

PPT Notes - AWS DevOps Engineer Professional

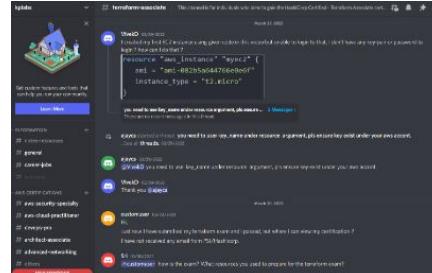


Our Community

You can join our **Discord community** for any queries / discussions. You can also connect with other students going through the same course in Discord (Optional)

Discord Link: <http://kplabs.in/chat>

Group: #devops-pro



PPT Version

PPT Release Date = 21st June 2023

We regularly release new version of PPT when we update this course.

Please check regularly that you are using the latest version.

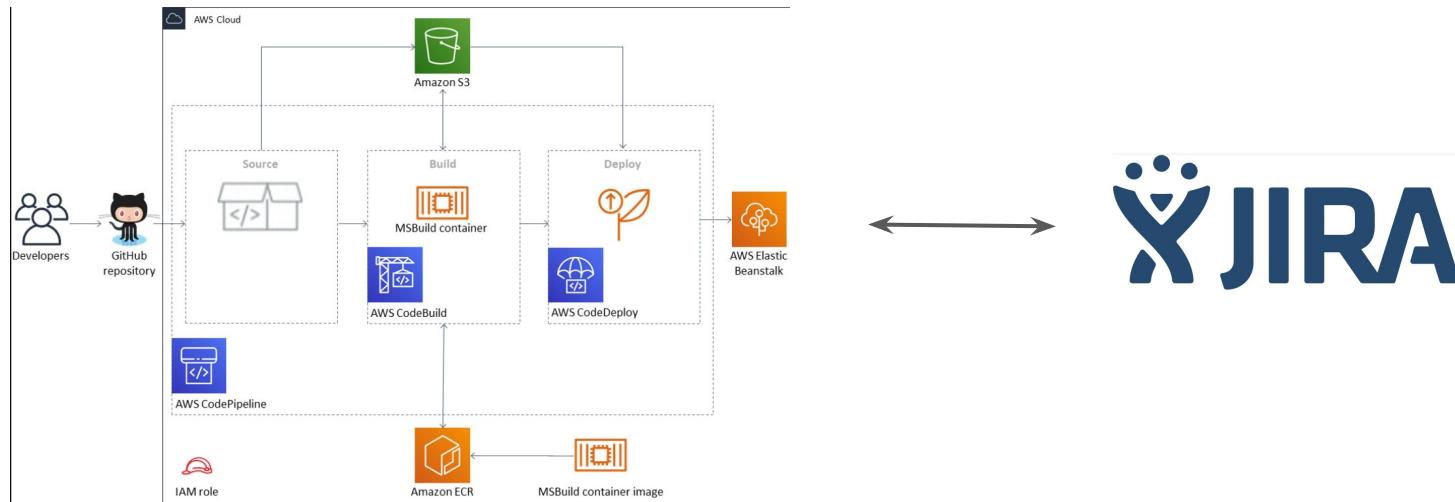
The Latest Version Details are mentioned in the PPT Lecture in Section 1.

AWS CodeStar

It's just awesome!

Use-Case: Building CI/CD Pipeline

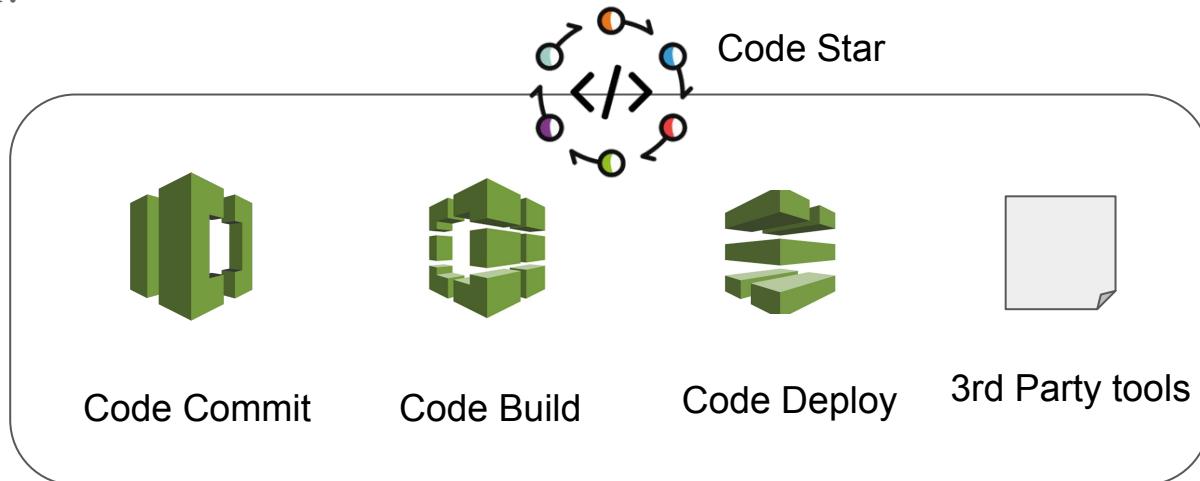
To build CI/CD pipeline, we need to configure and integrate many services like AWS CodeCommit, CodeDeploy, CodePipeline and others.



Overview of Code Star

AWS Code Star provides a unified interface to quickly develop, build and deploy application on AWS.

It allows us to launch entire continuous delivery toolchain in minutes, allowing releasing code faster.



AWS CodeCommit

Git Repository as Service

Overview of AWS CodeCommit

AWS CodeCommit is a managed source control service provided by AWS for hosting GIT repos.

The screenshot shows the AWS CodeCommit interface for comparing commits. The top navigation bar includes 'Developer Tools > CodeCommit > Repositories > MyDemoRepo > Compare'. The main title is 'MyDemoRepo'. Below it are tabs: 'Commits' (selected), 'Commit visualizer', and 'Compare commits'. Underneath, there are dropdown menus for 'Destination' (set to 'AnotherBranch') and 'Source' (set to '6b65eb76'). A large orange 'Compare' button is next to the source dropdown. To the right are 'Cancel', 'Hide whitespace changes' (radio button), 'Unified' (radio button selected), and 'Split' (radio button). Below these are buttons for 'Page 1 of 1' and 'Go to file'. The main content area displays a diff for the file 'ahs_count.py'. The diff shows code changes between the destination branch and the source commit. Lines 8 and 9 are highlighted in red, while line 10 is highlighted in green. The code snippet includes comments about printing results to standard output. At the bottom, another file 'anothernew/dir2/anotheritest.txt' is listed as 'Added'.

```
*** *** @@ -5,6 +5,6 @@
 5   5
 6   6     total = (ess + z)
 7   7     ahs = "Number of alveolar hissing sibilants: {}"
 8 - print(ahs.format(total))
 9 + print(alv.format(total))
10 10   #when using this script, make sure that you ask the subject to use one of the provided texts, such as bumblebee.txt.
```

anothernew/dir2/anotheritest.txt Added

Repository Tags

Repositories can be tagged in AWS CodeCommit which further helps to identify and organize your AWS resources.

The screenshot shows the AWS CodeCommit repository settings interface. The navigation bar at the top includes links for Developer Tools, CodeCommit, Repositories, demo-code-commit, and Settings. Below the navigation is the repository name, "demo-code-commit". A horizontal menu bar contains tabs for General, Notifications, Triggers, Repository tags (which is highlighted in orange), and Amazon CodeGuru Reviewer. The main content area is titled "Repository tags" and includes a "Info" link. A descriptive text explains that a tag is a label assigned to an AWS resource, consisting of a key and an optional value, used to help manage resources. Below this text is a table with two columns: "Key" and "Value". A single tag is listed: "Team" in the Key column and "Payments" in the Value column.

Key	Value
Team	Payments

Identity Policies Based On Tags

You can create a policy that allows or denies actions on repositories based on the AWS tags associated with those repositories.

Developer Tools > CodeCommit > Repositories > demo-code-commit > Settings

demo-code-commit

General | Notifications | Triggers | **Repository tags** | Amazon CodeGuru Reviewer

Repository tags Info

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help ma

Key	Value
Team	Payments



```
{  
    "Version": "2012-10-17",  
    "Statement" : [  
        {  
            "Effect" : "Allow",  
            "Action" : "codecommit:*"  
            "Resource" : "*",  
            "Condition" : {  
                "StringEquals" : "aws:ResourceTag/Team": "Payments"  
            }  
        }  
    ]  
}
```

Identity Policies For AWS CodeCommit

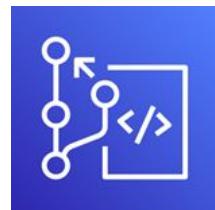
IAM Policy allows administrators to put a granular level of control over AWS CodeCommit service.

Sample Use-Cases:

- Deny all access to a repository with tag Status as Secret.
- Allow access to repository only in Mumbai region.
- Allow user only connecting from 10.77.2.50 to connect to repository.
- Deny Push actions to Master branch.

Integration with AWS Services

AWS CodeCommit integrates with various other services like Lambda, EventsBridge, CloudTrail, CodeBuild, AWS KMS and others.



AWS CodeCommit

Trigger Lambda

Trigger configuration

CodeCommit aws developer-tools git

Repository name
Select the repository to add a trigger to.
demo-code-commit

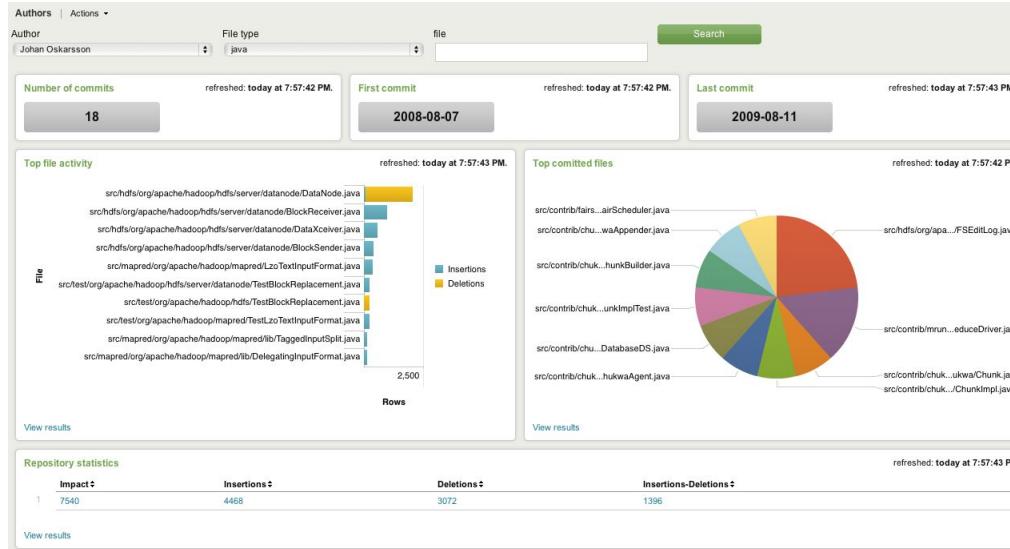
Trigger name
Provide a name for the trigger that will invoke this function.
lambda-trigger

Events
Choose one or more events to listen for. If you choose "All repository events", you cannot choose other event types.
Create branch or tag X

Branch names
This trigger will be configured for all repository branches and tags by default. For a more specific configuration, choose up to 10 branches. If you choose "All branches", you cannot choose specific branches.
main X

Logging CodeCommit API Calls

AWS CodeCommit is integrated with CloudTrail that will allow administrators to capture insights into activities by users.



Notification Rules

You can set up notification rules for a repository so that repository users receive emails about the repository event types you specify.

Notifications are sent when events match the notification rule settings

Notification rule settings

Notification name

Detail type
Choose the level of detail you want in notifications. [Learn more about notifications and security](#)

Full
Includes any supplemental information about events provided by the resource or the notifications feature.

Basic
Includes only information provided in resource events.

Events that trigger notifications

Comments	Approvals	Pull request	Branches and tags
<input type="checkbox"/> On commits	<input checked="" type="checkbox"/> Status changed	<input type="checkbox"/> Source updated	<input checked="" type="checkbox"/> Created
<input checked="" type="checkbox"/> On pull requests	<input type="checkbox"/> Rule override	<input type="checkbox"/> Created	<input checked="" type="checkbox"/> Deleted
		<input type="checkbox"/> Status changed	<input type="checkbox"/> Updated
		<input checked="" type="checkbox"/> Merged	

Data Protection

Data in CodeCommit repositories is encrypted in transit and at rest. When data is pushed into a CodeCommit repository (for example, by calling git push), CodeCommit encrypts the received data as it is stored in the repository.

When data is pulled from a CodeCommit repository (for example, by calling git pull), CodeCommit decrypts the data and then sends it to the caller.

Data sent or received is transmitted using the HTTPS or SSH encrypted network protocols.

Identity Policy for AWS CodeCommit

Standardized Policies

AWS Managed Policies for CodeCommit

AWS addresses many common use cases by providing standalone IAM policies that are created and administered by AWS.

Managed Policy	Description
AWSCodeCommitFullAccess	Grants full access to CodeCommit. Apply this policy only to administrative-level users
AWSCodeCommitPowerUser	Allows users access to all of the functionality of CodeCommit and repository-related resources, except it does not allow them to delete CodeCommit repositories
AWSCodeCommitReadOnly	Grants read-only access to CodeCommit and repository-related resources

Data Protection

Data in CodeCommit repositories is encrypted in transit and at rest. When data is pushed into a CodeCommit repository (for example, by calling git push), CodeCommit encrypts the received data as it is stored in the repository.

It is important to ensure that there is no explicit deny added for KMS in the policy associated with the users otherwise CodeCommit operations would not work.

The screenshot shows a user interface for managing AWS Lambda functions. At the top, there's a breadcrumb navigation: "Developer Tools > CodeCommit > Repositories". Below this, a modal or alert box is displayed. It features a red circular icon with a white "X" inside. To its right, the error type is shown as "EncryptionKeyAccessDeniedException". Below the error type, a descriptive message reads: "User is not authorized for the KMS default master key for CodeCommit 'alias/aws/codecommit' in your account".

Identity Policies For AWS CodeCommit

IAM Policy allows administrators to put a granular level of control over AWS CodeCommit service.

Sample Use-Cases:

- Deny all access to a repository with tag Status as Secret.
- Allow access to repository only in Mumbai region.
- Allow user only connecting from 10.77.2.50 to connect to repository.
- Deny Push actions to Master branch.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "codecommit:*"  
            "Resource": "*"  
            "Condition": {  
                "StringEquals": "aws:ResourceTag/Team": "Payments"  
            }  
        }  
    ]  
}
```

Identity Policies Based On Tags

You can create a policy that allows or denies actions on repositories based on the AWS tags associated with those repositories.

Developer Tools > CodeCommit > Repositories > demo-code-commit > Settings

demo-code-commit

General | Notifications | Triggers | **Repository tags** | Amazon CodeGuru Reviewer

Repository tags Info

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help ma

Key	Value
Team	Payments



```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "codecommit:*"  
            "Resource": "*"  
            "Condition": {  
                "StringEquals": "aws:ResourceTag/Team": "Payments"  
            }  
        }  
    ]  
}
```

Identity Policies Based On Branch

You can create a policy that deny certain write related actions towards the main branch

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Deny",  
            "Action": [  
                "codecommit:GitPush",  
                "codecommit>DeleteBranch",  
                "codecommit:PutFile",  
                "codecommit:Merge*"  
            ],  
            "Resource": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",  
            "Condition": {  
                "StringEqualsIfExists": {  
                    "codecommit:References": [  
                        "refs/heads/main"  
                    ]  
                }  
            }  
        }  
    ]  
}
```

Approval Rule Templates

Code Commit In-Detail

Overview of Approval Rule Templates

Approval Rule Template allows administrators to automatically apply approval rules to pull requests created in the repository.

Approval rule template

Approval rule template name
Approval Required for Main Branch Commit By Senior Member

Description - *optional*
Before Pull Request is Merged to Master, at-least one senior developer must approve the changes.

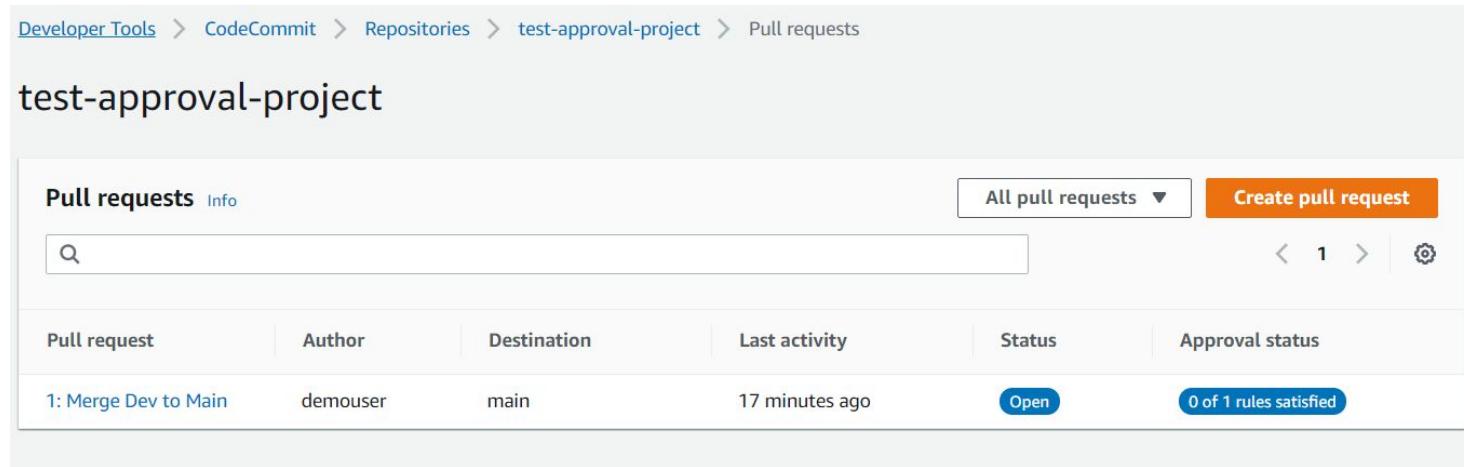
Number of approvals needed
1

Approval pool members - *optional*
If approval pool members are specified, only approvals from these members will count toward satisfying this rule. You can use wildcards to match multiple approvers with one value.

Approver type <small>Info</small>	Value	
Fully qualified ARN <input type="button" value="▼"/>	arn:aws:iam::693331494763:user/Stacy	<input type="button" value="Remove"/>
Fully qualified ARN <input type="button" value="▼"/>	arn:aws:iam::693331494763:user/Matt	<input type="button" value="Remove"/>

Overall Workflow - 1

Whenever a user creates a Pull Request to the specified branch, an approval status would be associated with it.



The screenshot shows the AWS CodeCommit interface for the repository 'test-approval-project'. The navigation bar at the top includes 'Developer Tools' > 'CodeCommit' > 'Repositories' > 'test-approval-project' > 'Pull requests'. The main page title is 'test-approval-project'. Below the title, there's a search bar and a button labeled 'Create pull request'. A table displays the current pull request:

Pull request	Author	Destination	Last activity	Status	Approval status
1: Merge Dev to Main	demouser	main	17 minutes ago	Open	0 of 1 rules satisfied

Overall Workflow - 2

Approval pool members must approve the change.

The screenshot shows a pull request in the AWS CodeCommit interface. At the top, a green success message box says "Success" and "You have approved the pull request." Below the message, the navigation path is: Developer Tools > CodeCommit > Repositories > test-approval-project > Pull requests > 1. The pull request title is "1: Merge Dev to Main". On the right, there are three buttons: "Revoke approval", "Close pull request", and a prominent orange "Merge" button. Below the title, there are status indicators: "Open", "Approved", and "No merge conflicts". The destination is set to "main" and the source is "dev". The author is listed as "demouser" and there is "Approvals: 1". A horizontal navigation bar below the title includes tabs for Details, Activity, Changes, Commits, and Approvals, with the Approvals tab being active. In the Approvals section, there is a table with one row. The table has two columns: "Approver" and "Status". The approver is "Stacy" and the status is "Approved". There is also a "Override approval rules" button and a gear icon.

Approver	Status
Stacy	Approved

Override approval rules

Overall Workflow - 3

The approval status changes to Approved.

Developer Tools > CodeCommit > Repositories > test-approval-project > Pull requests

test-approval-project

Pull requests Info

Open pull requests ▾ Create pull request

Search < 1 > ⚙️

Pull request	Author	Destination	Last activity	Status	Approval status
1: Merge Dev to Main	demouser	main	21 minutes ago	Open	Approved

AWS CodeBuild

Building Code

Overview of AWS CodeBuild

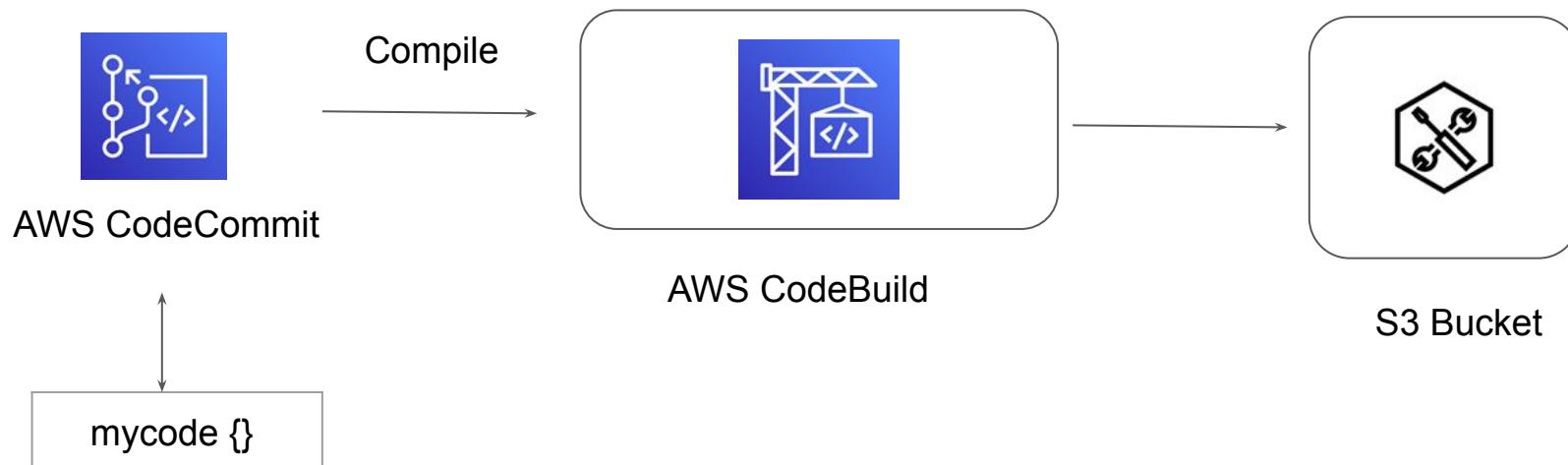
AWS CodeBuild is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy.



Input and Output Sources

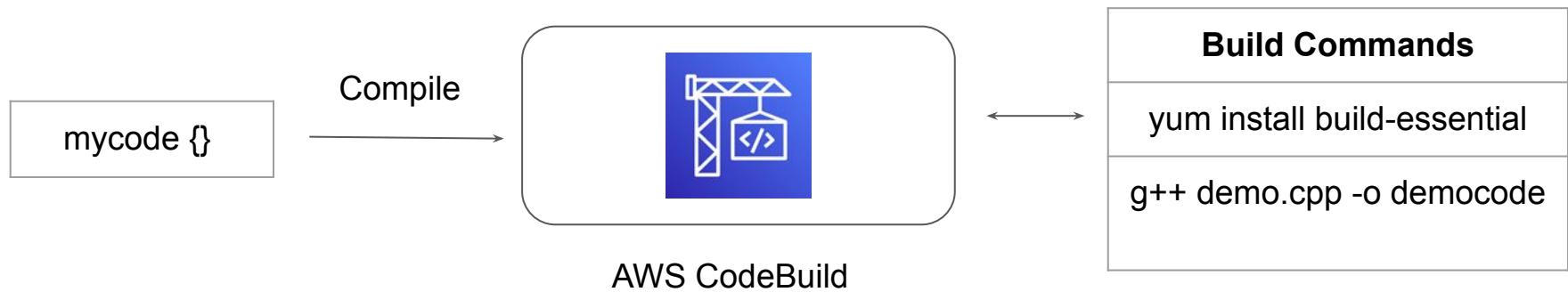
AWS CodeBuild can fetch the code from various sources like S3, CodeCommit, GitHub.

