Untitled - Notepad	US East (N. Virginia) us-east-1	Bucket and objects not public
aws-cloudtrail-logs-249100885601-af1304dc	US East (N. Virginia) us-east-1	Bucket and objects not public
cobrakai-vm-117432060	US East (N. Virginia) us-east-1	Bucket and objects not public

Web Application Firewall and Amazon Shield

The standard AWS Shield is enabled by default and protects against DDOS attacks on the website and application.

The AWS WAF protects web application from common web exploits on layer 7 (HTTP). We enable the WAF on the Application Load Balancer because ALB is internet facing.

We enabled the free services which include rules such as admin protection, or protection against known bad IP address or malicious inputs

AWS WAF > Web ACLs > Create web ACL

Step 1
Describe web ACL and associate it to AWS resources

Step 2
Add rules and rule groups

Step 3
Set rule priority

Step 4
Configure metrics

Step 5
Review and create web ACL

Describe web ACL and associate it to AWS resource

Web ACL details

Name
 The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Description - optional
 The description can have 1-256 characters.

CloudWatch metric name
 The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Resource type
Choose the type of resource to associate with this web ACL.
 CloudFront distributions
 Regional resources (Application Load Balancer, API Gateway, AWS AppSync, Amazon Cognito User Pools)

Region
Choose the AWS region to create this web ACL in.

Add AWS resources

X

Resource type

Select the resource type and then select the resource you want to associate with this web ACL.

Amazon API Gateway

Application Load Balancer

Amazon Cognito User Pools

 Find AWS resources to associate

< 1 >

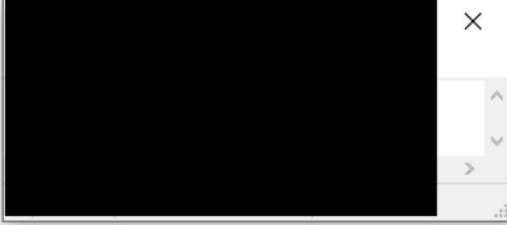


Name

CobraKaiALB

Cobrakai-loadbalancer-lb1

Free rule groups

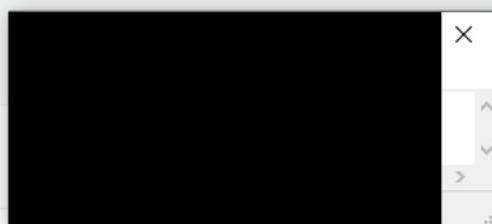
Name	Capacity	Additional fees	Action
Admin protection Contains rules that allow you to block external access to exposed admin pages. This may be useful if you are running third-party software or would like to reduce the risk of a malicious actor gaining administrative access to your application.	100		<input checked="" type="checkbox"/> Add to web ACL <button>Edit</button>
Amazon IP reputation list This group contains rules that are based on Amazon threat intelligence. This is useful if you would like to block sources associated with bots or other threats.			<input checked="" type="checkbox"/> Add to web ACL <button>Edit</button>
Anonymous IP list This group contains rules that allow you to block requests from services that allow obfuscation of viewer identity. This can include request originating from VPN, proxies, Tor nodes, and hosting providers. This is useful if you want to filter out viewers that may be trying to hide their identity from your application.	50		<input checked="" type="checkbox"/> Add to web ACL <button>Edit</button>
Core rule set Contains rules that are generally applicable to web applications. This provides protection against exploitation of a wide range of vulnerabilities, including those described in OWASP publications.	700		<input checked="" type="checkbox"/> Add to web ACL <button>Edit</button>
Known bad inputs Contains rules that allow you to block request patterns that are known to be invalid and are associated with exploitation or discovery of vulnerabilities. This can help reduce the risk of a malicious actor discovering a vulnerable application.	200		<input checked="" type="checkbox"/> Add to web ACL <button>Edit</button>
Linux operating system Contains rules that block request patterns associated with exploitation of			<input checked="" type="checkbox"/> Add to web ACL

CloudFront

As cobrakai would be required to supply videos on demand, we can use the AWS CloudFront which is the content delivery functionality of AWS. "CloudFront is a web service which speeds up the distribution of the web content to the users. CloudFront helps with an increased reliability and availability as the copies of the files are already cached in multiple edge locations around the world, and hence it is served instantly."^[3] We can add the SSL certificates for secure connections, add the WAF to cloudfront to protect it from common web exploits.

Create distribution

Origin



Origin domain

Choose an AWS origin, or enter your origin's domain name.

 cobrakai-vm-117432060.s3.us-east-1.amazonaws.com X

Origin path - optional Info

Enter a URL path to append to the origin domain name for origin requests.