AWS CodeBuild can send the output artifacts to AWS S3



Build Specification

A buildspec is a collection of build commands and related settings, in YAML format, that CodeBuild uses to run a build.



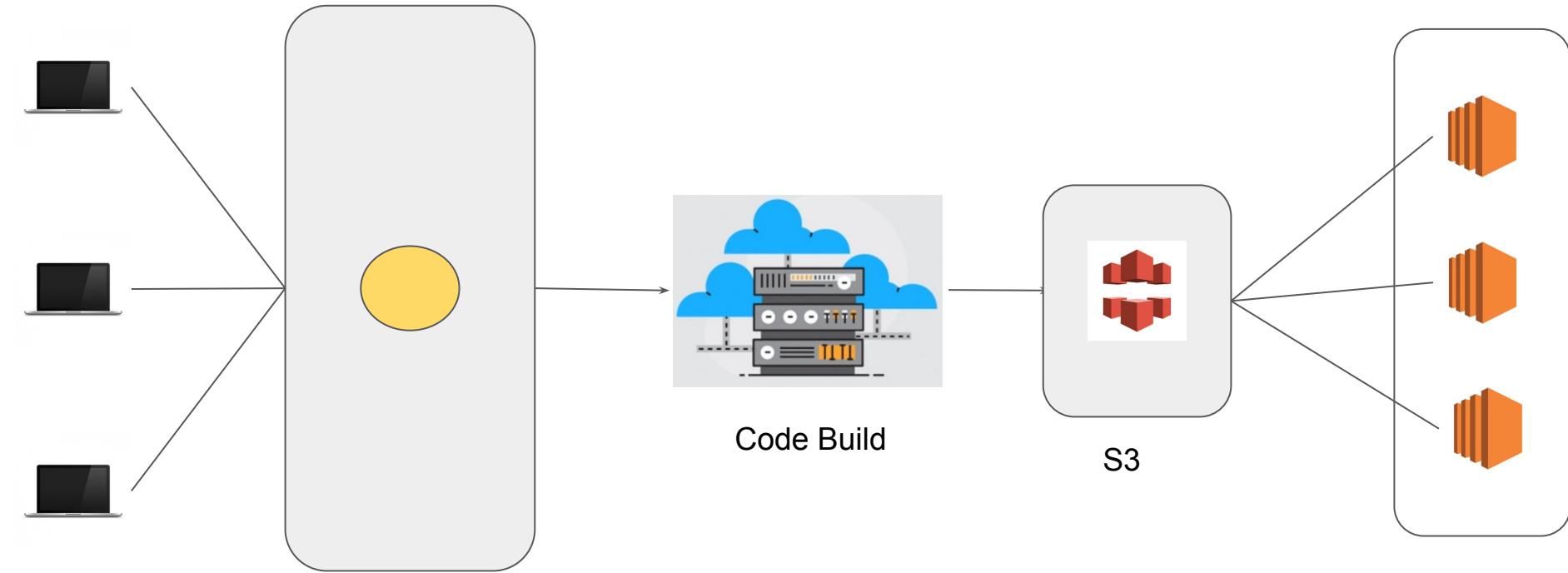
Sample Build Specification

```
version: 0.2
phases:
  install:
    commands:
      - apt-get update -y
      - apt-get install -y build-essential
  build:
    commands:
      - g++ demo.cpp -o democode
```

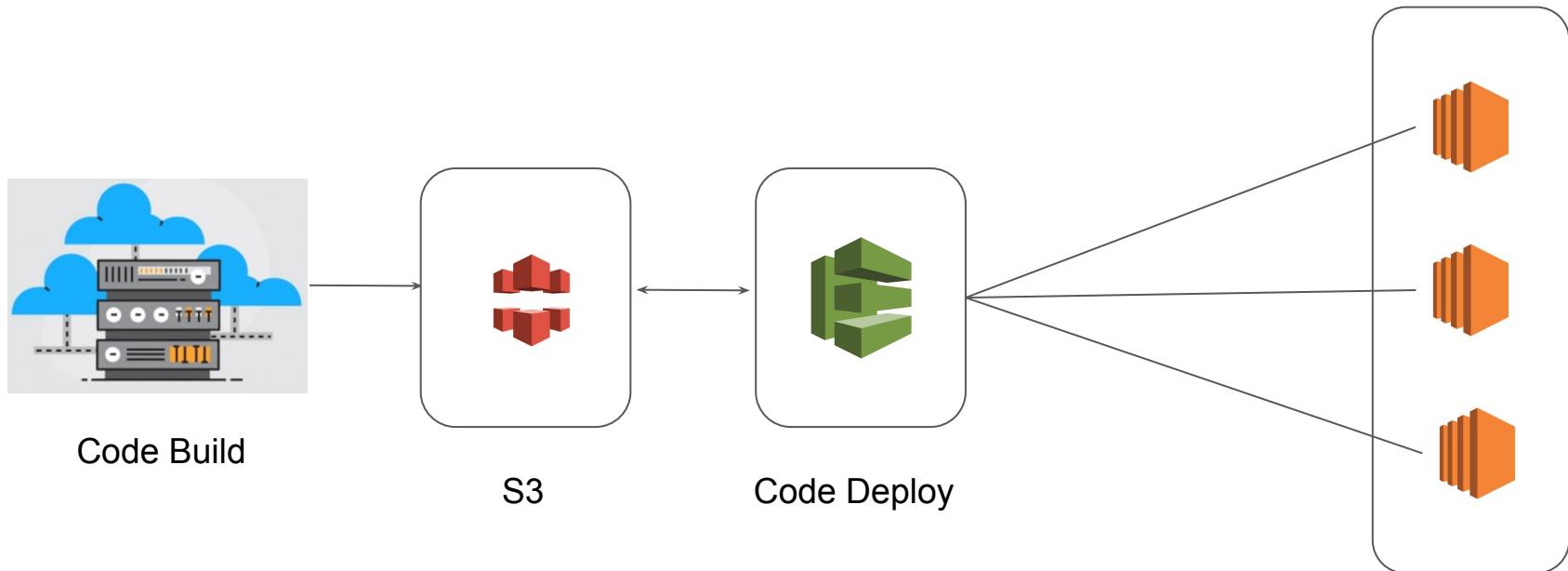
Code Deploy

Deployment is the key

Deployment of Code



Deployment of Code via Code Deploy



Code Deploy Configuration

There are certain steps needed for Codedeploy to be configured.

- Create an IAM role with permission for S3 and EC2.
- Install Codedeploy Agent
- Configure Codedeploy Application

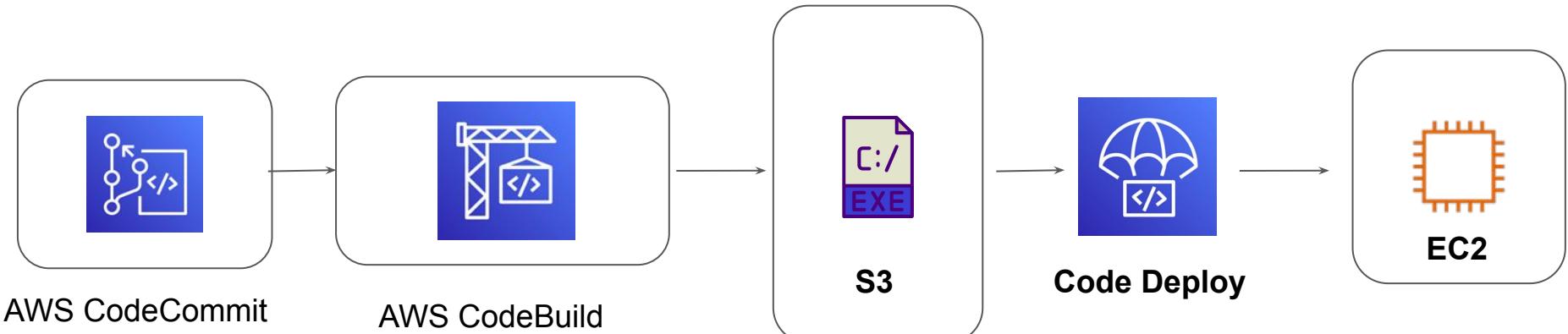
Code Pipeline

Automating Deployments

Current Setup

At this stage, we have the pipeline setup using Code Commit, Code Build and CodeDeploy

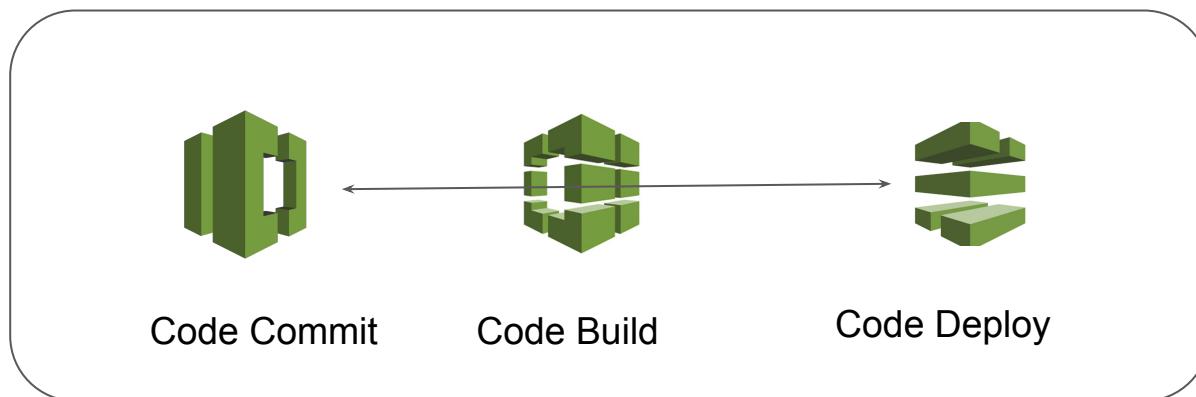
Challenge: The Entire Process is Manual.



Overview of Code Pipeline

AWS Codepipeline is a continuous delivery service to automate steps required to release the software.

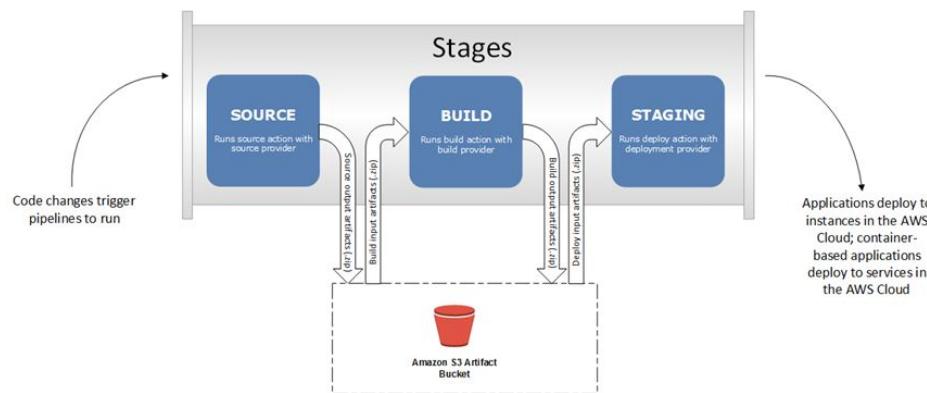
It allows us to launch the entire continuous delivery toolchain in minutes, allowing releasing code faster.



Important Pointer - Code Pipeline

Codepipeline automatically triggers your pipeline whenever there is a commit in the source repository.

- Output artifact is ingested into input artifact to the Build stage.
- Output artifact from build stage (build) acts as input to the deploy stage.



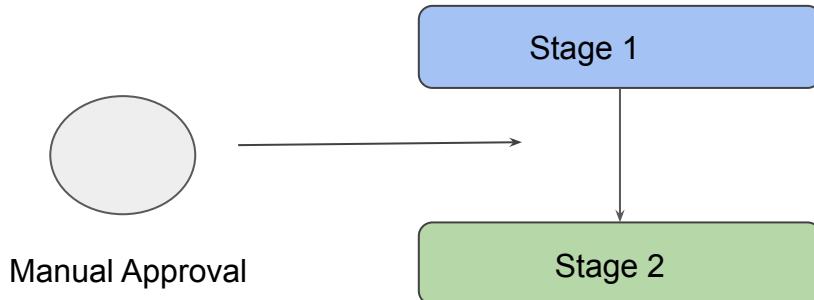
Stage Transitions in CodePipeline

Deployments, yet again!

Getting Started

Transitions are links between pipeline stages that can be disabled or enabled.

We can make use of an approval action to pause the run of a pipeline until it is manually approved to continue.



CodeDeploy - Deployment Configuration

Need to learn the backend

Getting Started

A deployment configuration is a set of rules and success and failure conditions used by AWS CodeDeploy during a deployment.

There are three deployment configuration which is provided:

- CodeDeployDefault.AllAtOnce
- CodeDeployDefault.HalfAtATime
- CodeDeployDefault.OneAtATime

Default All At Once

Attempts to deploy an application revision to as many instances as possible at once.

The status of the overall deployment is displayed as Succeeded if the application revision is deployed to one or more of the instances.

The status of the overall deployment is displayed as Failed if the application revision is not deployed to any of the instances.

Default All At Once

Example Use-Case:

If there are 9 EC2 instances, , `CodeDeployDefault.AllAtOnce` will attempt to deploy to all nine instances at once.

The overall deployment succeeds if deployment to even a single instance is successful. It fails only if deployments to all nine instances fail.

Default Half At A Time

Deploys to up to half of the instances at a time.

The overall deployment succeeds if the application revision is deployed to at least half of the instances, otherwise deployment fails.

Example:

If there are 9 EC2 instances, it deploys to up to four instances at a time. The overall deployment succeeds if deployment to five or more instances succeed. Otherwise, the deployment fails.

Default One At A Time

Deploys the application revision to only one instance at a time.

The overall deployment succeeds if the application revision is deployed to all of the instances. With exception that if deployment in last instance fails, the overall deployment still succeeds.

The overall deployment fails as soon as the application revision fails to be deployed to any but the last instance.

AppSpec Hooks

Deployment Flexibility

Overview of Deployment LifeCycle

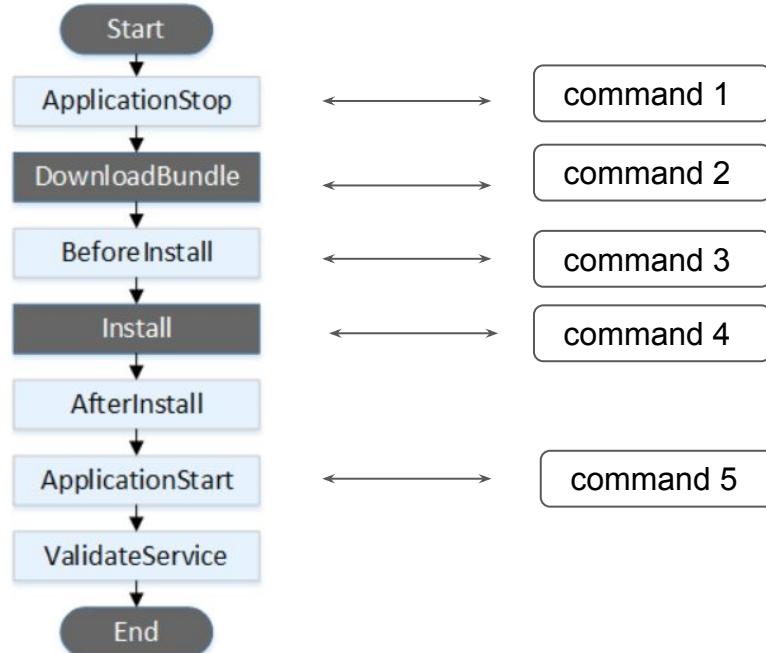
During the deployment lifecycle, there are many steps you might take while installing and configuring the updated application.

Example:

1. Block the Traffic to the Server.
2. Stop and Remove the older application.
3. Pull the latest application from repository.
4. Install the application.
5. Start the Service.
6. Validate.

AppSpec Hooks in CodeDeploy

The hooks section in AppSpec contains mappings that link deployment lifecycle event hooks to one or more scripts.



Hooks	Description
ApplicationStop	Gracefully stop your existing application.
DownloadBundle	CodeDeploy agent copies latest application to a temporary location.
BeforeInstall	You can use this deployment lifecycle event for preinstall tasks, such as decrypting files and creating a backup of the current version.
Install	Install the application.
After Install	You can use this deployment lifecycle event for tasks such as configuring your application or changing file permissions.
ApplicationStart	Start your application.
ValidateService	Create a Validate Logic to verify if deployment is successful.

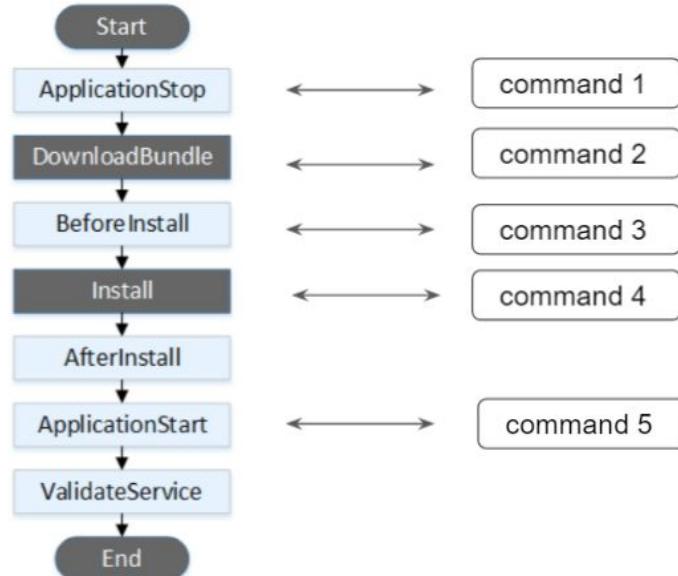
Sample AppSpec Files Using Hooks

```
version: 0.0
os: linux
hooks:
  BeforeInstall:
    - location: Scripts/UnzipResourceBundle.sh
    - location: Scripts/UnzipDataBundle.sh
  AfterInstall:
    - location: Scripts/RunResourceTests.sh
      timeout: 180
  ApplicationStart:
    - location: Scripts/RunFunctionalTests.sh
      timeout: 3600
  ValidateService:
    - location: Scripts/MonitorService.sh
      timeout: 3600
      runas: codedeployuser
```

Challenges with ELB Environment

For ELB environment, we cannot directly start with ApplicationStop.

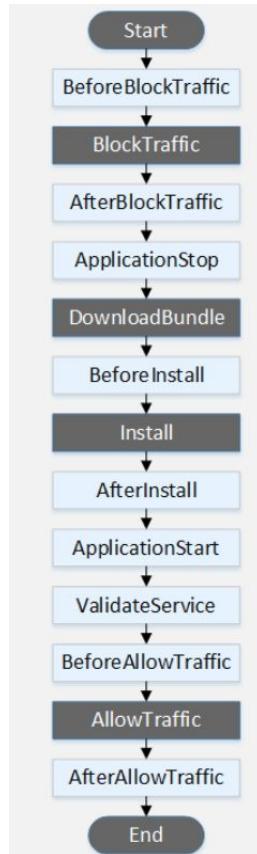
Users will start seeing errors with the following lifecycle hooks.



LifeCycle for ELB

High-Level Workflow

1. De-Register EC2 from ELB
2. Perform App Updates
3. Register EC2 with ELB



Hooks	Description
BeforeBlockTraffic	Run tasks on instances before they are deregistered from a load balancer.
BlockTraffic	Internet traffic is blocked from accessing instances that are currently serving traffic. This event is reserved for the CodeDeploy agent and cannot be used to run scripts.
AfterBlockTraffic	Run tasks on instances after they are deregistered from a load balancer.
BeforeAllowTraffic	Run tasks on instances before they are registered with a load balancer.
AllowTraffic	Internet traffic is allowed to access instances after a deployment. This event is reserved for the CodeDeploy agent and cannot be used to run scripts.
AfterAllowTraffic	Run tasks on instances after they are registered with a load balancer.

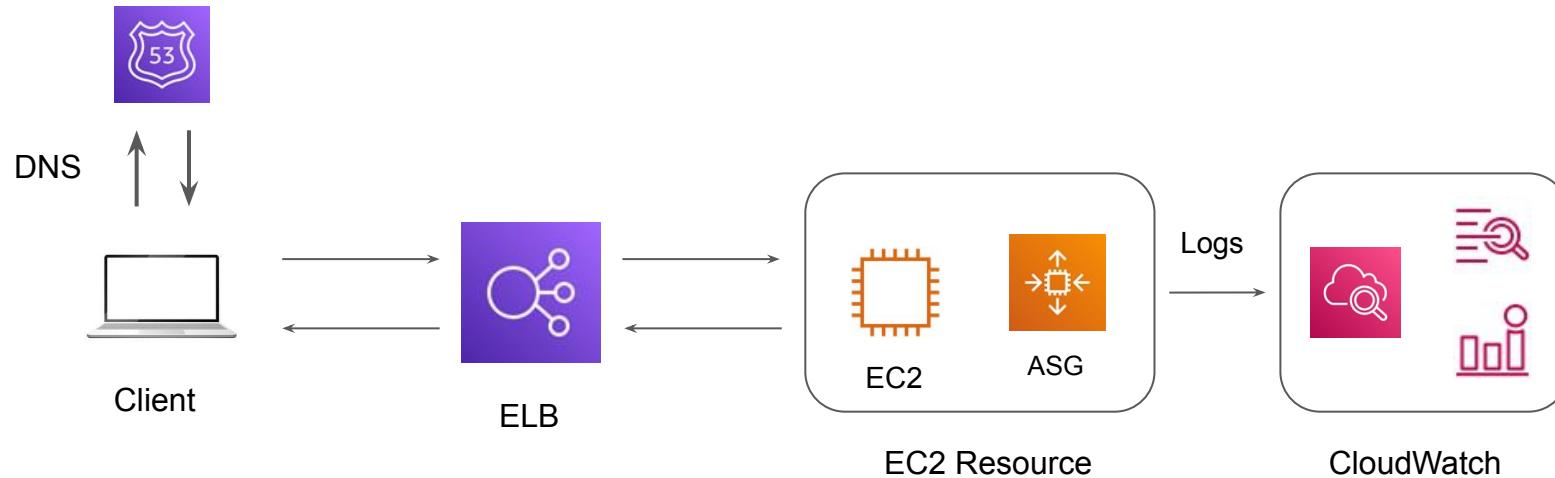
Elastic Beanstalk

Orchestration

Traditional Deployment Approach

Use-Case: Deploy a simple Hello World application for production.

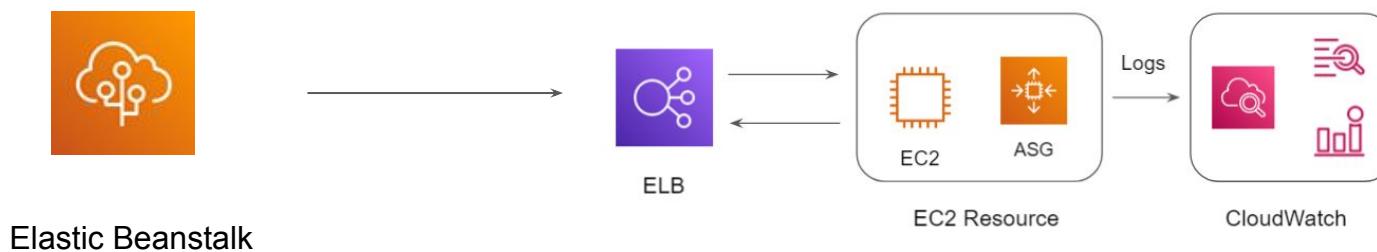
Resources to be created: AWS EC2, ELB, Auto-Scaling, Web-Server Configuration, and others.



Elastic Beanstalk Deployment Approach

Use-Case: Deploy a simple Hello World application for production.

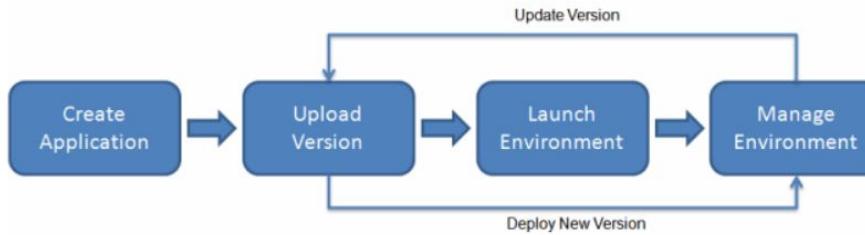
Create Elastic Beanstalk Environment



Overview of Elastic Beanstalk

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services.

You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring.



EB Deployment Policy

Deploying Application Updates

Deployment Policy Options

Deployment Policy specifies how new updates for the applications are pushed into the EB environment.

Elastic Beanstalk supports several options for deployments

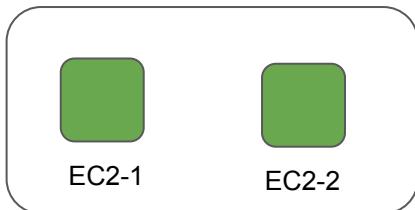
- All at Once
- Rolling
- Rolling with Additional Batch
- Immutable
- Traffic Splitting
- Blue/Green

All at Once

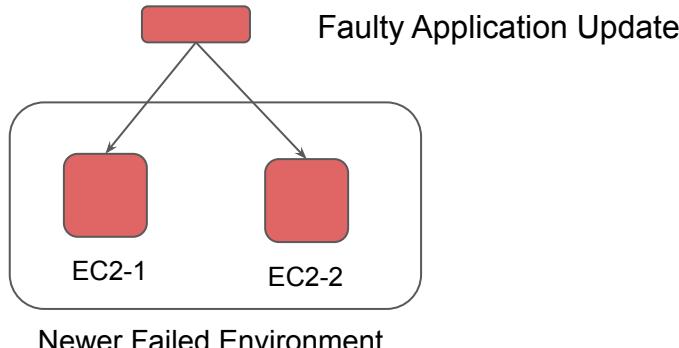
Deploys new version to all the instances simultaneously.

Applications might be unavailable for the users for short-period of time.

If updates fail then you will need to roll back changes by re-deploying the previous working version.



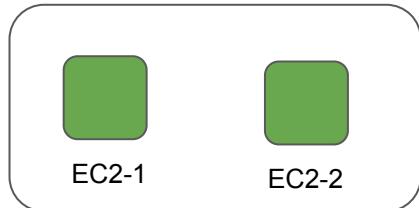
Working Environment



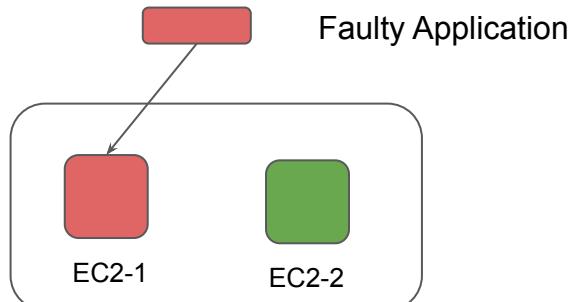
Rolling Deployment Policy

Your application is deployed to your environment one batch of instances at a time.

Each batch of instances is taken out of service while deployment is taking place.



Older Working Environment



Newer Failed Environment

Important Pointer - Rolling Deployment Policy

The overall capacity (in terms of servers) will be reduced while the deployment is happening.

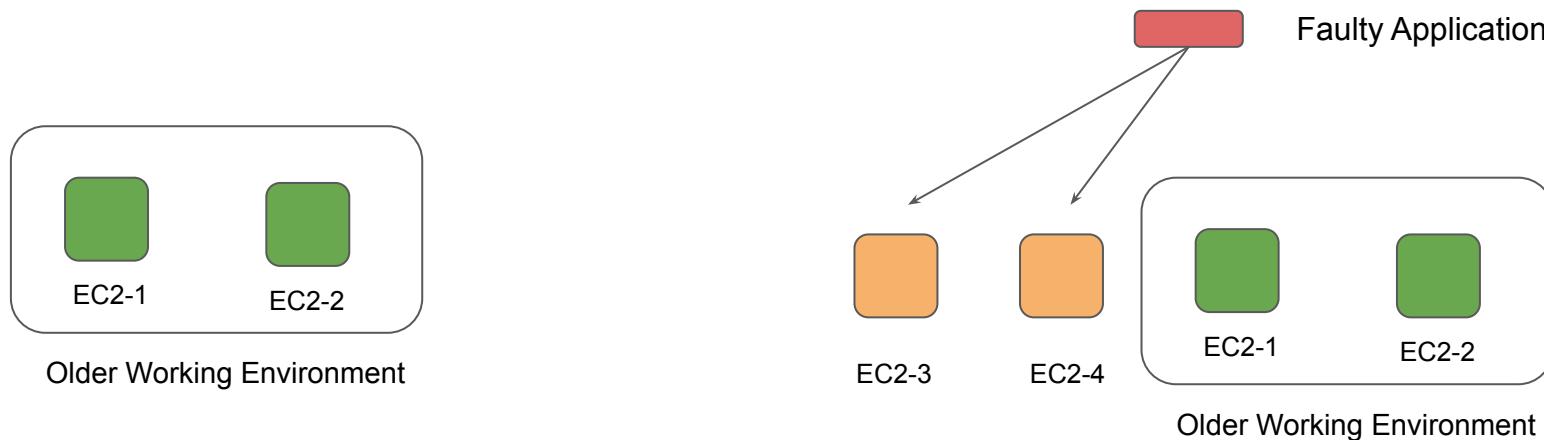
Not recommended for performance-critical applications.

The overall deployment time is much longer then All at Once approach.

Rolling with additional batch

With this method, Elastic Beanstalk launches an extra batch of instances, then performs a rolling deployment.

This maintains full capacity during deployments.



Important Pointer - Rolling with Additional Batch

The deployment is first made to the additional batch instances.

If the deployment fails, these additional batch instances are terminated.

If the deployment succeeds, these additional batch instances are registered under load balancer and the older instances are terminated.

Immutable Deployment Policy

Deploys newer version of the application in a completely new servers under new auto-scaling group.

When new instances passes their health-check, they are moved to older auto-scaling group and older ones are terminated.

Impact of failed update is less as all we need to do is delete the new auto-scaling group and EC2.

Preferred option for mission critical production systems.

EB Deployment - Rolling Update

Deploying Application Updates

Deployment Policy Options

Deployment Policy specifies how new updates for the applications are pushed into the EB environment.

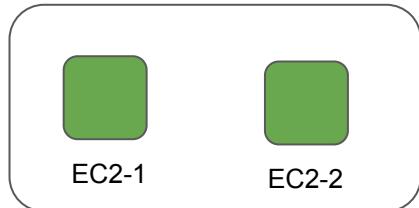
Elastic Beanstalk supports several options for deployments

- All at Once
- Rolling
- Rolling with Additional Batch
- Immutable
- Traffic Splitting
- Blue/Green

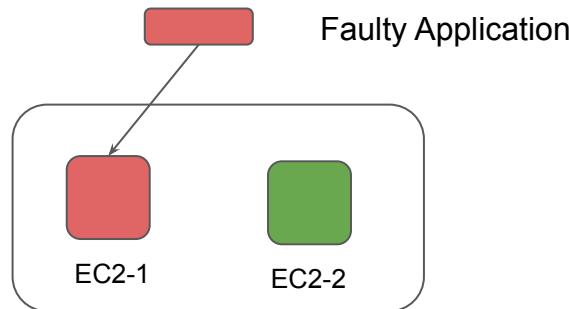
Rolling Deployment Policy

Your application is deployed to your environment one batch of instances at a time.

Each batch of instance is taken out of service while deployment is taking place.



Older Working Environment



Newer Failed Environment

Important Pointer - Rolling Deployment Policy

The overall capacity (in terms of servers) will be reduced while the deployment is happening.

Not recommended for performance critical applications.

The overall deployment time is much longer than All at Once approach.

Rolling with Additional Batch

EB Deployment Policy

Deployment Policy Options

Deployment Policy specifies how new updates for the applications are pushed into the EB environment.

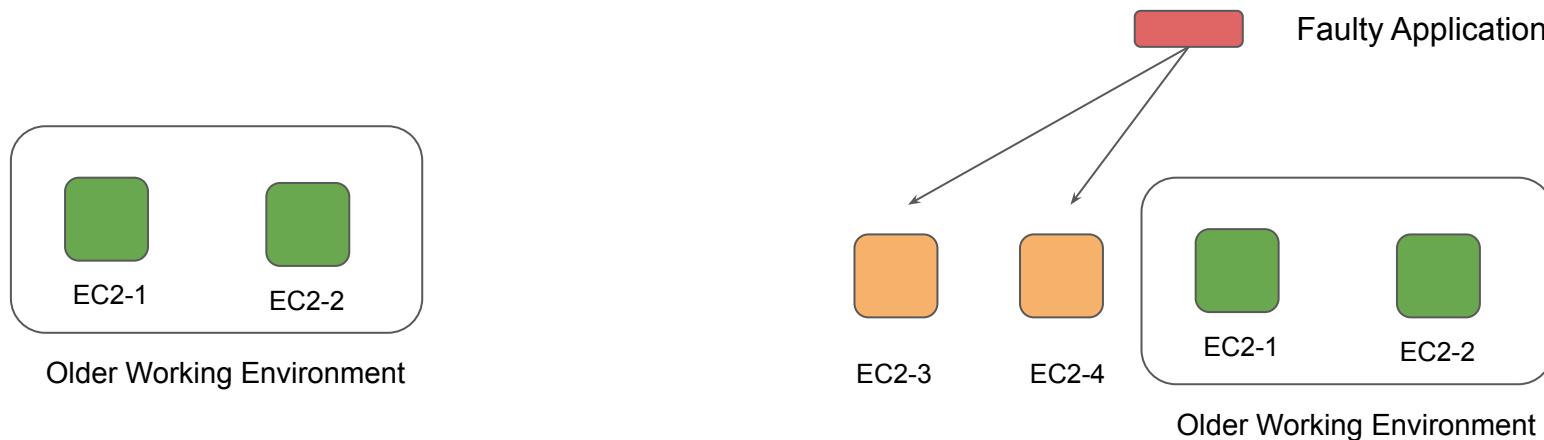
Elastic Beanstalk supports several options for deployments

- All at Once
- Rolling
- Rolling with Additional Batch
- Immutable
- Traffic Splitting
- Blue/Green

Rolling with additional batch

With this method, Elastic Beanstalk launches an extra batch of instances, then performs a rolling deployment.

This maintains full capacity during deployments.



Important Pointer - Rolling with Additional Batch

The deployment is first made to the additional batch instances.

If the deployment fails, these additional batch instances are terminated.

If the deployment succeeds, these additional batch instances are registered under load balancer and the older instances are terminated.

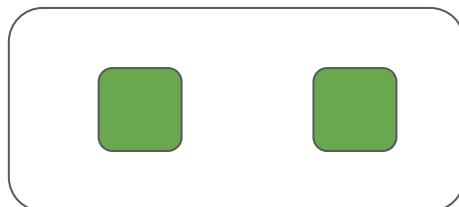
Immutable Deployment Policy

Deploying Application Updates

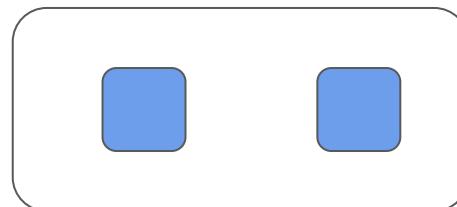
Immutable Deployment Policy

Deploys a newer version of the application in a completely new server under a new auto-scaling group.

When new instances pass their health check, they are moved to the older auto-scaling group and older ones are terminated.



Auto-Scaling Group 1



Auto-Scaling Group 2

Blue Green Deployments

Deployments, yet again!

Getting Started

Blue environment is an existing environment in production receiving live traffic.

Green environment is parallel environment running different version of the application.

Deployment = Routing production traffic from blue to green environment.



Various ways to achieve Blue Green

1. Updating DNS Routing via Route53
2. Swap the Auto-Scaling Groups behind your ELB
3. Swap Launch Configurations
4. Swap Beanstalk Environments (applies only to EB environments)
5. Clone Stack with AWS OpsWorks and updating DNS

EB CLI

Need to learn the backend

Getting Started

EB CLI is a command-line-interface for elastic beanstalk that simplifies the working with the EB way in an automated way.

There are two things that we should be aware about:

- i) Installing the EB CLI
- ii) Basic working with the EB CLI.

ebextensions

Elastic Beanstalk Extensions

Understanding ebextensions

ebextensions allows us to customize all the things that is needed for our application.

The ideal suggested method is that we never need to SSH into the servers to configure , all the things must be setup via the ebextensions.

.ebextensions are based on YAML format and are quiet easy to write.

We need to start from base

ElasticBeanstalk extension offers following approaches to customize and configure things according to the application being deployed.

- Packages
- Groups
- Users
- Sources
- Files
- Commands
- Services
- Container commands

Commands vs Container Commands

Need to learn the backend

Getting Started

When you deploy something in EB, the defined commands will be executed on all the instances part of your environment.

Container Commands:

- Primary goal is to ensure that your commands is run on only one instance.
- The above can be achieved with `leaders_only: true` directive.

It is very useful in use-cases such as creation of databases, running DB migration scripts and more.

Deployment Instruction

Sample Container Commands:

```
container_commands:  
  collectstatic:  
    command: "django-admin.py collectstatic --noinput"  
  01syncdb:  
    command: "django-admin.py syncdb --noinput"  
    leader_only: true  
  02migrate:  
    command: "django-admin.py migrate"  
    leader_only: true  
  99customize:  
    command: "scripts/customize.sh"
```

Command vs Container Commands

The commands run **before** the application and web server are set up and the application version file is extracted.

Container commands run **after** the application and web server have been set up and the application archive has been extracted, **but before** the application version is deployed.

Canary Deployments

Deployments, yet again!

Story behind Canary

The story goes back to the old british mining practise.

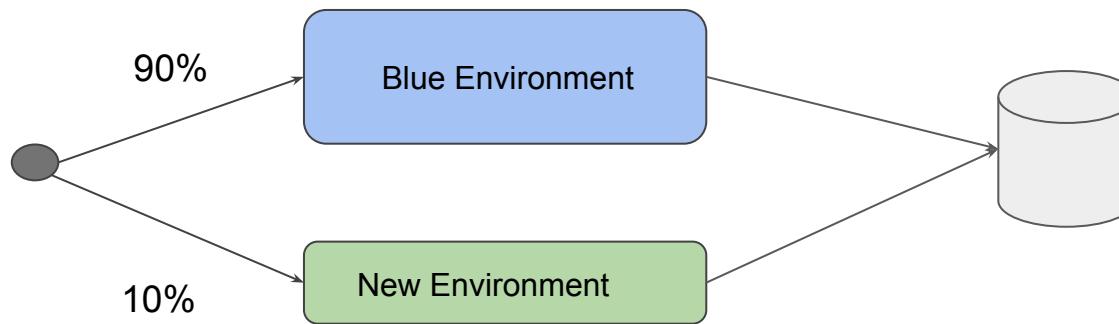
Practice included using "canaries" in coal mines to detect carbon monoxide and other toxic gases before they hurt humans."

To make sure mines were safe for them to enter, miners would take canaries; **if something bad happened** to the canary, it was a warning for the miners to abandon the mine.

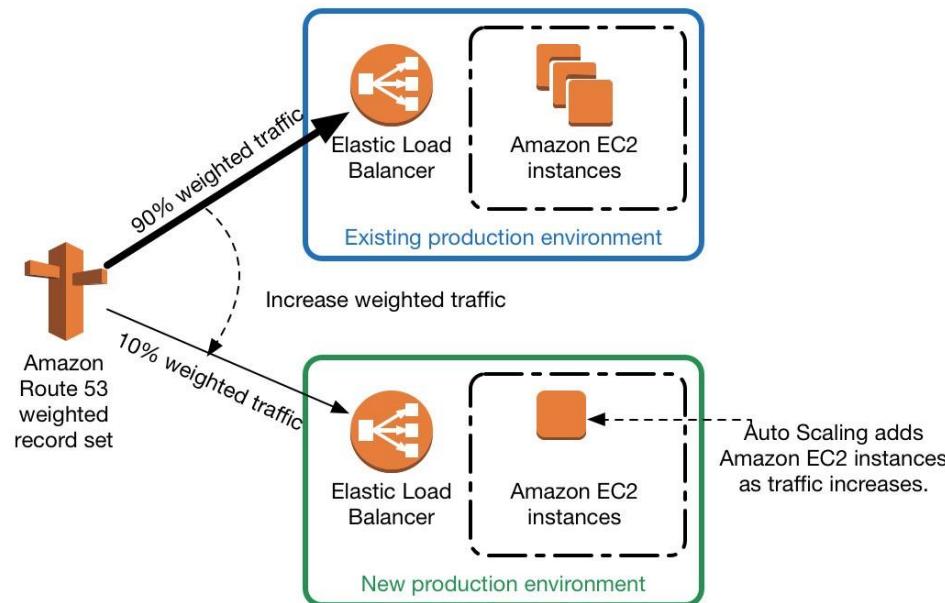


Understanding Canary Deployment Model

Canary Deployment is a process where we deploy a new feature and shift some % of traffic to the new feature to perform some analysis to see if feature is successful.



Canary Deployment Model Architecture



Lambda@Edge

Running Serverless at the Edge

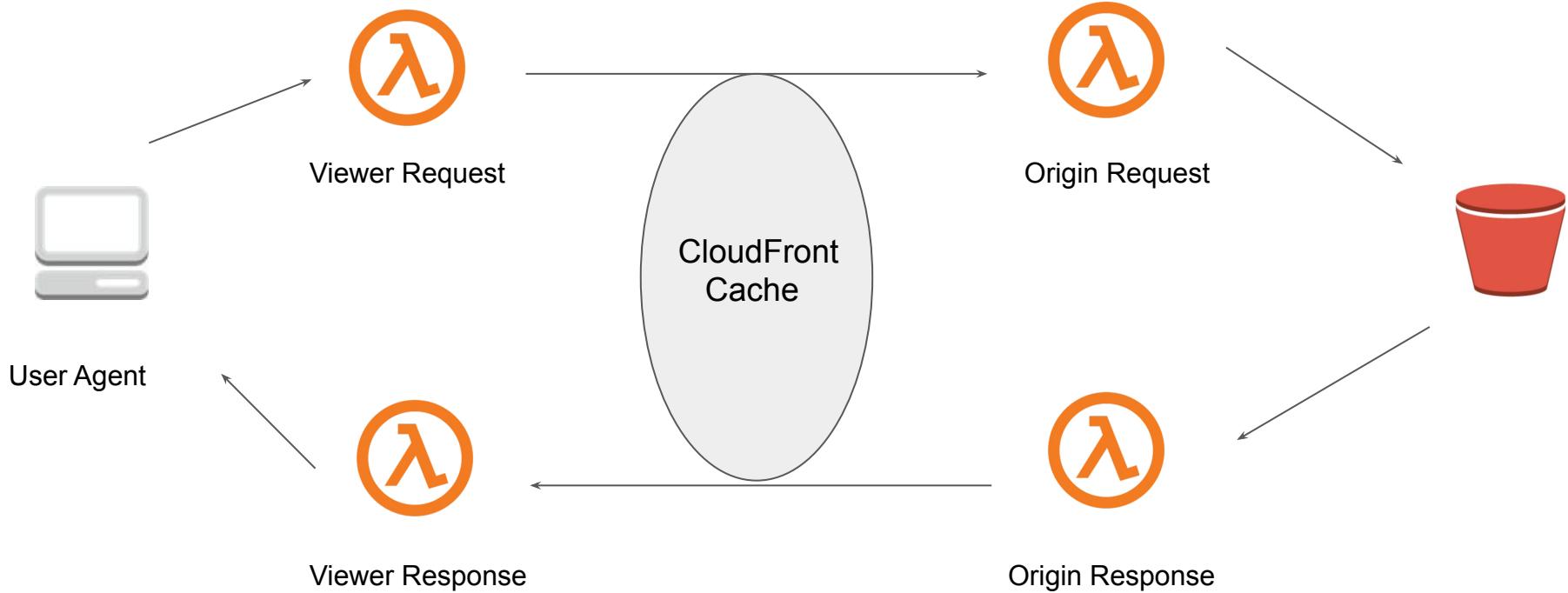
Getting started

Lambda@Edge lets you run Lambda functions to customize content that CloudFront delivers.

You can use Lambda functions to change CloudFront requests and responses at the following points:

1. After CloudFront receives a request from a viewer ([viewer request](#))
2. Before CloudFront forwards the request to the origin ([origin request](#))
3. After CloudFront receives the response from the origin ([origin response](#))
4. Before CloudFront forwards the response to the viewer ([viewer response](#))

Diagrammatic Representation



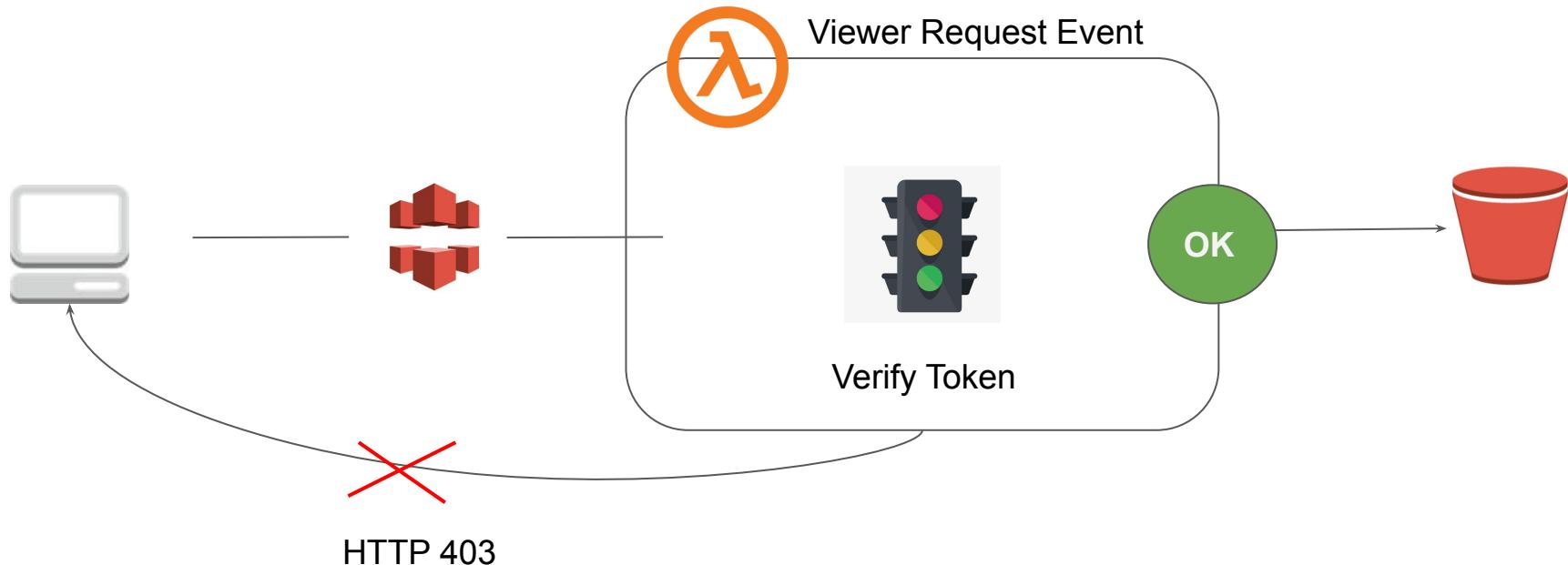
Viewer Request

Viewer Request is executed on every request before CloudFront cache is checked.