 Enter the origin path

Name

Enter a name for this origin.

 cobrakai-vm-117432060.s3.us-east-1.amazonaws.com

Origin access Info

Public

Bucket must allow public access.

Origin access control settings (recommended)

Bucket can restrict access to only CloudFront.

Legacy access identities

Use a CloudFront origin access identity (OAI) to access the S3 bucket.

Compress objects automatically Info

No

Yes

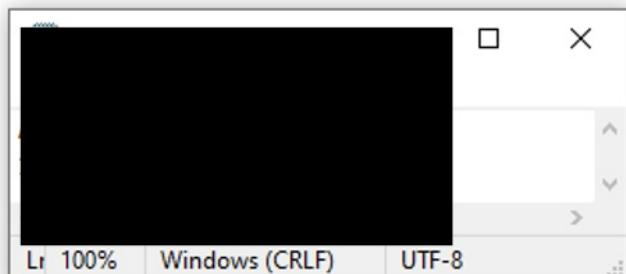
Viewer

Viewer protocol policy

HTTP and HTTPS

Redirect HTTP to HTTPS

HTTPS only



Allowed HTTP methods

GET, HEAD

GET, HEAD, OPTIONS

GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

Restrict viewer access

If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.

No

Yes

Cache key and origin requests

We recommend using a cache policy and origin request policy to control the cache key and origin requests.

- Cache policy and origin request policy (recommended)
- Legacy cache settings

Cache policy

Choose an existing cache policy or create a new one.

CachingOptimized

Recommended for S3 origins ▾

Default policy when CF compression is enabled



[Create policy](#) [View policy](#)

Origin request policy - optional

Choose an existing origin request policy or create a new one.

Select origin policy ▾

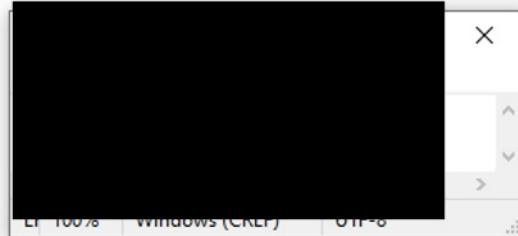


[Create policy](#)

Price class | [Info](#)

Choose the price class associated with the maximum price that you want to pay.

- Use all edge locations (best performance)
- Use only North America and Europe
- Use North America, Europe, Asia, Middle East, and Africa



AWS WAF web ACL - optional

Choose the web ACL in AWS WAF to associate with this distribution.

Choose web ACL ▾

Alternate domain name (CNAME) - optional

Add the custom domain names that you use in URLs for the files served by this distribution.

[Add item](#)

To add a list of alternative domain names, use the [bulk editor](#).

Custom SSL certificate - optional

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

Choose certificate ▾



[Request certificate](#)

Supported HTTP versions

Add support for additional HTTP versions. HTTP/1.0 and HTTP/1.1 are supported by default.

HTTP/2

HTTP/3

AWS Key Management Service (KMS)

KMS is an AWS product which helps with the management of the encryption keys. Using these keys we are able to encrypt the backups, S3, the databases, etc.

Step 1
Configure key

Step 2
Add labels

Step 3
Define key administrative permissions

Add labels

Alias

You can change the alias at any time.

Alias

While setting KMS for the first time, we will have to identify the user who would have key administrative permissions. Here we are assigning those permissions to Hawk

Define key administrative permissions

Key administrators

Choose the users or groups that will administer this key through the KMS API. You may need to add additional users or groups.



Type

<input type="checkbox"/>	abhaykhattar	/	User
<input type="checkbox"/>	a robinson	/	User
<input type="checkbox"/>	AzureADRoleManager	/	User
<input checked="" type="checkbox"/>	hmoskowitz	/	User

KMS > Customer managed keys > Create key

Step 1
Configure key

Step 2
Add labels

Step 3
Define key administrative permissions

Step 4
Define key usage permissions

Step 5
Review

Key configuration

Key type Symmetric	Key spec SYMMETRIC_DEFAULT	Key usage Encrypt and decrypt
Origin AWS KMS	Regionality Single-Region key	

You can edit the key configuration after the key is created.

Untitled - Notepad

Aliases	Key ID	Status	Key spec	Key usage
cobrakai-s3	637a1923-66ac-482b-bf86-3375d3dde9f5	Enabled	SYMMETRIC_DEFAULT	Encrypt and decrypt

AWS System Manager

We use the AWS System Manager to create a Patching Schedule. It is essential to have a patching strategy so that we are able to keep the systems up-to-date with the latest patches and in compliance with the security standards.