There are various things that we can do at this stage, like:

- Modify URLs, cookies query strings etc.
- Perform Authentication and Authorization Checks.

Viewer Request



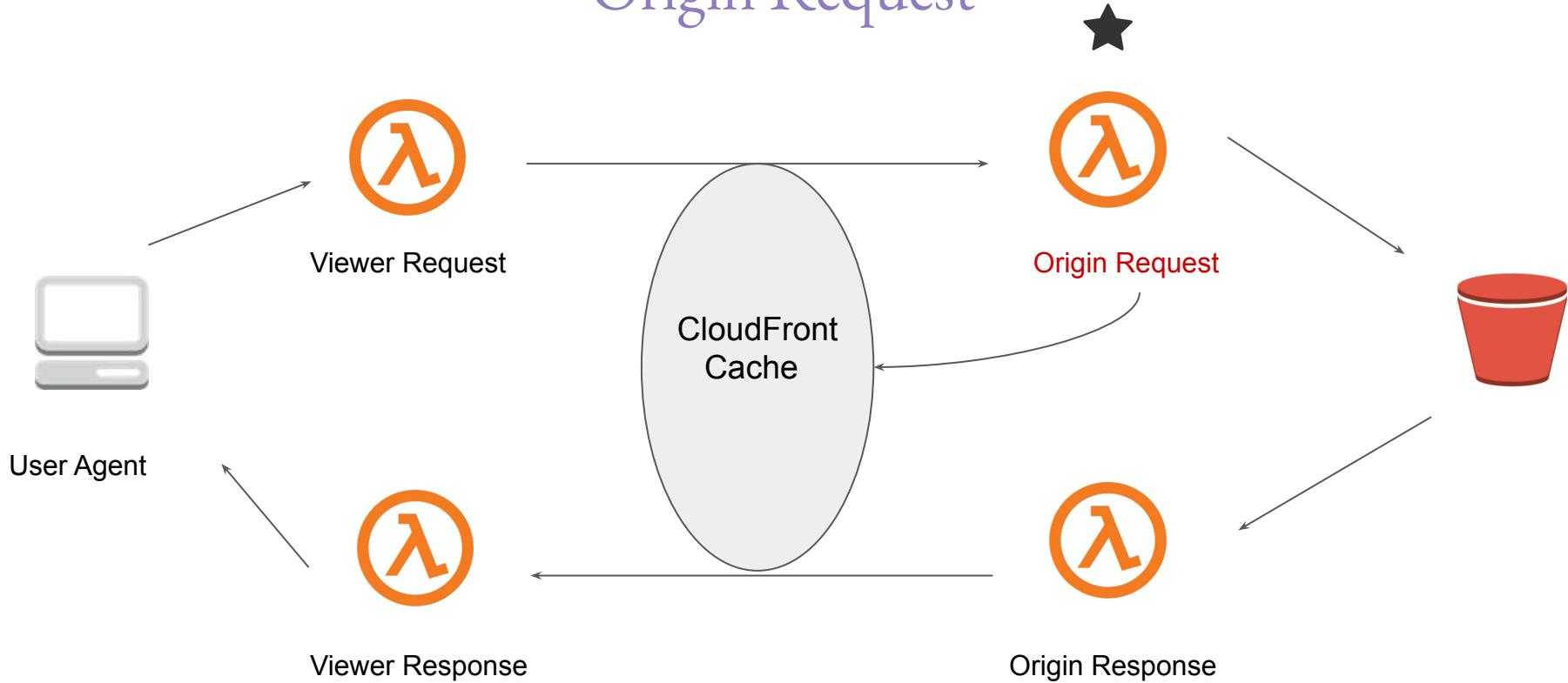
Origin Request

Executed on cache miss, before a request is forwarded to the origin.

There are various things that we can do at this stage, like:

- Dynamically select origin based on the request headers

Origin Request



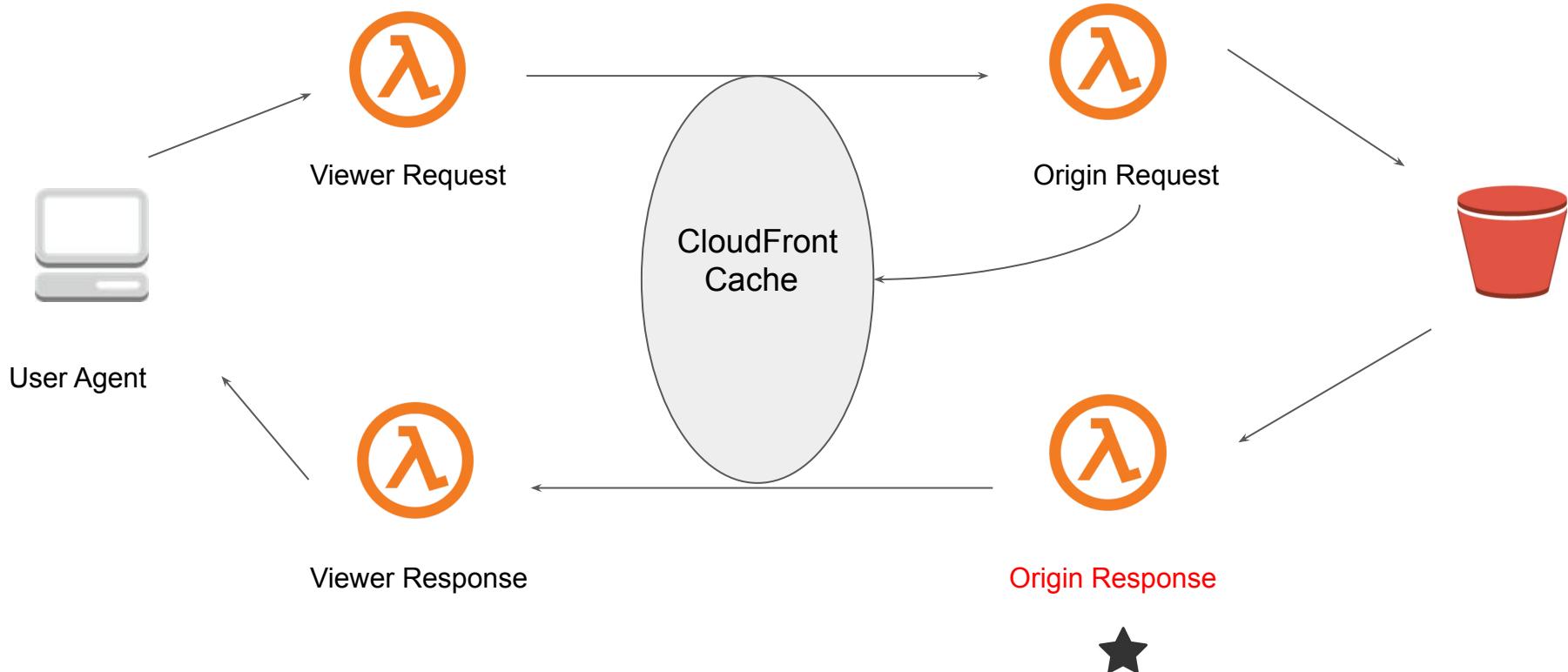
Origin Response

Executed on a cache miss, after a response is received from the origin.

There are various things that we can do at this stage, like:

- Modify the response headers.
- Intercept and replace various 4XX and 5XX errors from the origin.

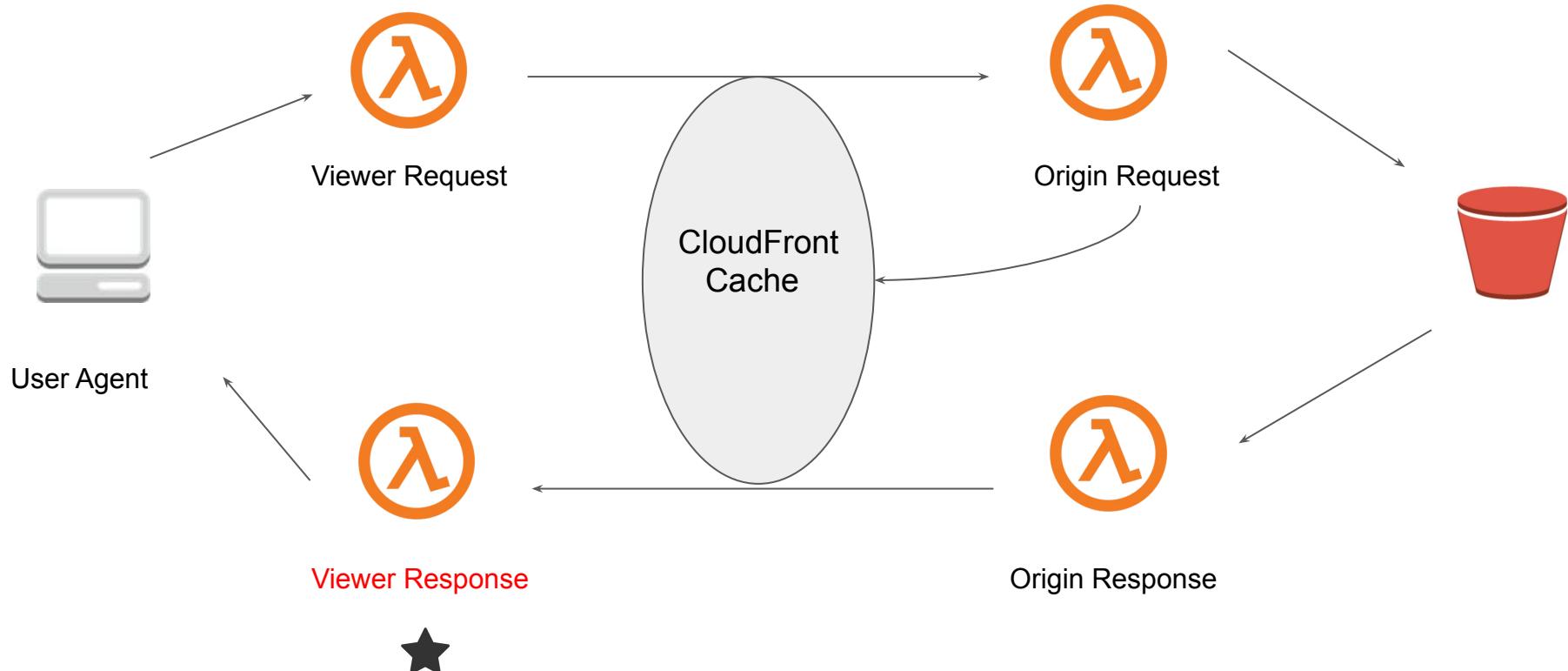
Origin Response



Viewer Response

Executed on all the responses received either from the origin or the cache.

Modifies the response headers before caching the response.

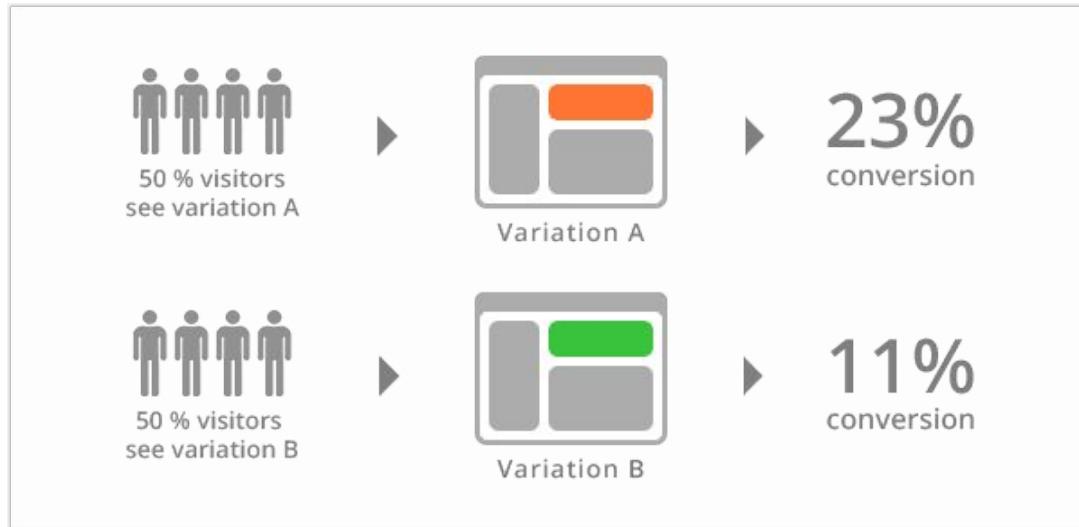


A/B Testing

Deployments, yet again!

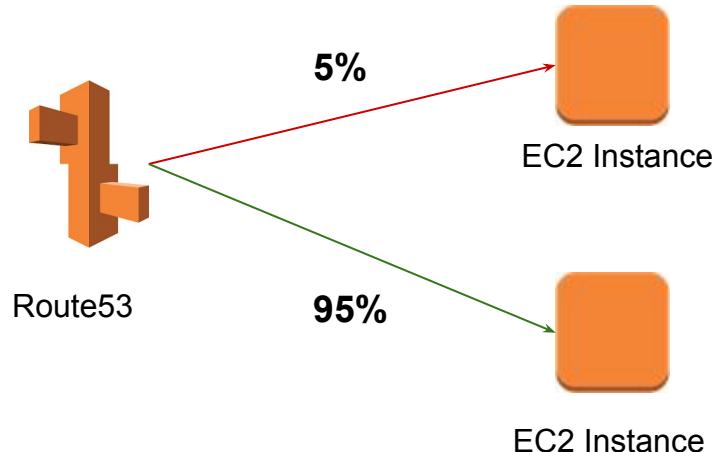
Understanding A/B Testing

A/B Testing (also referred as split testing) is comparing two versions of a web page to see which one performs better.



A/B Testing Architecture

- The deployment strategy of A/B is similar to Blue/Green deployment model.
- There are two copies of our environment running.
- Collect feedback and verify if scaling works as expected.



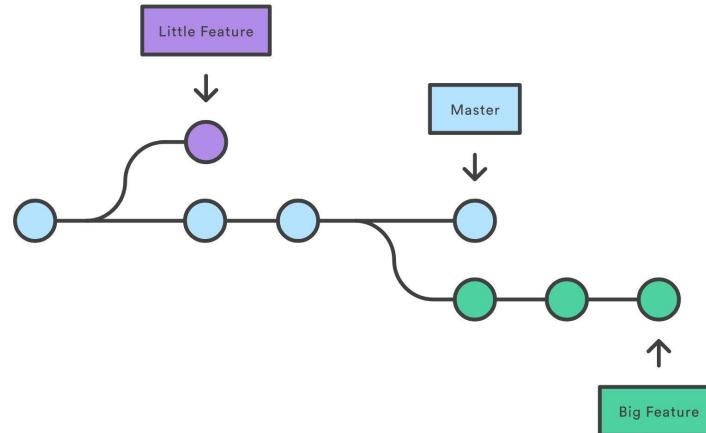
Git Branching Model

Overview of Git

Overview of Git Branches

When you want to add a new feature or fix a bug—no matter how big or how small—you spawn a new branch to encapsulate your changes.

- In the below diagram, we have two branches, Little Feature and Big Feature.
- Working on code in branch allows master to be clean from questionable code.



Overview of Pull Requests

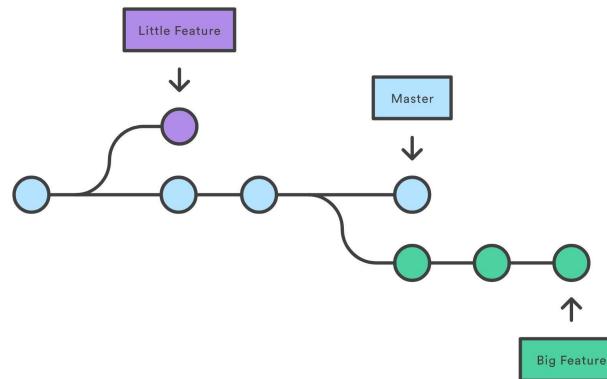
Overview of Git

Overview of Pull Request

Pull Requests are generally used in conjunction with the Git branching workflow.

In simple terms, pull requests are a mechanism for a developer to notify team members that they have completed a feature.

This lets everybody involved know that they need to review the code and merge it into the master branch.



Validating PR with Code Build

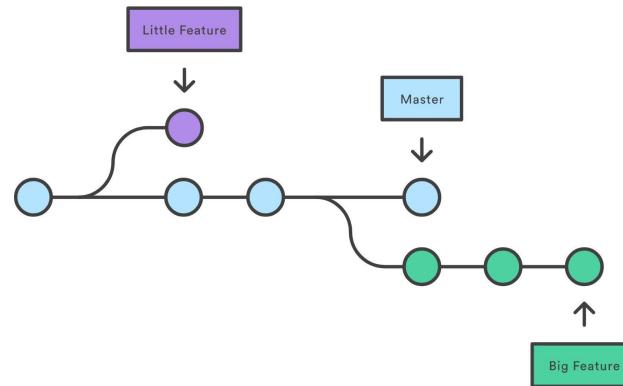
Understanding the Best Practices

Overview of PR Validation

Whenever a developer completes working on his branch, he can create a pull request.

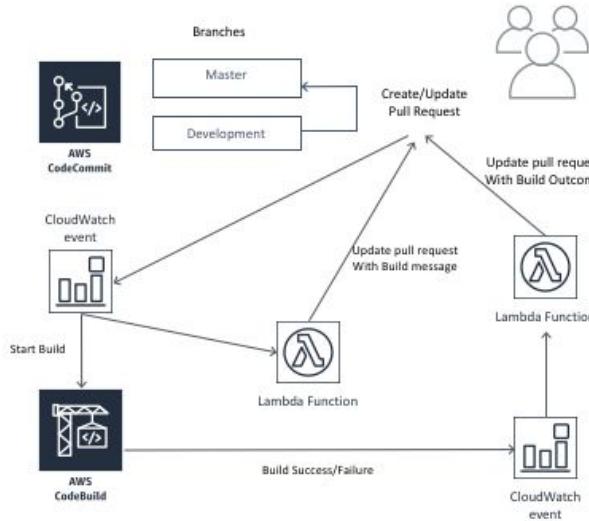
If the code from branch has issue and is merged with master, the prod CI/CD pipeline will break.

Hence. It's important to validate whether changes in Pull Request are good enough.

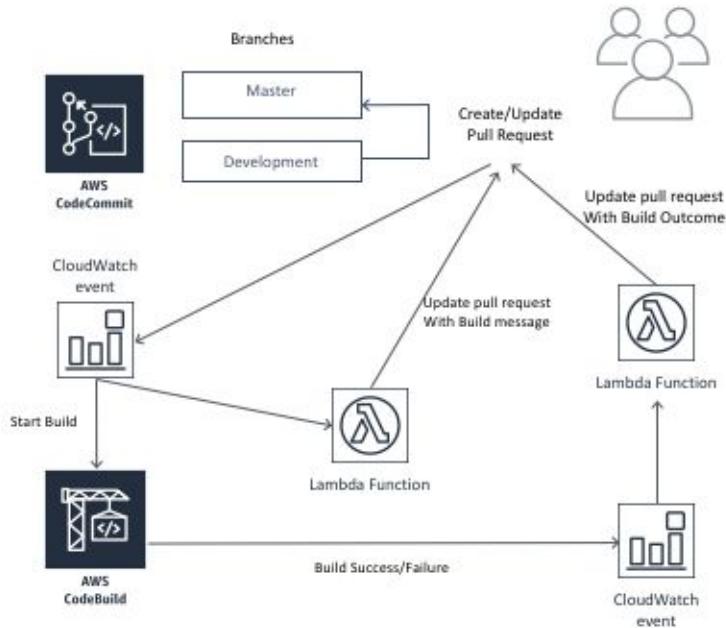


Understanding the Workflow

1. Developer creates Pull Request to merge into master.
2. Creation of PR will trigger CloudWatch which will invoke Lambda and Code Build.
3. Code Build will test the and validate the changes.
4. Once Build completes, CW Events detects it. The outcome is updated in PR.



Understanding the Workflow



Automated Tests

SDLC Automation

Overview of Unit Tests

Unit Tests are conducted by developers and test the unit of code(module, component) he or she developed.

Unit testing checks a single component of an application.



Overview of Integration Test

Integration test is the approach in which multiple component of applications are combined and tested in a group,

The purpose of this level of testing is to expose faults in the interaction between units of an application.



Overview of Regression Test

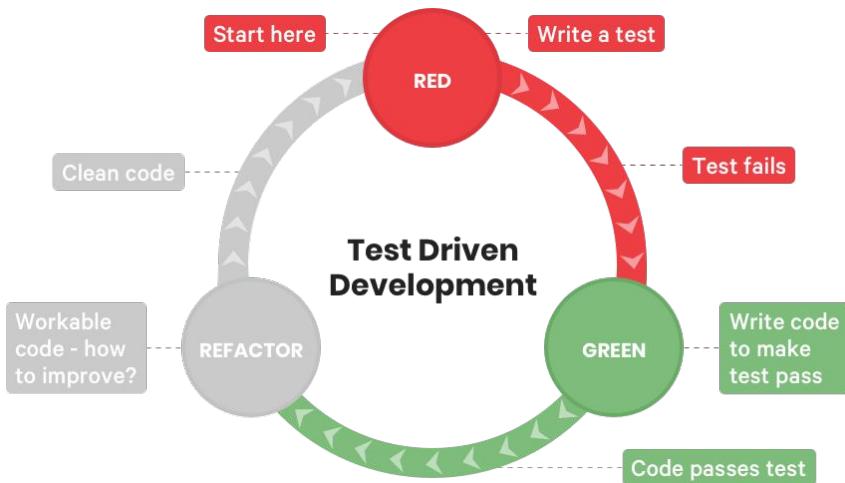
When new software is released, the need to test new functionality is obvious.

Equally important, however, is the need to re-run old tests that the application previously passed, to ensure that new software does not re-introduce old defects or create new ones: side effects called regressions.



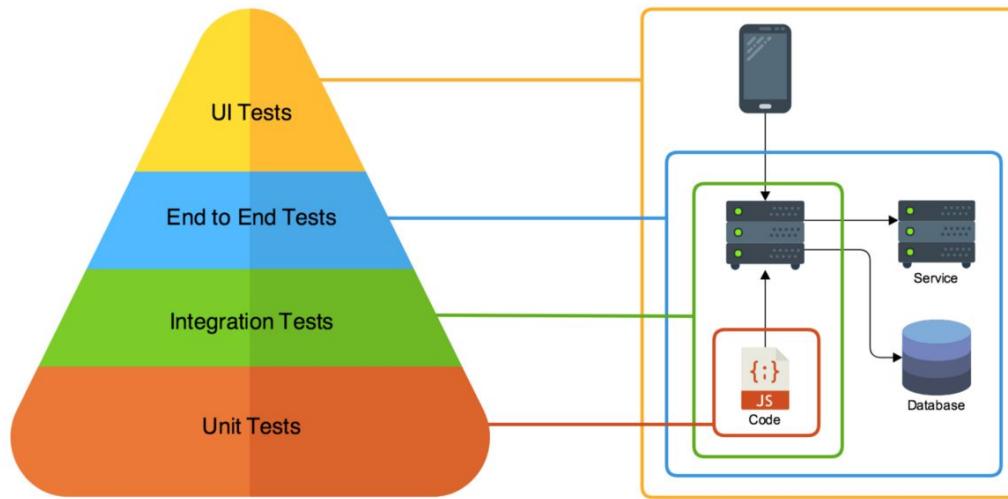
Test Driven Development (TDD)

Practice of writing tests that matches intended behaviour, before you write your code.

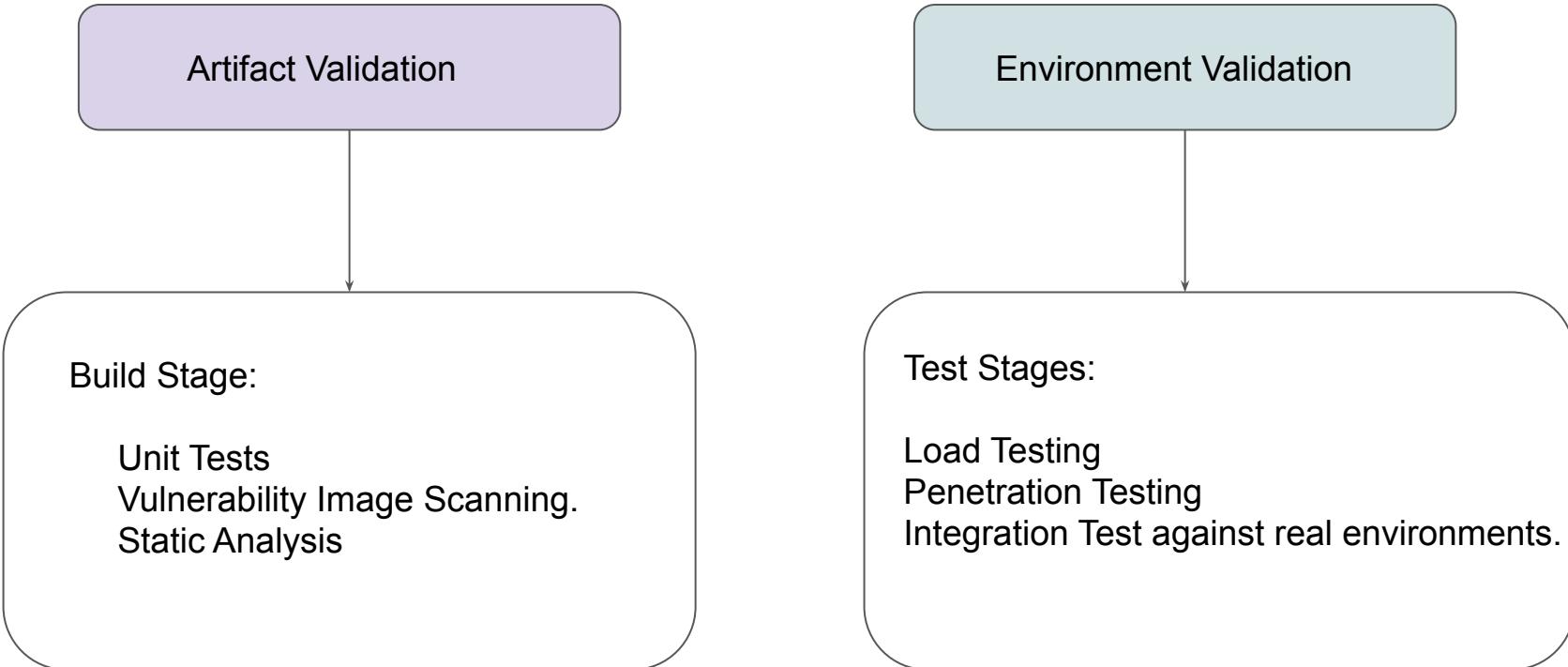


Overview of the Video

There are various tests that are performed at various levels of the SDLC.

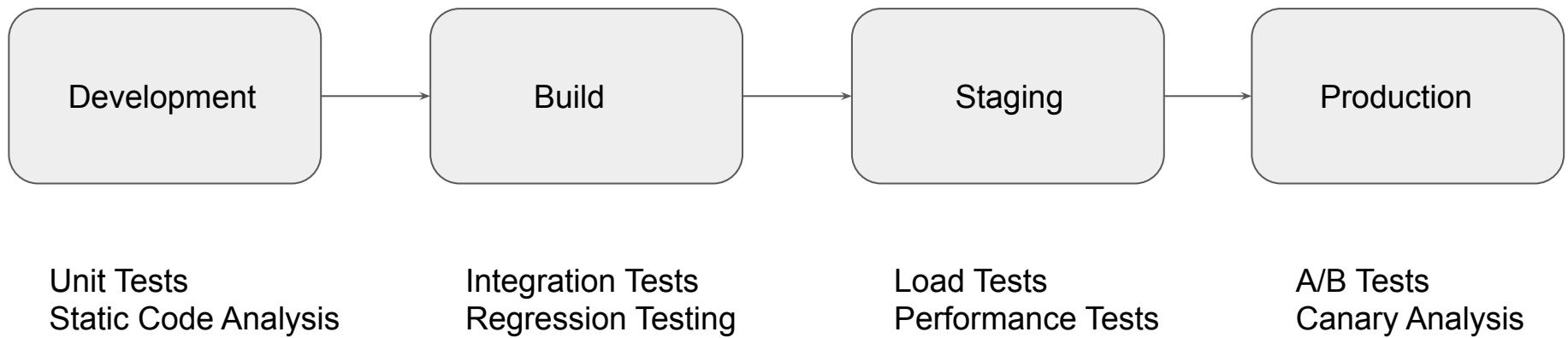


Continues Testing



Continuous Integration

Continuous delivery/ deployment

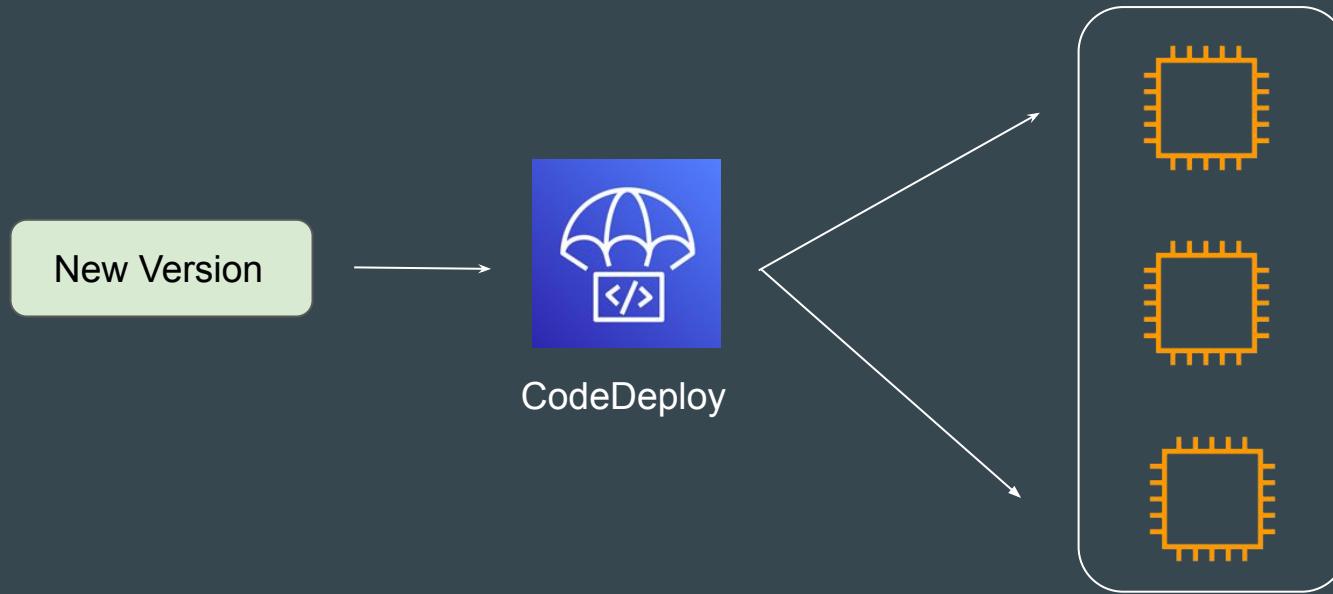


CodeDeploy - Deployment Configuration



Understanding the Basics

Deployment Configuration specifies how a new deployment is rolled out to the underlying platforms.



Point to Note

Depending on the type of platform used, the deployment configuration related options also differs.

Part 1 - EC2/on-premises compute platform

There are 3 predefined deployment configurations for an EC2/on-premises compute platform

You can even create custom deployment configuration if necessary.

Deployment configuration
CodeDeployDefault.AllAtOnce
CodeDeployDefault.HalfAtATime
CodeDeployDefault.OneAtATime

CodeDeployDefault.AllAtOnce

Attempts to deploy an application revision to as many instances as possible at once.

Using an example of nine instances, CodeDeployDefault.AllAtOnce attempts to deploy to all nine instances at once.

The status of the overall deployment is displayed as Succeeded if the application revision is deployed to one or more of the instances.

The status of the overall deployment is displayed as Failed if the application revision is not deployed to any of the instances

CodeDeployDefault.HalfAtATime

Deploys to up to half of the instances at a time (with fractions rounded down).

The overall deployment succeeds if the application revision is deployed to at least half of the instances (with fractions rounded up).

Otherwise, the deployment fails.

In the example of nine instances, it deploys to up to four instances at a time. The overall deployment succeeds if deployment to five or more instances succeed. Otherwise, the deployment fails.

CodeDeployDefault.OneAtATime

Deploys the application revision to only one instance at a time.

For deployment groups that contain more than one instance:

1. The overall deployment succeeds if the application revision is deployed to all of the instances.
2. The exception to this rule is that if deployment to the last instance fails, the overall deployment still succeeds.
3. The overall deployment fails as soon as the application revision fails to be deployed to any but the last instance.

ECS and Lambda Platform

If your deployment uses the AWS Lambda or the Amazon ECS compute platform, you can specify how traffic is routed to your updated Lambda function or ECS task set.

Deployment Configuration	Description
Canary	Traffic is shifted in two increments
Linear	Traffic is shifted in equal increments with an equal number of minutes between each increment.
All-At-Once	All traffic is shifted from the original Lambda function or ECS task set to the updated function or task set all at once.

Part 2 - AWS Lambda compute platform

Deployment configuration	Description
CodeDeployDefault.LambdaCanary10Percent5Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed five minutes later.
CodeDeployDefault.LambdaCanary10Percent10Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed 10 minutes later.
CodeDeployDefault.LambdaCanary10Percent15Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed 15 minutes later.
CodeDeployDefault.LambdaCanary10Percent30Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed 30 minutes later.
CodeDeployDefault.LambdaLinear10PercentEvery1Minute	Shifts 10 percent of traffic every minute until all traffic is shifted.
CodeDeployDefault.LambdaLinear10PercentEvery2Minutes	Shifts 10 percent of traffic every two minutes until all traffic is shifted.
CodeDeployDefault.LambdaLinear10PercentEvery3Minutes	Shifts 10 percent of traffic every three minutes until all traffic is shifted.
CodeDeployDefault.LambdaLinear10PercentEvery10Minutes	Shifts 10 percent of traffic every 10 minutes until all traffic is shifted.
CodeDeployDefault.LambdaAllAtOnce	Shifts all traffic to the updated Lambda functions at once.

Part 3 - ECS Compute Platform

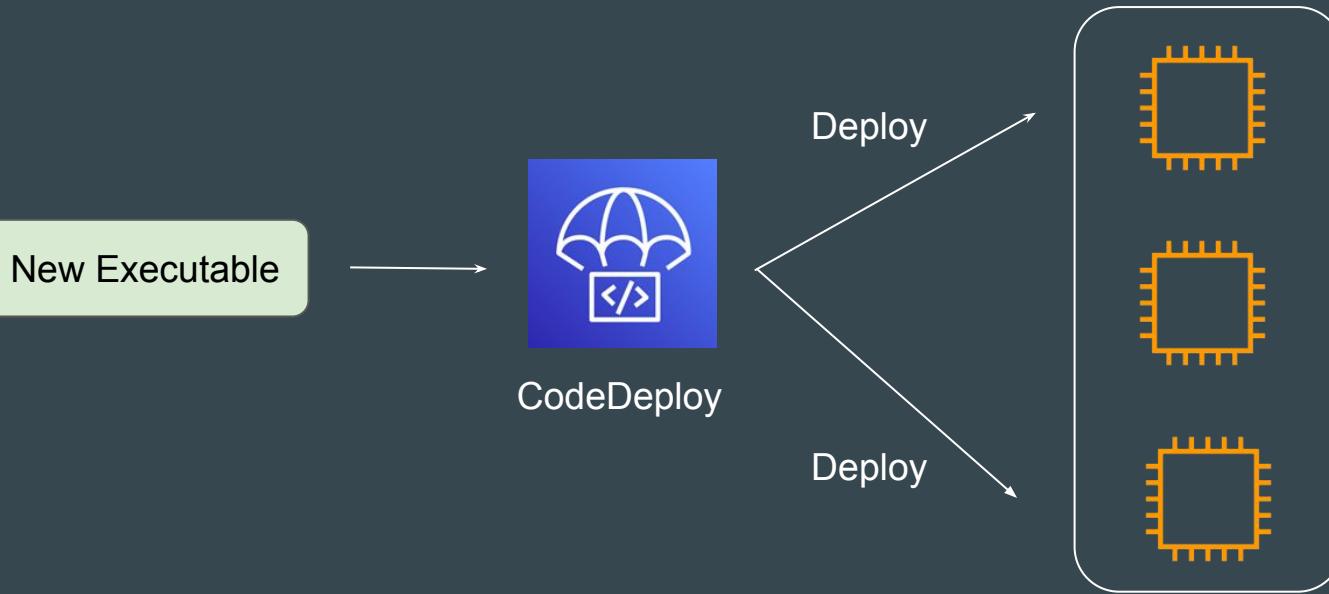
Deployment configuration	Description
CodeDeployDefault.ECSTrafficTypeWithHealthCheck	Shifts 10 percent of traffic every minute until all traffic is shifted.
CodeDeployDefault.ECSLinear10PercentEvery3Minutes	Shifts 10 percent of traffic every three minutes until all traffic is shifted.
CodeDeployDefault.ECSCanary10Percent5Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed five minutes later.
CodeDeployDefault.ECSCanary10Percent15Minutes	Shifts 10 percent of traffic in the first increment. The remaining 90 percent is deployed 15 minutes later.
CodeDeployDefault.ECSAllAtOnce	Shifts all traffic to the updated Amazon ECS container at once.

CodeDeploy - Deployment Types



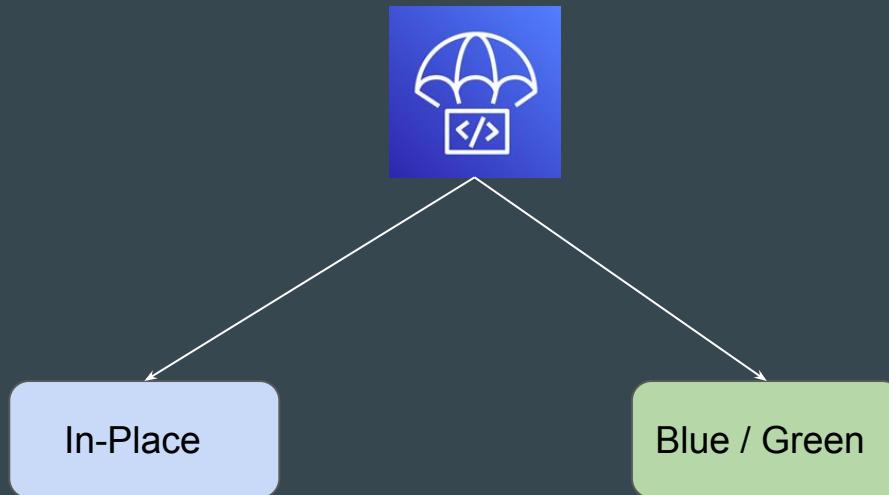
Understanding the Basics

A **deployment** is the process of installing content on one or more instances. This content can consist of code, web and configuration files, executables, packages, scripts, and so on



Basic Architecture

CodeDeploy provides two deployment type options, in-place deployments and blue/green deployments.



1 - In-Place Deployments

The application on each instance in the deployment group is stopped, the latest application revision is installed, and the new version of the application is started and validated.

You can use a load balancer so that each instance is deregistered during its deployment and then restored to service after the deployment is complete.

Only deployments that use the EC2/On-Premises compute platform can use in-place deployments

2 - Blue/green on an EC2/On-Premises compute platform

Instances are provisioned for the replacement environment.

The latest application revision is installed on the replacement instances.

Instances in the replacement environment are registered with an Elastic Load Balancing load balancer, causing traffic to be rerouted to them. Instances in the original environment are deregistered and can be terminated or kept running for other uses.

3 - Blue/green on an Lambda or Amazon ECS

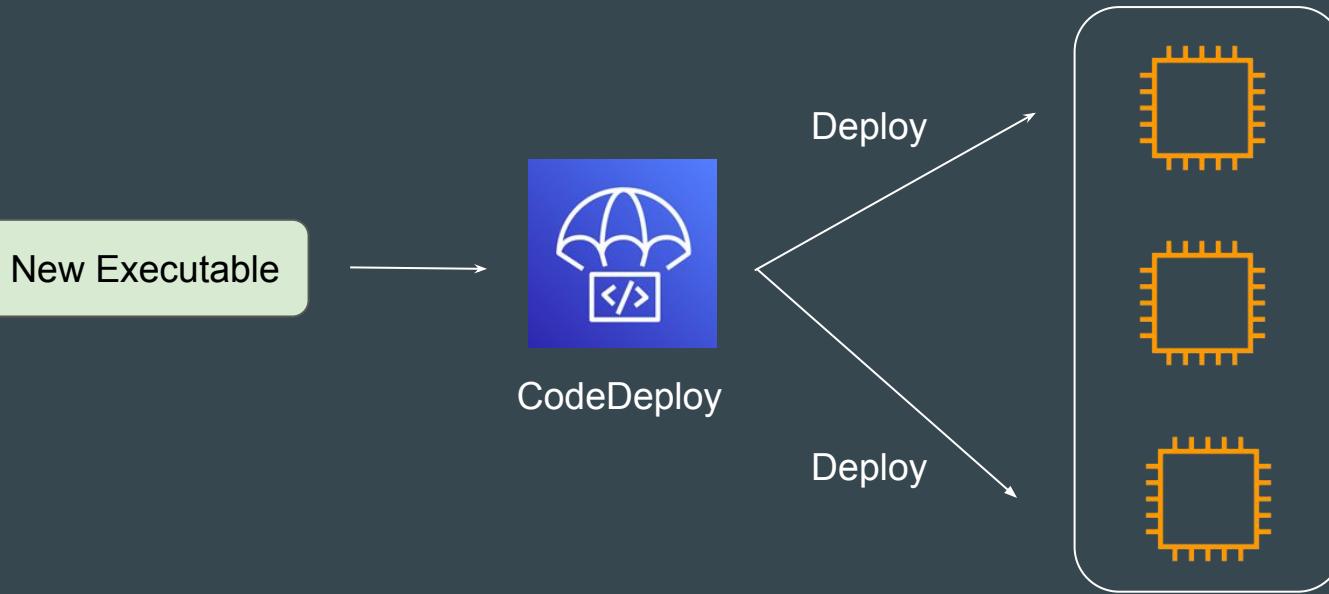
Traffic is shifted in increments according to a canary, linear, or all-at-once deployment configuration.

CodeDeploy - Deployment Types



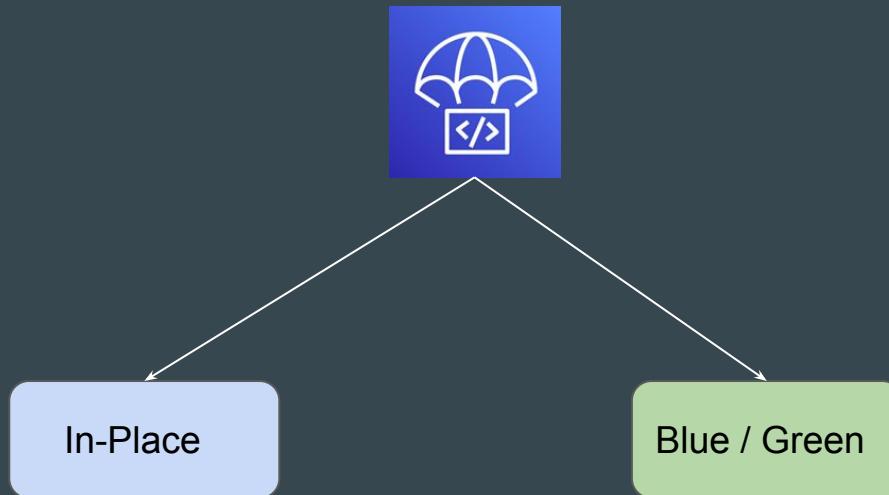
Understanding the Basics

A **deployment** is the process of installing content on one or more instances. This content can consist of code, web and configuration files, executables, packages, scripts, and so on



Basic Architecture

CodeDeploy provides two deployment type options, in-place deployments and blue/green deployments.



1 - In-Place Deployments

The application on each instance in the deployment group is stopped, the latest application revision is installed, and the new version of the application is started and validated.

You can use a load balancer so that each instance is deregistered during its deployment and then restored to service after the deployment is complete.

Only deployments that use the EC2/On-Premises compute platform can use in-place deployments

2 - Blue/green on an EC2/On-Premises compute platform

Instances are provisioned for the replacement environment.

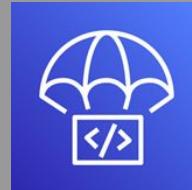
The latest application revision is installed on the replacement instances.

Instances in the replacement environment are registered with an Elastic Load Balancing load balancer, causing traffic to be rerouted to them. Instances in the original environment are deregistered and can be terminated or kept running for other uses.

3 - Blue/green on an Lambda or Amazon ECS

Traffic is shifted in increments according to a canary, linear, or all-at-once deployment configuration.

CodeDeploy - Deployment Settings



Setting the Base Right

CodeDeploy has created new Green environment with new version of application.

What now?



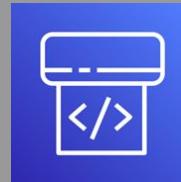
Blue Environment

Green Environment

Several Questions

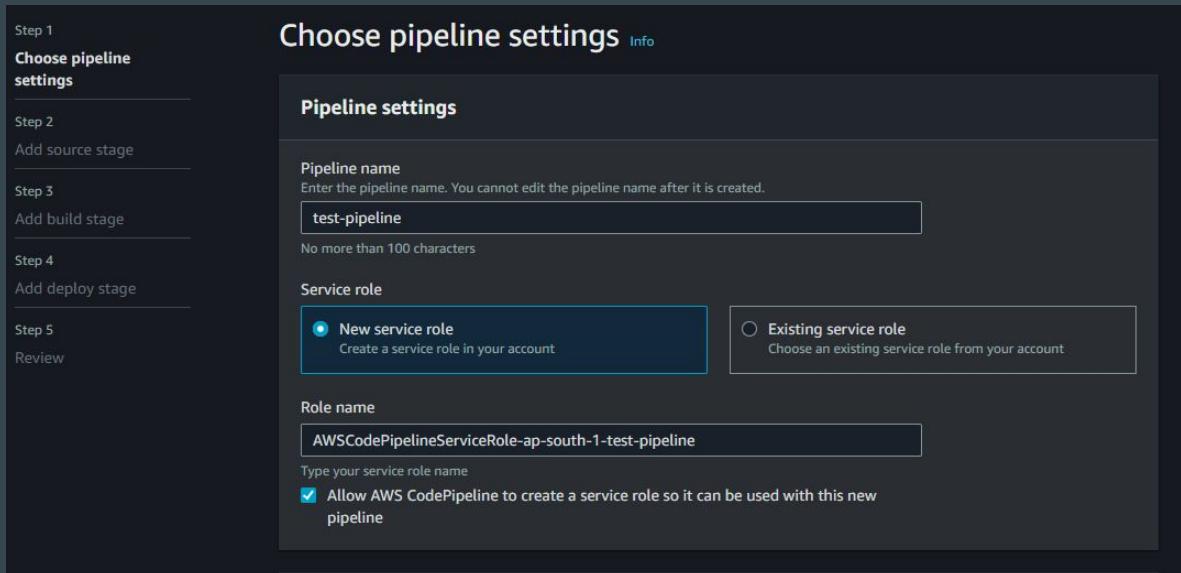
1. When will traffic be rerouted to the new Green environment?
2. What happens to the original instance in the deployment group?