For the purpose of patching, we identify the machines using tags. According to the current configuration, we are patching machines which are in an autoscaling group, and have the group name as **CobrKai-ASG** on every Tuesday at 12 noon.

- Enter instance tags
- Select a patch group
- Select instances manually

Instance tags

Specify one or more instance tag key/value pairs to identify the instances you want to patch.

Tag key	Tag value (optional)	Add
---------	----------------------	-----

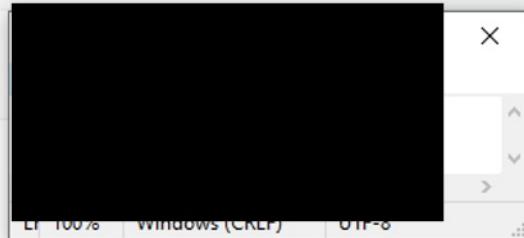
Enter a tag key and optional value applied to the instances you want to target, and then choose Add.

aws:autoscaling:groupName : CobraKai-ASG X

Patching schedule

How do you want to specify a patching schedule?

- Select an existing Maintenance Window
- Schedule in a new Maintenance Window
- Skip scheduling and patch instances now



How do you want to specify a Maintenance Window schedule?

- Use a CRON schedule builder
- Use rate schedule builder
- Enter a CRON/Rate expression

Maintenance Window run frequency

Every 12 hours

Every Tuesday at 12:00

Maintenance Window duration

Maximum number of hours to allow a Maintenance Window to run.

1

Enter a number between 1 and 24

Maintenance Window name

cobrakai-patching-window

We set the baseline as the AWS-UbuntuDefaultPatchBaseline which is provided by AWS (our base operating system is Ubuntu)