CodePipeline - Service Role



Setting the Base

The **CodePipeline service role** is configured with one or more policies that control access to the AWS resources used by the pipeline.



Points to Note

Depending on the use-case, you can attach one or more policies to the codepipeline role

You can also want to attach a policy to a role when you configure cross-account access to your pipeline.

Remove Unnecessary Permissions

You can edit the service role statement to remove access to resources you do not use.

For example, if none of your pipelines include CodeDeploy, you can edit the policy statement to remove the section that grants access to CodeDeploy resources.

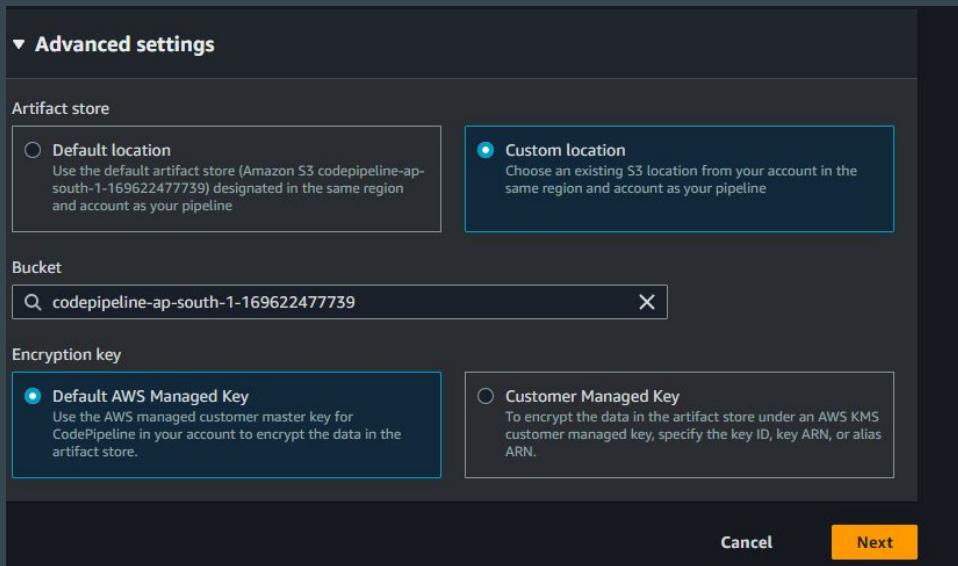
```
{
  "Action": [
    "codedeploy>CreateDeployment",
    "codedeploy>GetApplicationRevision",
    "codedeploy>GetDeployment",
    "codedeploy>GetDeploymentConfig",
    "codedeploy>RegisterApplicationRevision"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
```

CodePipeline - Artifact Encryption



Setting the Base

CodePipeline allows customers to **choose the encryption key** to encrypt the data in the artifact store.



Points to Note

If you are using the default S3 key, you cannot change or delete this AWS managed key.

If you are using a customer managed key in AWS KMS to encrypt or decrypt artifacts in the S3 bucket, you can change or rotate this customer managed key as necessary.

S3 Bucket Policy for Encryption

Amazon S3 supports bucket policies that you can use if you require server-side encryption for all objects that are stored in your bucket.

The following bucket policy denies upload object (s3:PutObject) permission to everyone if the request does not include the x-amz-server-side-encryption header requesting server-side encryption with SSE-KMS.

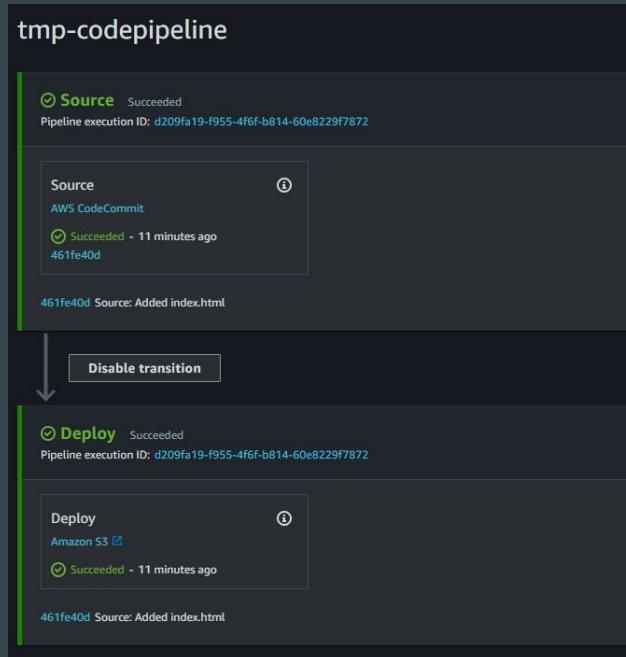
```
{
    "Version": "2012-10-17",
    "Id": "SSEAndSSLPolicy",
    "Statement": [
        {
            "Sid": "DenyUnEncryptedObjectUploads",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::codepipeline-us-west-2-89050EXAMPLE/*",
            "Condition": {
                "StringNotEquals": {
                    "s3:x-amz-server-side-encryption": "aws:kms"
                }
            }
        },
    ],
}
```

CodePipeline Stage Actions



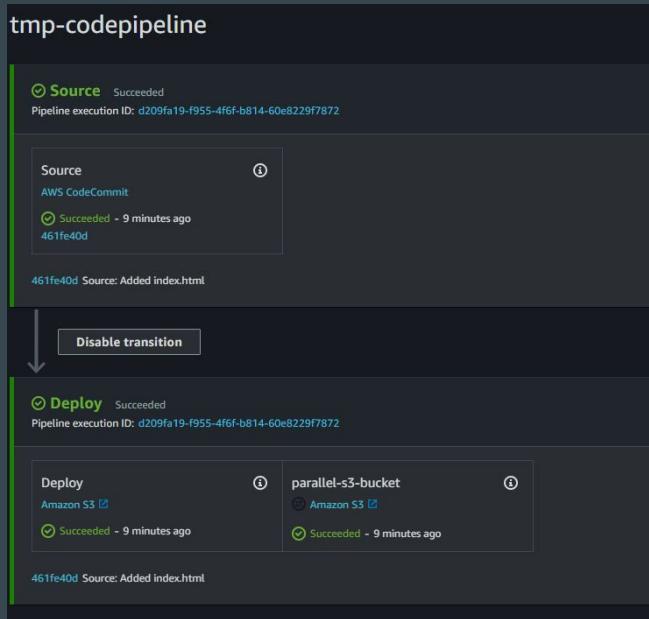
Setting the Base

CodePipeline can consist of set of sequential stages, each stage containing one or more actions.



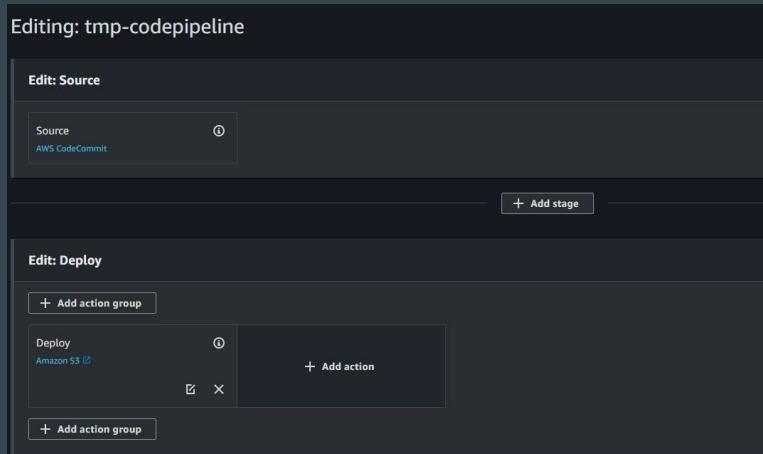
Stage Actions

Actions within a stage can be executed **sequentially**, according their run order **or** **in parallel**



Stage Actions

In the console, you can specify a serial sequence for an action by choosing **Add action group** at the level in the stage where you want it to run, or you can specify a parallel sequence by choosing **Add action**.



Setting runOrder

The default runOrder value for an action is 1.

To specify parallel actions, use the same integer for each action you want to run in parallel

```
[  
  {  
    "inputArtifacts": [  
      An input artifact structure, if supported for the action category  
    ],  
    "name": "ActionName",  
    "region": "Region",  
    "namespace": "source_namespace",  
    "actionTypeId": {  
      "category": "An action category",  
      "owner": "AWS",  
      "version": "1"  
      "provider": "A provider type for the action category",  
    },  
    "outputArtifacts": [  
      An output artifact structure, if supported for the action category  
    ],  
    "configuration": {  
      Configuration details appropriate to the provider type  
    },  
    "runOrder": A positive integer that indicates the run order within the stage,  
  }  
]
```

runOrder Examples

If you want three actions to run in sequence in a stage, you would give the first action the runOrder value of 1, the second action the runOrder value of 2, and the third the runOrder value of 3

If you want the second and third actions to run in parallel, you would give the first action the runOrder value of 1 and both the second and third actions the runOrder value of 2.

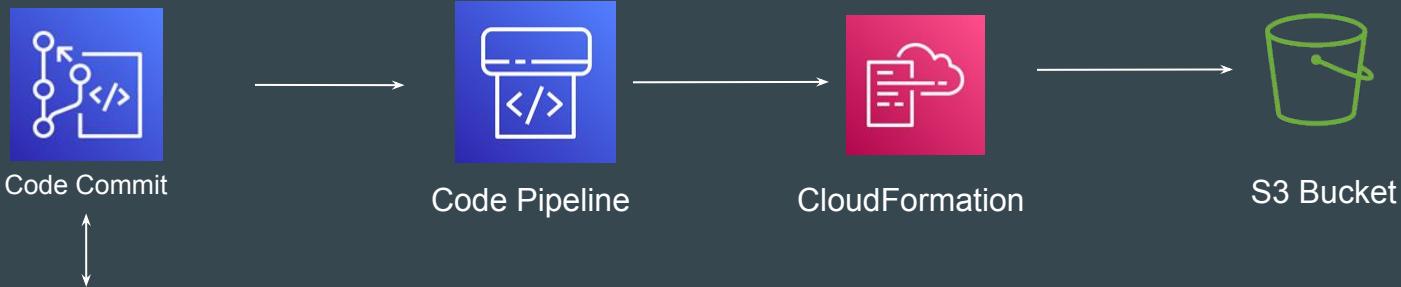
Deploying CloudFormation through CodePipeline



Setting the Base

Many organization uses CloudFormation template and manage resources.

You can integrate CloudFormation with CodePipeline for easier management.



```
Resources:  
S3Bucket:  
  Type: 'AWS::S3::Bucket'  
  DeletionPolicy: Retain  
  Properties:  
    BucketName: DOC-EXAMPLE-BUCKET
```

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
[Choose pipeline settings](#)

Step 2
[Add source stage](#)

Step 3
[Add build stage](#)

Step 4
[Add deploy stage](#)

Step 5
Review

Add deploy stage Info



You cannot skip this stage

Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

Deploy

Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.



- AWS AppConfig
- AWS CloudFormation
- AWS CloudFormation Stack Set
- AWS CodeDeploy
- AWS Elastic Beanstalk
- AWS OpsWorks Stacks
- AWS S3PublishDev
- AWS Service Catalog
- Amazon ECS
- Amazon ECS (Blue/Green)
- Amazon S3

[Previous](#)

[Next](#)

Points to Note

CodePipeline must have necessary permissions to invoke CloudFormation.

CloudFormation must have enough permissions to create resources defined in the CodeCommit repository code.

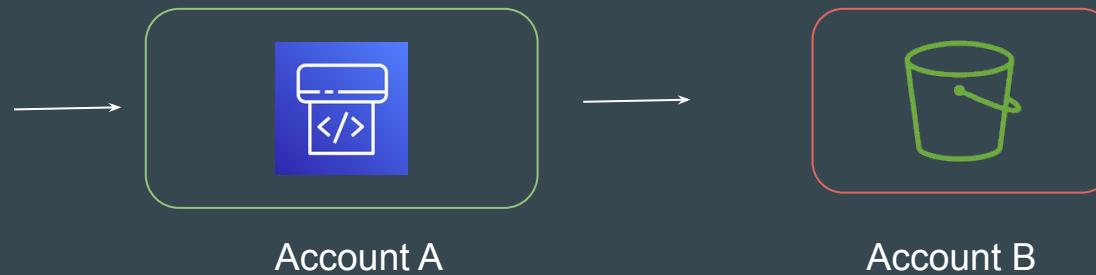
Cross-Account Pipeline Structure



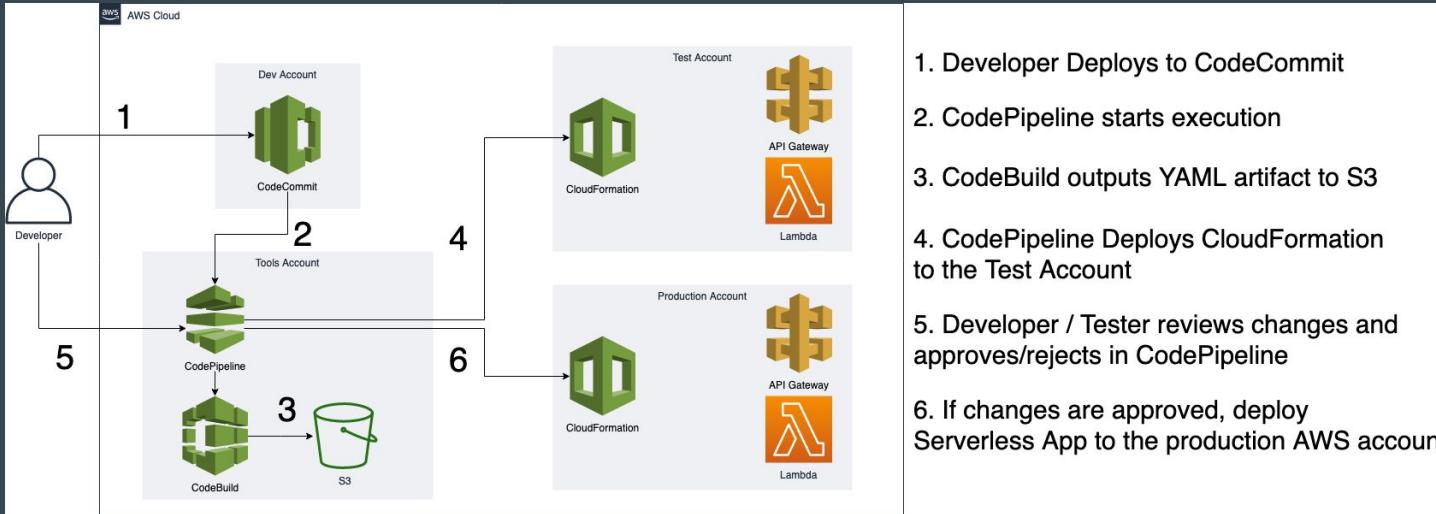
Setting the Base

There can be a requirement where CodePipeline in Account A wants to deploy some resources in Account B

```
Resources:  
  S3Bucket:  
    Type: 'AWS::S3::Bucket'  
    DeletionPolicy: Retain  
    Properties:  
      BucketName: DOC-EXAMPLE-BUCKET
```

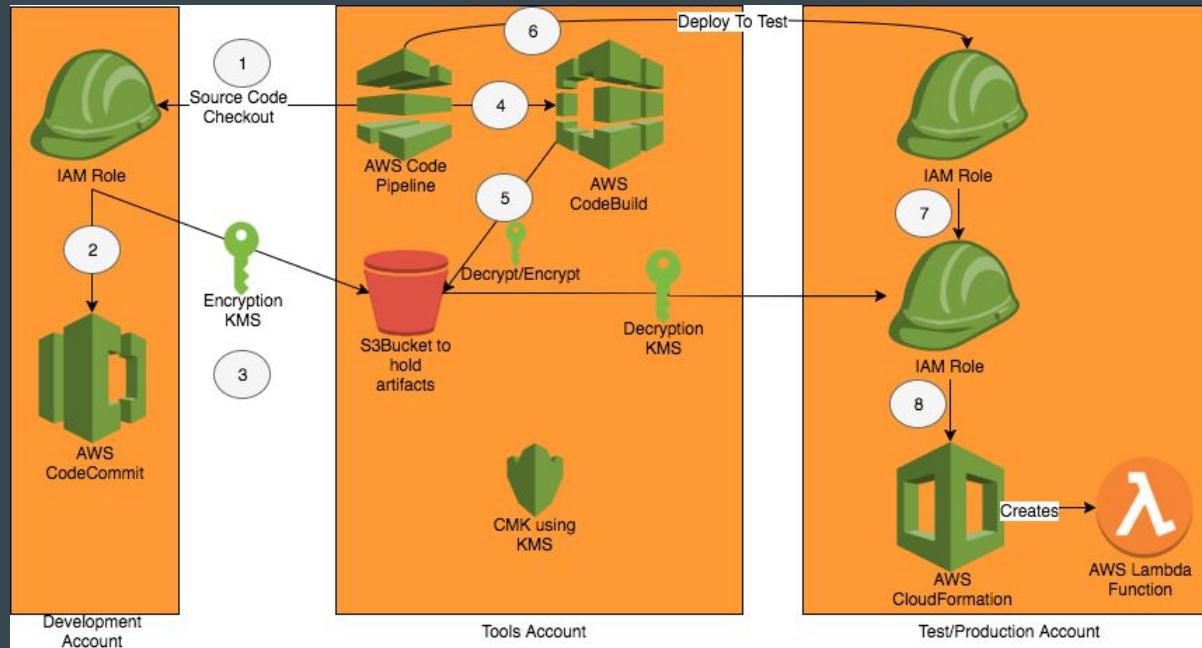


Cross-Account Architecture



1. Developer Deploys to CodeCommit
2. CodePipeline starts execution
3. CodeBuild outputs YAML artifact to S3
4. CodePipeline Deploys CloudFormation to the Test Account
5. Developer / Tester reviews changes and approves/rejects in CodePipeline
6. If changes are approved, deploy Serverless App to the production AWS account

Reference WorkFlow



Points to Note

1. Ensure that you create a new CMK instead of using AWS Managed Key.
2. The CMK must have appropriate key policy to allow access from other accounts for decryption.
3. S3 Bucket must have appropriate bucket policy to allow other accounts to fetch the artifacts.
4. Service Role for other accounts must have permissions to get object from S3 and decrypt using KMS.

Sample KMS Policy

Key policy

```
1 {
2     "Version": "2012-10-17",
3     "Id": "key-consolepolicy-3",
4     "Statement": [
5         {
6             "Sid": "Enable IAM User Permissions",
7             "Effect": "Allow",
8             "Principal": {
9                 "AWS": "arn:aws:iam::004417287555:root" ←
10            },
11            "Action": "kms:*",
12            "Resource": "*"
13        },
14    ]
15}
```

This is Account ID of Another AWS account.

Sample S3 Bucket Policy

Policy

```
1 [ {  
2     "Version": "2012-10-17",  
3     "Id": "Policy1553183091390",  
4     "Statement": [  
5         {  
6             "Sid": "",  
7             "Effect": "Allow",  
8             "Principal": {  
9                 "AWS": "arn:aws:iam::004417287555:root" ←  
10            },  
11            "Action": [  
12                "s3:Get*",  
13                "s3:Put*"  
14            ],  
15            "Resource": "arn:aws:s3:::kplabs-cloudformation-pipeline/*"  
16        },  
17        {  
18            "Sid": "",  
19            "Effect": "Allow",  
20            "Principal": {  
21                "AWS": "arn:aws:iam::004417287555:root"  
22            },  
23            "Action": "s3>ListBucket",  
24            "Resource": "arn:aws:s3:::kplabs-cloudformation-pipeline"  
25        }  
26    ] }
```

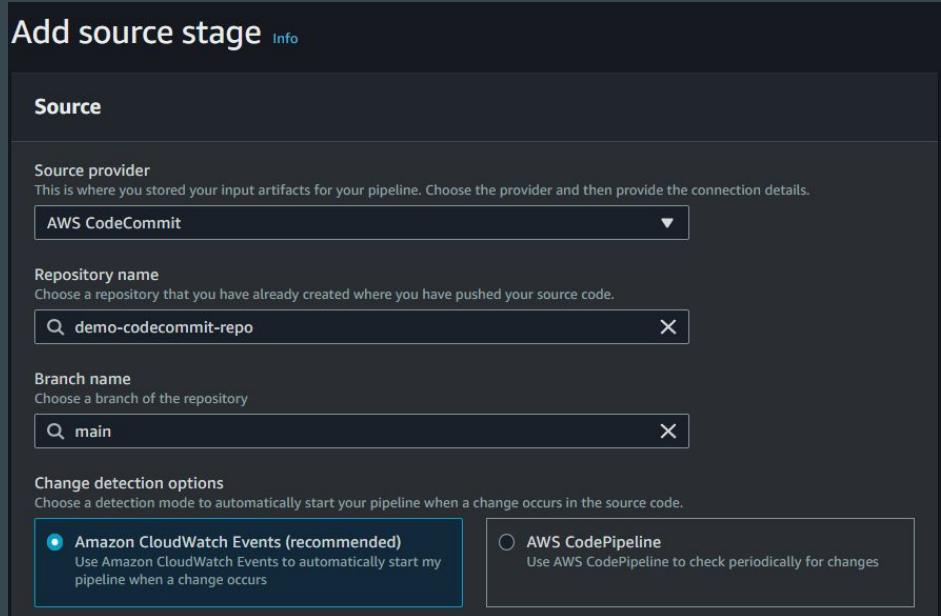
This is Account ID of Another AWS account.

CodePipeline - Change Detection Options



Setting the Base

CodePipeline Detection Mode allows customers to automatically start pipeline when a change occurs in the source code.



Reference Screenshot - EventBridge Rule

codepipeline-democo-main-139210-rule

Edit Disable Delete CloudFormation Template ▾

Rule details Info

Rule name codepipeline-democo-main-139210-rule	Status Enabled	Event bus name default	Type Standard
Description Amazon CloudWatch Events rule to automatically start your pipeline when a change occurs in the AWS CodeCommit source repository and branch. Deleting this may prevent changes from being detected in that pipeline. Read more: http://docs.aws.amazon.com/codepipeline/latest/userguide/pipelines-about-starting.html	Rule ARN arn:aws:events:ap-southeast-1:042025557788:rule/codepipeline-democo-main-139210-rule	Event bus ARN arn:aws:events:ap-southeast-1:042025557788:event-bus/default	

Event pattern Info Edit

```
1 {
2   "source": ["aws.codecommit"],
3   "detail-type": ["CodeCommit Repository State Change"],
4   "resources": ["arn:aws:codecommit:ap-southeast-1:042025557788:demo-codecommit-repo"],
5   "detail": {
6     "event": ["referenceCreated", "referenceUpdated"],
7     "referenceType": ["branch"],
8     "referenceName": ["main"]
9   }
10 }
```

Points to Note

The recommended event-based change detection method for pipelines is determined by the pipeline source, such as CodeCommit.

Pipeline source	Recommended event-based detection method
AWS CodeCommit	Amazon CloudWatch Events (recommended).
Amazon S3	Amazon CloudWatch Events (recommended) and an AWS CloudTrail trail.
GitHub version 1	Connections

Points to Note

There are two supported ways to start your pipeline upon a code change:

1. Event-based change detection.
2. Polling.

AWS recommends using event-based change detection for pipelines.

AWS CodeArtifact



Revising the Basics of Artifact

At a high-level overview, Artifacts are the binaries that are deployed during the deployment

The main aim of an artifact is to be downloaded as fast as possible on a server, and run immediately, with no service interruption.



Points to Note

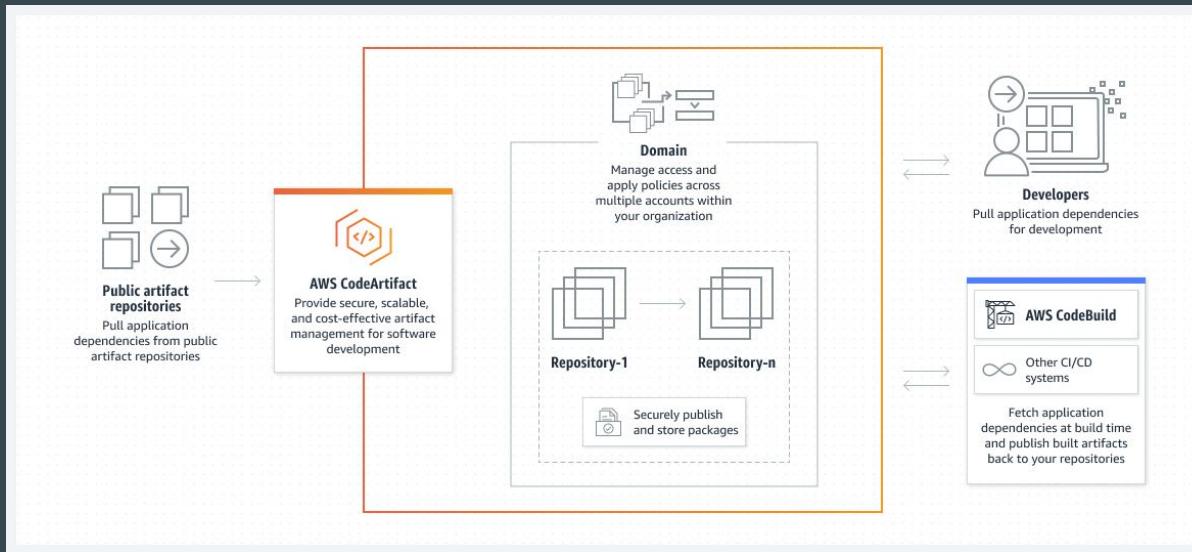
Developer needs a central place in which artifacts are stored.

Development team can also rely on on various open source software packages available in public repositories.

There is a **requirement of a central repository** to store and manage artifacts built by organization AND various 3rd party software packages from public repositories.

AWS CodeArtifact

AWS CodeArtifact is a secure, highly scalable, managed artifact repository service that helps organizations to **store and share software packages** for application development.

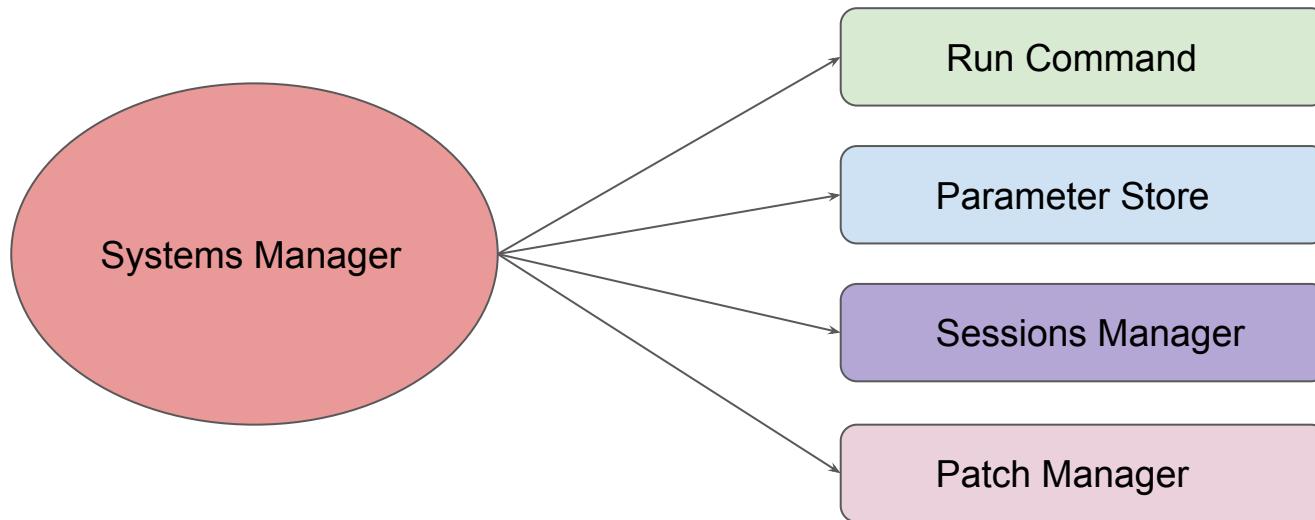


AWS Systems Manager

Interesting Set of Services

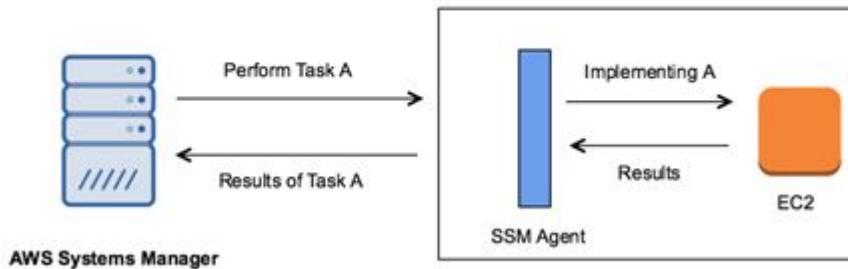
Overview of Systems Manager

AWS Systems Manager is a group of services which allows customers to have a better visibility and control of the infrastructure.



High Level Overview

The basic idea behind the " Systems Manager" is that there will be an SSM agent installed in the EC2 instances, and the customer can provide specific tasks to the installed agent from the systems manager console.

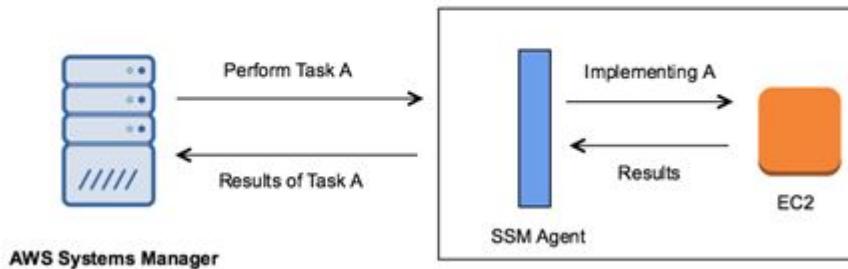


Configuring SSM Agent

Systems Manager Agent

High Level Overview

The basic idea behind the " Systems Manager" is that there will be an SSM agent installed in the EC2 instances, and the customer can provide specific tasks to the installed agent from the systems manager console.



Overview of the SSM Agent

AWS Systems Manager Agent (SSM Agent) is Amazon software that can be installed and configured on an Amazon EC2 instance, an on-premises server, or a virtual machine (VM).

SSM Agent is preinstalled, by default, on the following Amazon Machine Images (AMIs):

- Amazon Linux
- Amazon Linux 2
- Ubuntu Server 16.04, 18.04, and 20.04
- Amazon Linux 2 ECS-Optimized Base AMIs

Required Permissions

By default, AWS Systems Manager doesn't have permission to perform actions on your instances

You need to attach IAM role with [AmazonSSMManagedInstanceCore](#) policy to allow an instance to use Systems Manager service core functionality.

Systems Manager - Sessions Manager

Interesting Set of Services

Overview of Sessions Manager

Sessions Manager allows customers to connect to the instances through an interactive one-click browser-based shell or through the AWS CLI.



Difference Between EC2 Connect & Sessions Manager

	EC2 Connect	Sessions Manager
IAM Role for EC2	Not Required	Required
Security Group (22)	Required	Not Required
Public IP	Required	Not Required

Benefits of Sessions Manager

Some of the notable benefits of Sessions Manager are as follows:

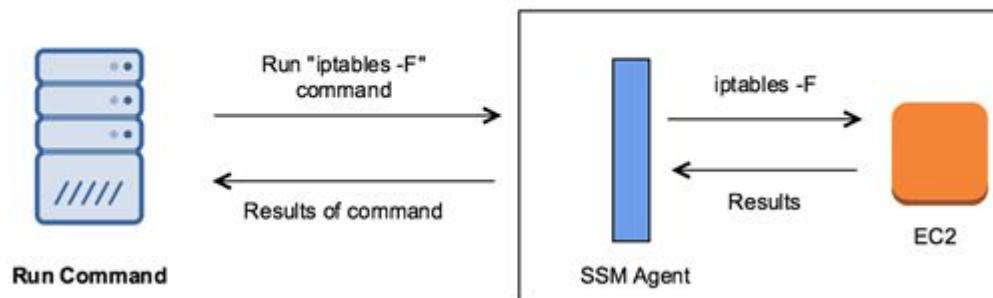
- Centralized Access Control using IAM Policies
- No Inbound Ports Needs to be Open
- Logging and auditing session activity
- One-click access to instances from the console and CLI
- No need of VPN to connect to instances.

Systems Manager - Run Command

Running Commands Remotely

Overview of Run Command

Run Command, as the name suggests allows us to run specific commands in the instances where SSM agent is installed.



Document Feature

Run Command provides much more granular features because of its “command document” feature.

There are various command document available that can perform certain ready-made actions.

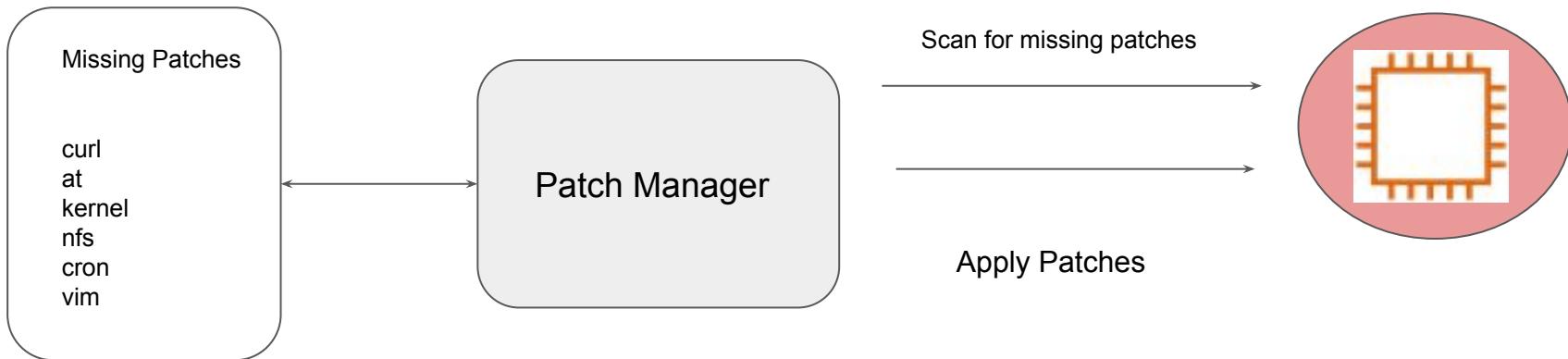
- AWS-RunAnsiblePlaybook
- AWS-ConfigureDocker
- AWS-InstallMissingWindowsUpdates
- AWS-RunShellScript

Systems Manager - Patch Manager

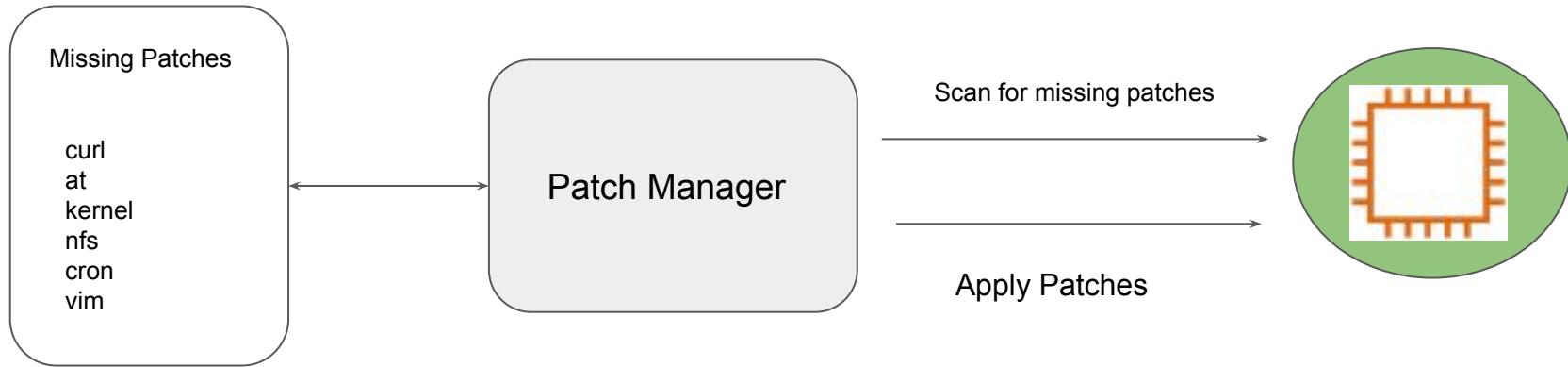
Interesting Set of Services

Overview of Patch Manager

Patch Manager automates the process of patching managed instances with both security related and other types of updates.



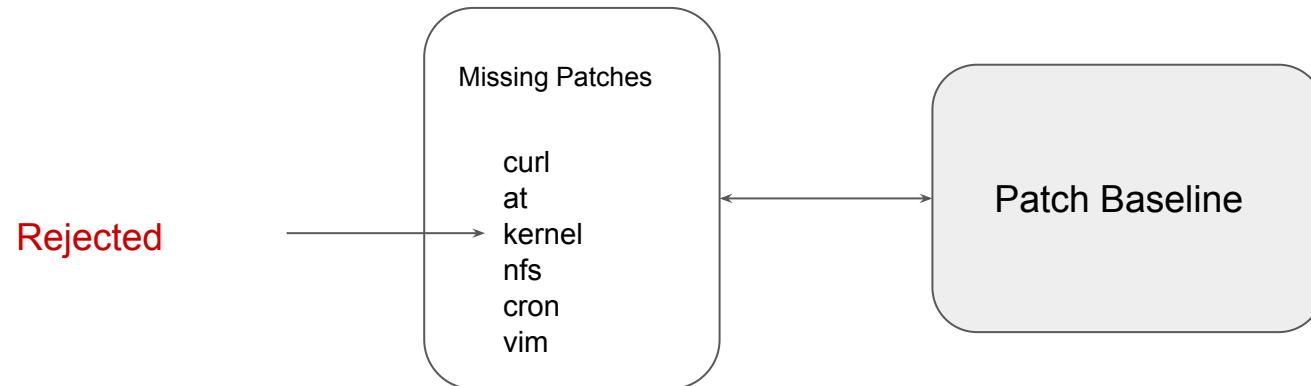
After Patching



Patch Baseline

Patch Baseline service determines the list of missing patches that need to be installed in the EC2 instance.

A patch baseline defines which patches are approved for installation on your instances. You can specify approved or rejected patches one by one.



Maintenance Window

Maintenance Window provides a mechanism for scheduling a particular activity on the specific target.

Example: Perform Patching activity at 2 AM in the morning.

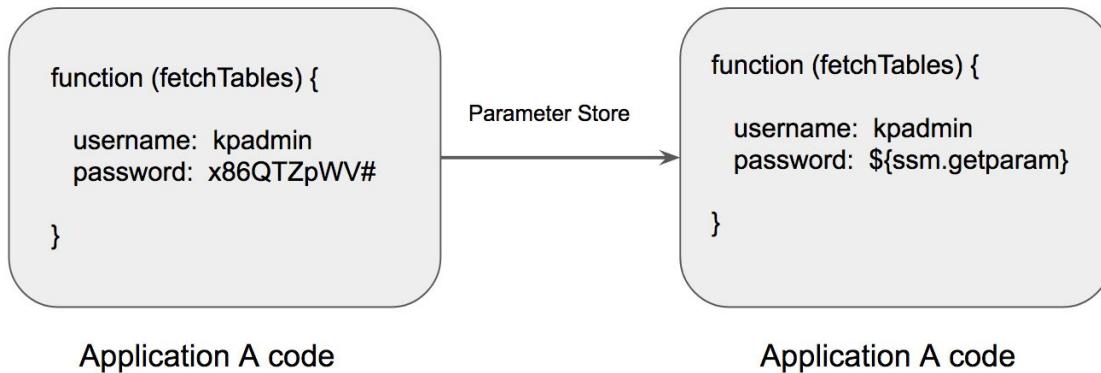


Systems Manager - Parameter Store

Not hardcoding the secrets

Getting the basics right

AWS Systems Manager Parameter Store provides centralized store to manage the configuration data, whether it is a plain-text data like database strings or even secrets such as passwords.



SSM - Automation

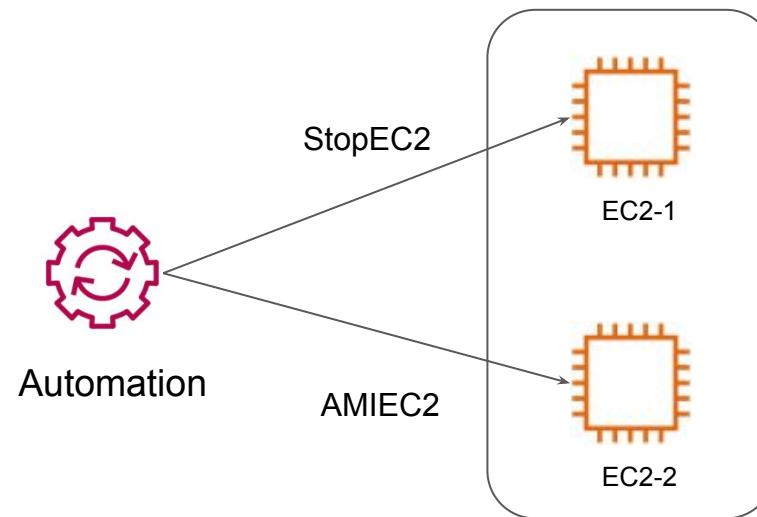
Automate Everything

Overview of SSM Automation

Automation, a capability of AWS Systems Manager, simplifies common maintenance and deployment tasks Amazon EC2 instances and other AWS resources.

Example Automation Tasks:

- Attach IAM to EC2 Instances
- Create AMI of Instances
- Perform Patching Activities



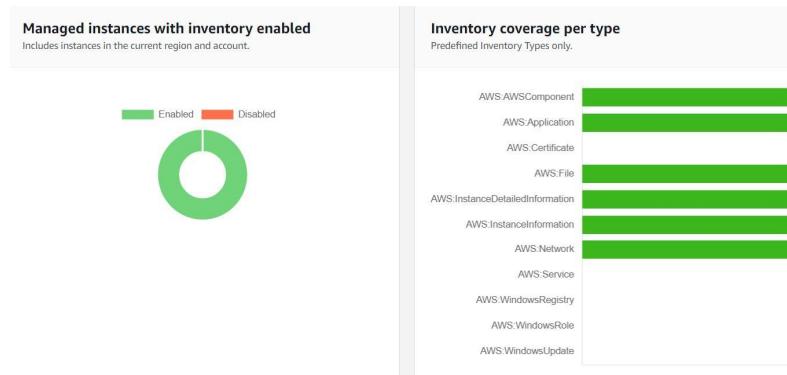
SSM - Inventory

Automate Everything

Overview of SSM Inventory

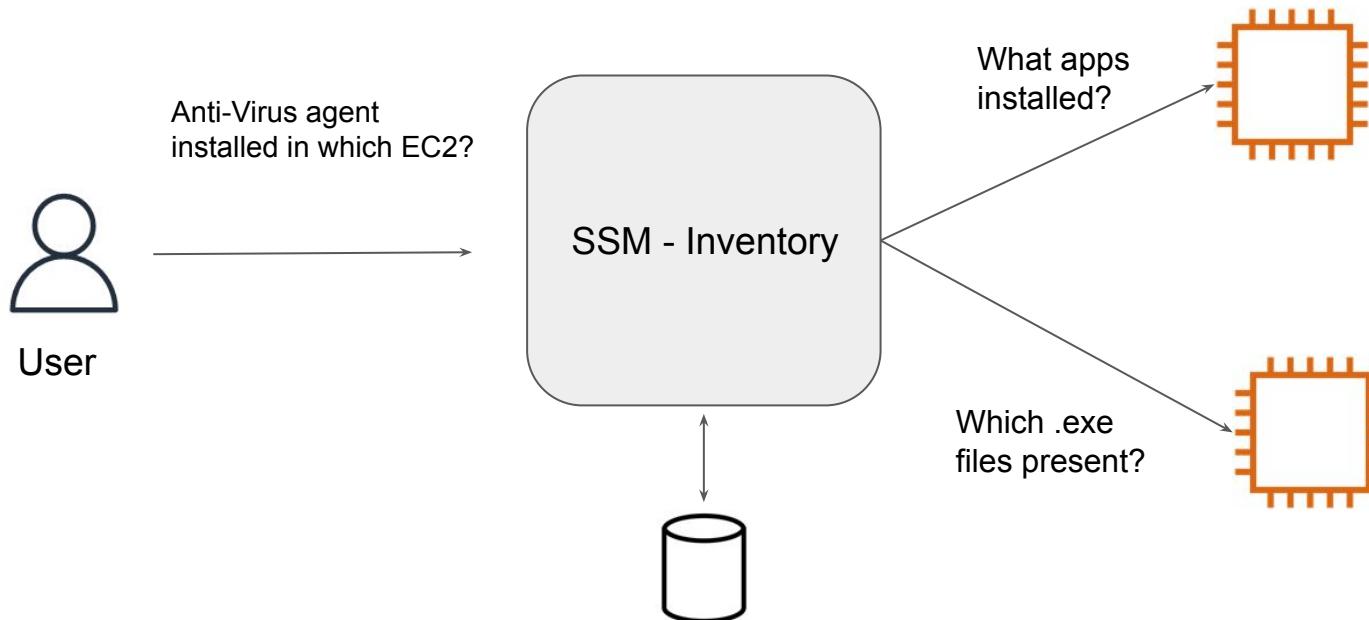
AWS Systems Manager Inventory provides visibility into your Amazon EC2 and on-premises computing environment.

It can capture various informations like Application Names, Files, Network Configuration, Instance Details, Windows Registry and others.

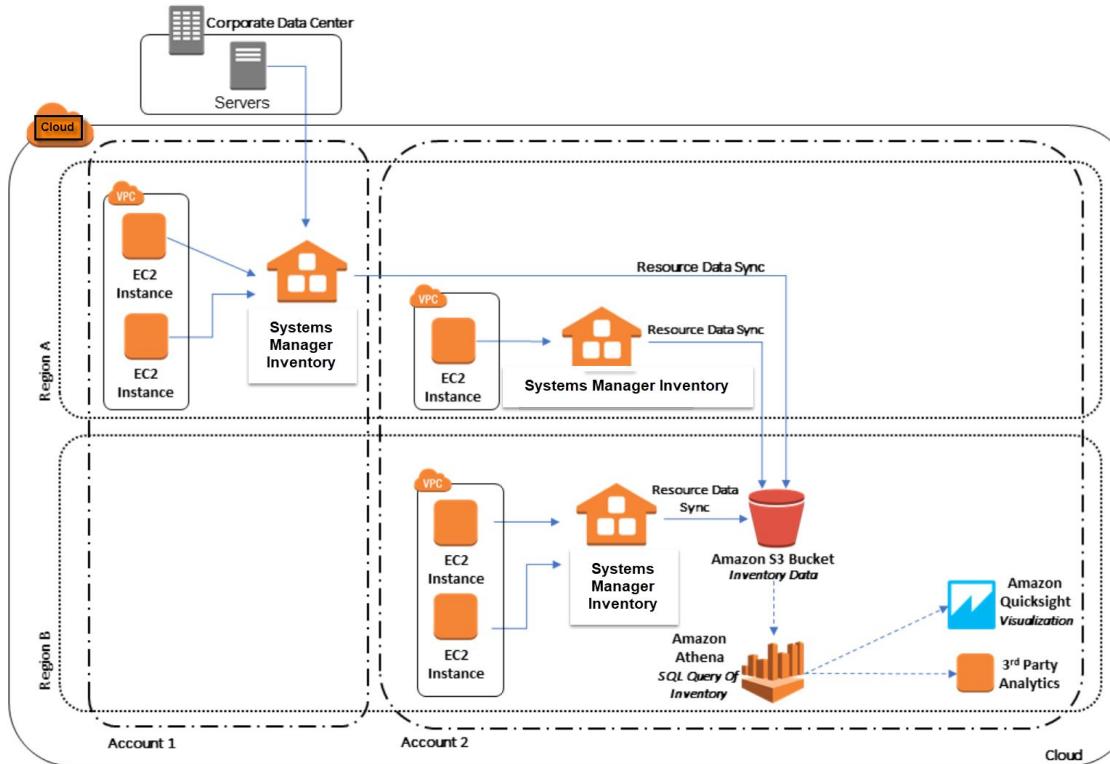


Overview of SSM Inventory

Administrator can run various queries to search for specific data based on the use-case.