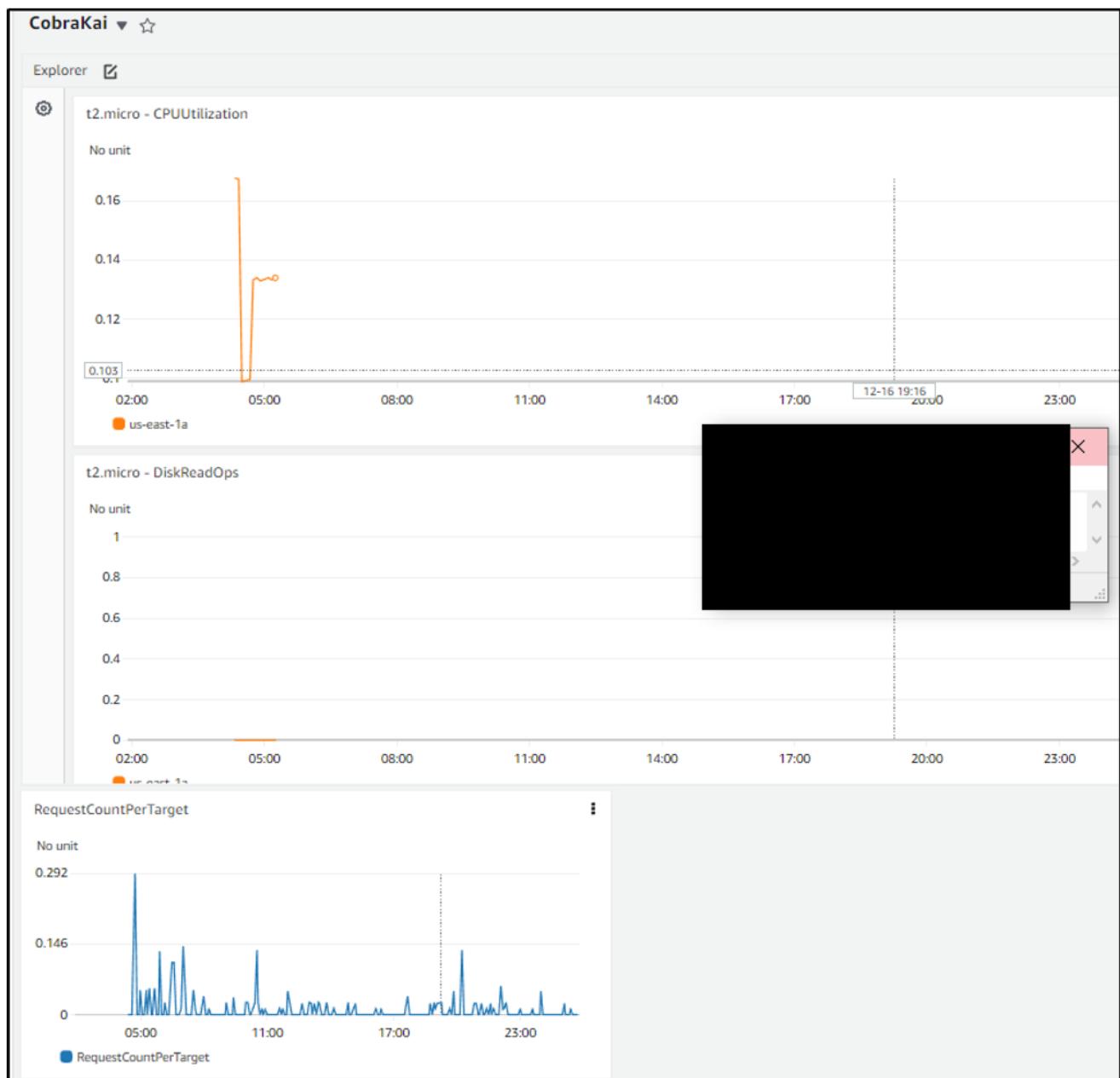
Patch Manager

Dashboard	Compliance reporting	Patch baselines	Patches	Patch groups	Settings
Patch baselines (15)					
<input type="text"/> Filter patch baselines					
		Baseline name = AWS-UbuntuDefaultPatchBaseline X		View details	Edit
		Clear filter			
Baseline ID	Baseline name	Description	Operating sys		
<input type="radio"/> pb-0c7e89f711c3095f4	AWS-UbuntuDefaultPatchBaseline	Default Patch Baseline for Ubuntu Provided by AWS.	Ubuntu		

CloudWatch

We can use CloudWatch to create dashboards and monitor all the resources in AWS. CloudWatch collects and visualizes real-time logs, metrics, and event data in automated dashboards to streamline your infrastructure and application maintenance.

[4]



CloudWatch makes it easier to monitor the resources by using customizable dashboards, and alerts.

AWS Guard Duty

GuardDuty monitors the AWS resources (AWS accounts, instance, storage, database, etc.) for potential threats. GuardDuty uses Machine Learning algorithms for threat

detection and is also capable of threats by initiating automated response.^[5]

The screenshot shows the AWS GuardDuty console. On the left, a sidebar menu includes 'Findings' (selected), 'Usage', 'Malware scans', 'Settings' (with 'Lists', 'S3 Protection', 'EKS Protection', 'Malware Protection', 'RDS Protection' (Preview)), and 'Accounts'. Below these are 'What's New' and 'Partners'. The main content area is titled 'Findings' and shows a message: 'You don't have any findings. GuardDuty continuously monitors your AW...'.

Global Accelerator

Global Accelerator is used to increase the speed for upload and download of resources to and from the cloud. It uses the end points and amazon's own network for the transfer of data

The screenshot shows the AWS Global Accelerator console. The top navigation bar shows 'AWS Global Accelerator > Accelerators'. The main content area is titled 'Accelerators (1)'. It displays a table with one row of data:

Name	Type	Last Refreshed	Status
cobrakai-accelerator	Standard	15.197.224.220 3.33.237.21	On

PCI DSS Compliance

Security has to be integrated at every layer/step in order to be PCI DSS compliant. AWS Architect provides a list of the best practices to be used while configuring services in

AWS in order to be PCI DSS compliant. The following are steps which we have integrated as part of security in the architecture →

- Creation of **Virtual Private Cloud** ensures that the resources are isolated
 - Using **Security Groups and Network Access Control List (NACL)** to create rules and allow/deny the traffic and access
 - **Encryption of Data** has been at every step
 - Enforcing strong Identity Access and Management
 - Enforcing a strong password policy, to mitigate password brute-force and password reuse
 - Using a Role based approach, and implementing the **principle of least privilege**
 - Enforcing MultiFactor authentication
 - Weekly patching for all the systems, so that the systems are up-to-date with the latest patches against vulnerabilities
 - Using CloudWatch to monitor the resources
 - Using CloudTrail to gather logs and event information which can be used for auditing and incident response
 - Using AWS GuardDuty to identify and respond to potential threats
-

References

1. <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-export-vm-using-import-export/>
2. <https://aws.amazon.com/rds/aurora/>
3. <https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Introduction.html>
4. <https://aws.amazon.com/cloudwatch/>
5. <https://aws.amazon.com/guardduty/>