Centralized Architecture



Systems Manager for Hybrid Environment

SSM is Awesome!



Hybrid Architecture

Systems Manager works well even for Hybrid Environments.

A hybrid environment includes on-premises servers and virtual machines (VMs) that have been configured for use with Systems Manager, including VMs in other cloud environments.

Once connected, you will be able to see your VM within the managed instances.

Standard vs Advanced Tier

SSM is Awesome!

Overview of Systems Manager Tier

AWS Systems Manager offers a standard-instances tier and an advanced-instances tier for servers and VMs in your hybrid environment.

Standard tier enables you to register a maximum of 1,000 on-premises servers or VMs per AWS account per AWS Region.

If you need to register more than 1,000 on-premises servers or VMs in a single account and Region, then use the advanced-instances tier.

Pricing Aspects

All instances configured for Systems Manager using the managed-instance activation for a Hybrid Environment are made available on a pay-per-use basis

For advanced tier, a charge of \$0.00695 per advanced on-premises instance per hour is applicable.

Infrastructure as Code

DevOps = Developers

Understanding the Basics

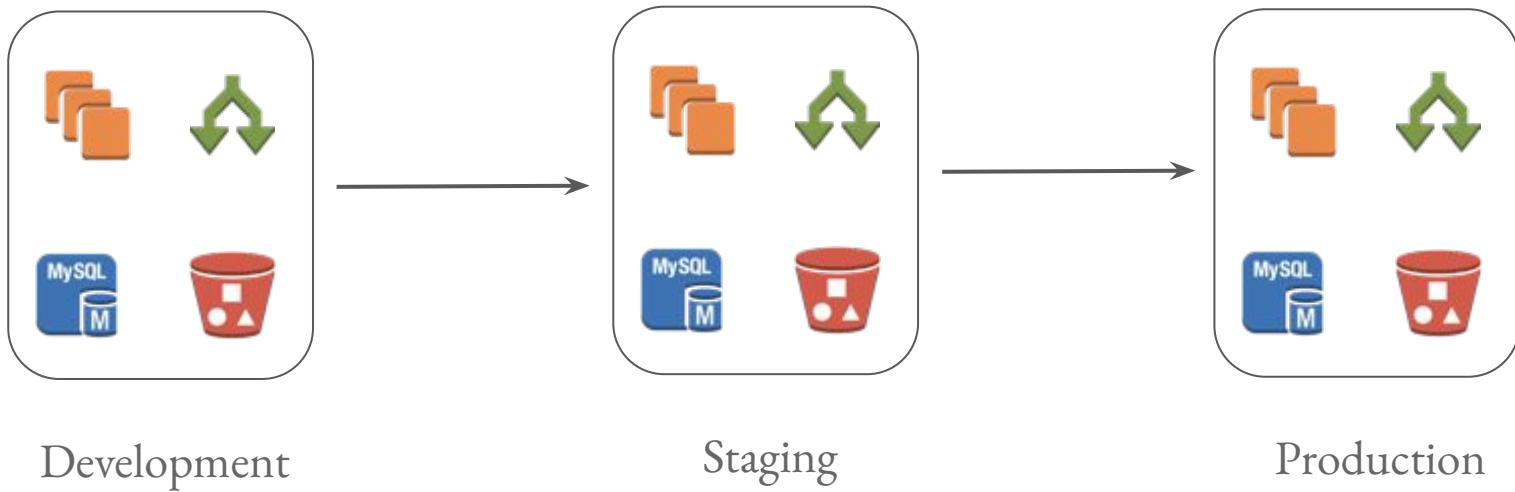
There are two ways in which you can build your infrastructure:

- Manually creating the infrastructure.
- Through Automation.

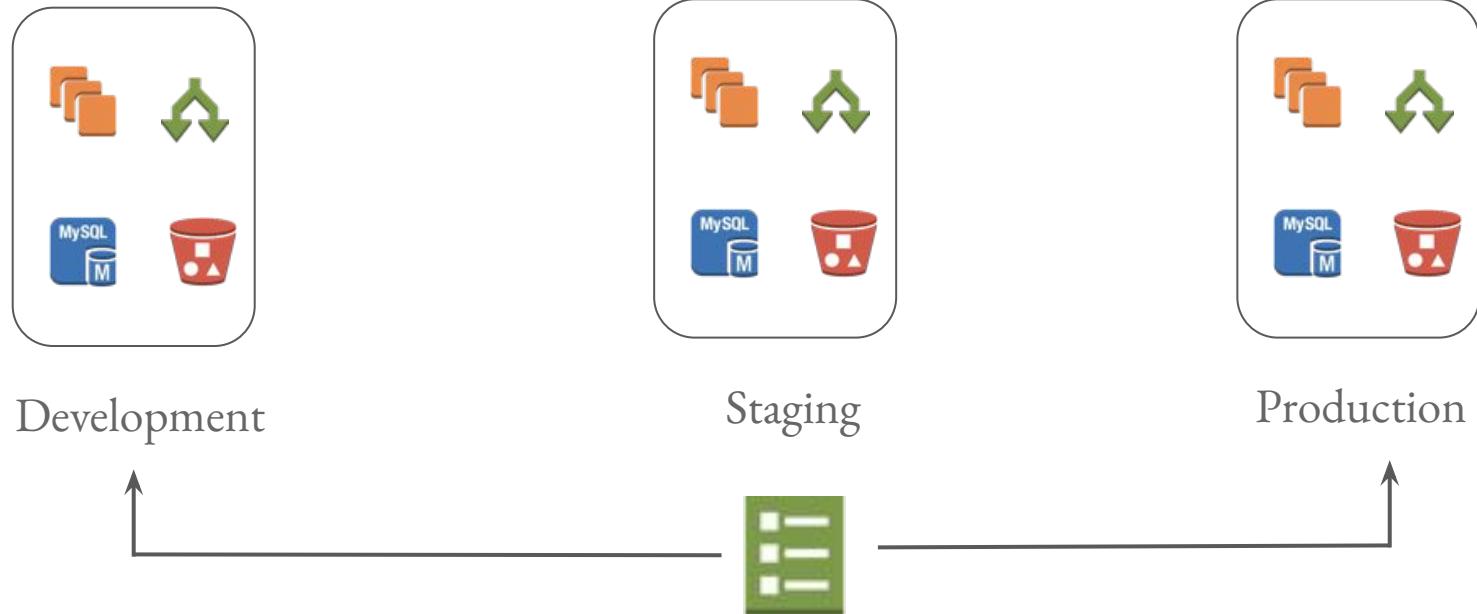
One of the major benefits of automation is the code usability.



Example of single service



Using IAAC



Benefits of Infrastructure as a Code

There are several benefits of designing your infrastructure as code:

Reusable Code

Managing infrastructure via source control.

Enable collaboration



CloudFormation - VPC

Let's Automate

Getting started

We will look into a minimal template for deployment of VPC via CloudFormation.

Simple VPC Template

```
1 AWSTemplateFormatVersion: "2010-09-09"
2 Description: VPC in North Virginia
3 Resources:
4   MyVPC:
5     Type: AWS::EC2::VPC
6     Properties:
7       CidrBlock: "10.77.0.0/16"
8       InstanceTenancy: default
9       Tags:
10      - Key: Name
11        Value: CFVPC
12      - Key: Environment
13        Value: Demo
14
```

Getting started

AWSTemplateFormatVersion specifies the version of template being used.

Currently 2010-09-09 is the only valid value that can be associated.

All the resource you create goes inside the “**Resource**” section of the template.

CloudFormation - Stack Dependencies

Deep Dive into CloudFormation

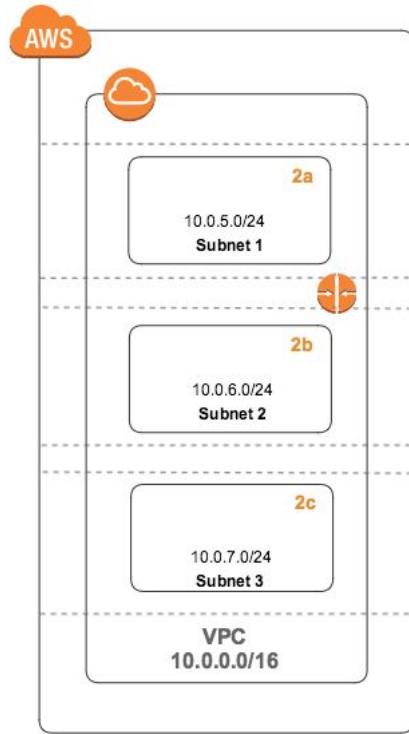
Understanding Stack Dependencies

In the previous video we had created a simple VPC.

However having just a VPC is not of great use.

We need Subnets, Internet Gateways, NAT Gateways and others.

However while defining subnet, we need to provide the VPC ID.



Main Route Table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-eb72s

Dependencies Pointer

If the properties of resource A use a !Ref to resource B, the following rule apply:

- Resource B is created before resource A.
- Resource A is deleted before resource B.

CloudFormation - DependsOn Attribute

Deep Dive into CloudFormation

Dependencies Pointer

If the properties of resource A use a !Ref to resource B, the following rule apply:

- Resource B is created before resource A.
- Resource A is deleted before resource B.

Let's Understand with Use-Case

Sample Use-Case:

For Application 1, you need 3 resource to be created:

- EC2 Instance
- RDS
- S3 Bucket.

Note: Application inside EC2 instance won't get initialized if RDS Instance is not ready.

CloudFormation - Errors and Rollbacks

Understanding CloudFormation

Getting started

There are two types of error that can occur within a CloudFormation template.

- i) Validation Error
- ii) Semantics Error / Post API Call Error

The **Validation error** occurs when CloudFormation cannot parse the template.

Semantic error is not detected until the resource is being created or being updated.

CloudFormation - Change Sets

Deep Dive into CloudFormation

Understanding Change Sets

We normally edit the templates and do CloudFormation UpdateStack operation to activate the changes.

It is important to have an additional insight into the changes that CloudFormation is planning to perform when it updates a stack

This should then allow us to be able to preview the changes, verify that they are in line with their expectations, and proceed with the update.

Change Sets

Change Sets allows us to submit the create a change set by submitting changes against the stack you want to update.

CloudFormation compares the stack to the new template and/or parameter values and produces a change set that you can review and then choose to apply

Changes (2)						
<input type="text"/> Search changes						
Action	Logical ID	Physical ID	Resource type	Replacement		
Add	MySubnet	-	AWS::EC2::Subnet	-		
Remove	MySubnet2c	subnet-03d07fda6b5a53761	AWS::EC2::Subnet	-		

CloudFormation - Parameters

Deep Dive into CloudFormation

Getting Started

Parameters in CloudFormation enable you to input custom values to your template each time you create or update a stack.

Let's understand with a use-case:

- We have created a CloudFormation template which will create an EC2 instance.
- Anyone within organization who wants to launch EC2 should use the template.

Problem:

- Template has a hard-coded value of m5.large and it needs constant modification.

Defining Parameter in Template

In following example, we define an **InstanceTypeParameter**.

This allows users to specify the Amazon EC2 instance type for the stack to use when you create or update the stack.

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

InstanceTypeParameter

Enter t2.micro, m1.small, or m1.large. Default is t2.micro.

t2.micro	▼
t2.micro	
m1.small	
m1.large	

Deletion Policy Attribute

Managed is better

CloudFormation

In CloudFormation, when we delete the stack, all the resources created through that stack will be deleted.

In certain case, we want that certain resources should not be deleted OR snapshot must be taken before resources are terminated.

This is controlled with help of “Deletion Policy Attribute” in cloudformation.



Introduction

Deletion policy attribute is allows us to preserve or (in some case) backup a resource when it's stack is deleted.

- We must specify deletion policy attribute for each resource that we want to control.
- If no such attribute is defined, cloudformation deletes the resources by default.

```
{  
    "AWSTemplateFormatVersion" : "2010-09-09",  
    "Resources" : {  
        "myS3Bucket" : {  
            "Type" : "AWS::S3::Bucket",  
            "DeletionPolicy" : "Retain"  
        }  
    }  
}
```

Deletion Policy

There are two options when we use deletion policy attributes

- i) Retain : CloudFormation keeps the resource without deleting it.
- ii) Snapshot : CloudFormation creates snapshot of the resource before deleting it.

CloudFormation - StackSets

Need to learn the backend

Getting Started

CloudFormation StackSets basically allows us to deploy stacks across multiple AWS account / AWS regions from single location.

Simple Use-Case:

- AWS Config is recommended to be enabled in all regions.
- Before we had to maintain stack across each region.
- This can now be solved easily using Stack Sets

Deployment Instruction

Two IAM Roles required:

1 for the Administrator Account of StackSets

1 for the Destination AWS Accounts.

Role Name for Admin Account: AWSCloudFormationStackSetAdministrationRole

Role Name for Dest Account: AWSCloudFormationStackSetExecutionRole

CloudFormation - Nested Stacks

Need to learn the backend

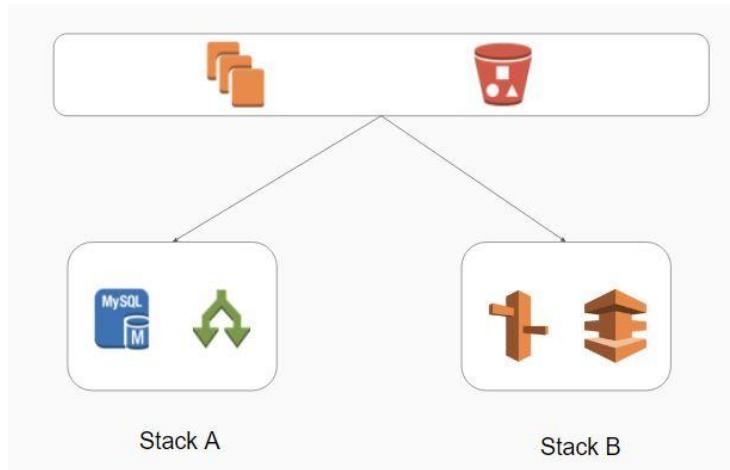
Understanding Common Components

As your stack grows, the number of common components being used also grows.



Nested Architecture

Instead of we copy-pasting the same component, we can create a dedicated template with those components defined and reference them from the other template, called nested stacks.



Creation Policy Attributes

Need to learn the backend

Understanding the Challenge

CloudFormation will generally mark the resource status as “Completed” when the resource like EC2 instance is created.

However resource creation does not mean resource readiness.

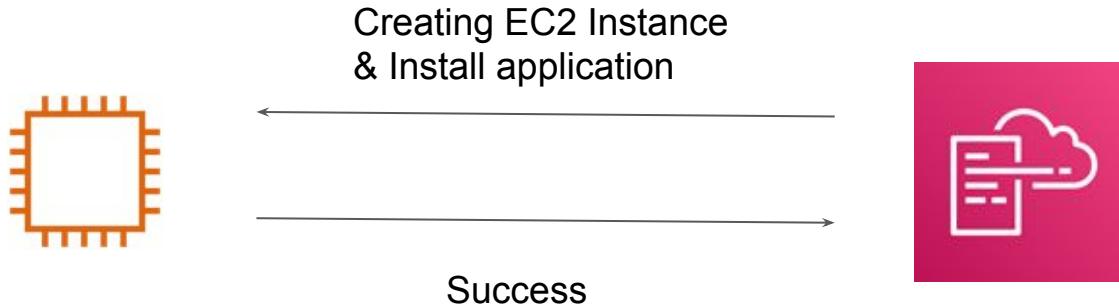
Example:

EC2 instance might be created, but application inside it is not ready yet.

Overview of Creation Policies

CreationPolicy so that CloudFormation proceeds with stack creation only after your configuration actions are done.

That way you'll know your applications are ready to go after stack creation succeeds.



Important Pointers

Two major AWS resources which are supported by Creation Policies are:

- Auto-Scaling Groups
- EC2 Instances

There is also an option to make use of WaitCondition to achieve similar purpose.
However it is used with different resource types.

WaitCondition and WaitConditionHandle

Need to learn the backend

Overview of WaitCondition

WaitCondition pauses the execution of stack and waits for success signals before it resumes the stack creation.

If no signal is received before the timeout, the WaitCondition enters the **CREATE_FAILED** state and the stack creation is rolled back.

WaitCondition:

```
Type: AWS::CloudFormation::WaitCondition
```

```
Properties:
```

```
  Handle: !Ref PrivateWaitHandle
```

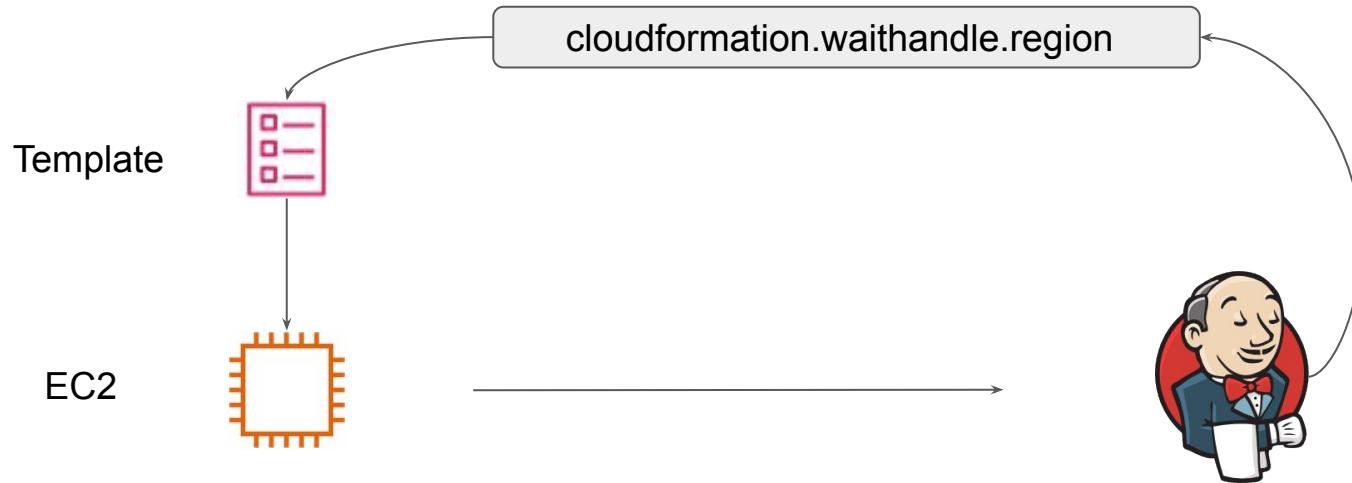
```
  Timeout: PT15M
```

```
  Count: 5
```

Overview of WaitConditionHandle

To make use of WaitCondition, you need to make use of WaitConditionHandle.

WaitConditionHandle reference resolves to a pre-sign URL that is used to signal success or failure signals to waitcondition.



Difference between WaitCondition and CreationPolicy

For CreationPolicy, any dependent resource waits on the CreationPolicy associated with the resource being created.

For WaitCondition, the dependent resource waits on the WaitCondition and not the original resource

A resource will not be marked created unless the CreationPolicy is completed.

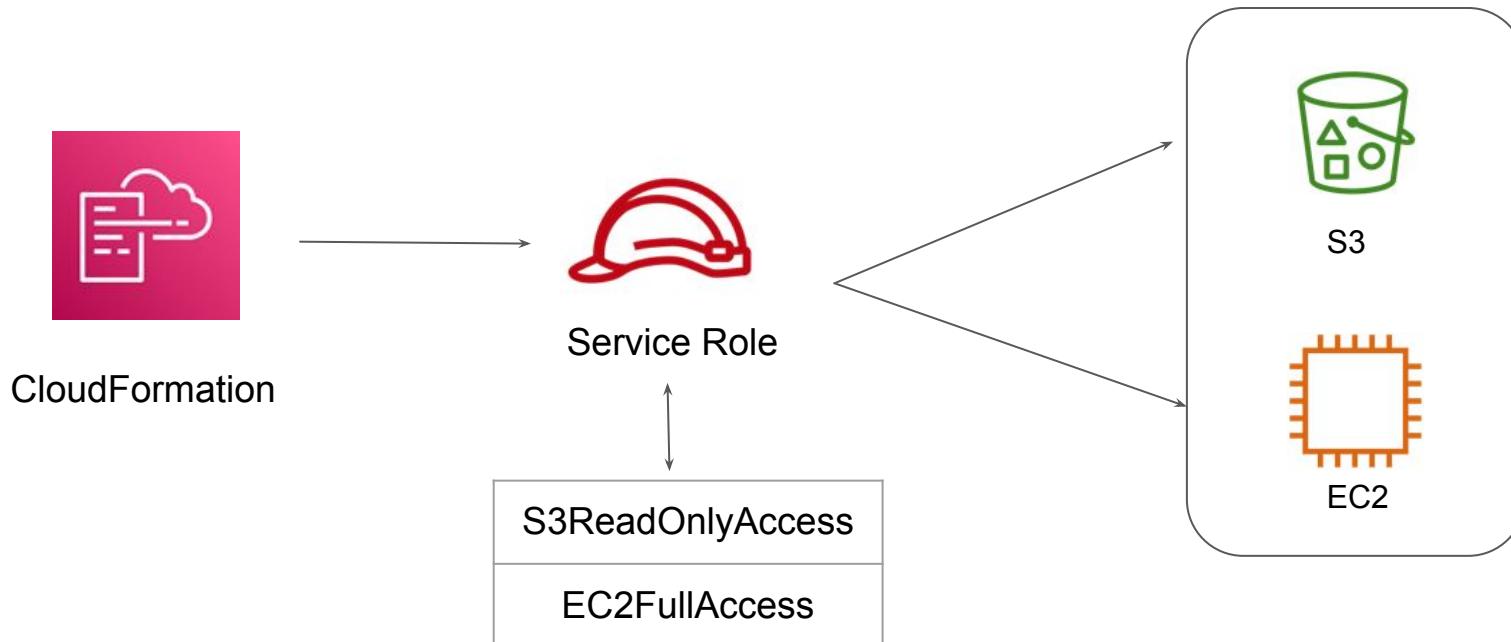
For WaitCondition, resource can be considered to be created, but WaitCondition itself would not be marked as Completed until it is complete.

Service Role & Pass Role

[Back to IAM](#)

Overview of Service Roles

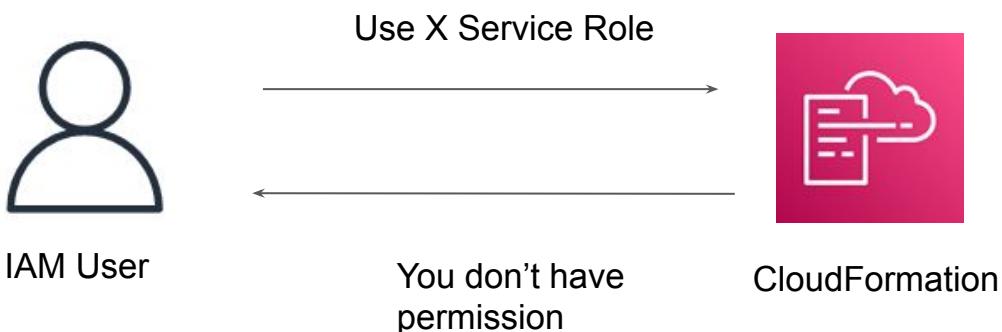
A service role is a role that an AWS service assumes to perform actions on your behalf.



Overview of PassRole

Pass Role allows the service to assume the role and perform actions on your behalf.

To pass a role (and its permissions) to an AWS service, a user must have permissions to pass the role to the service.

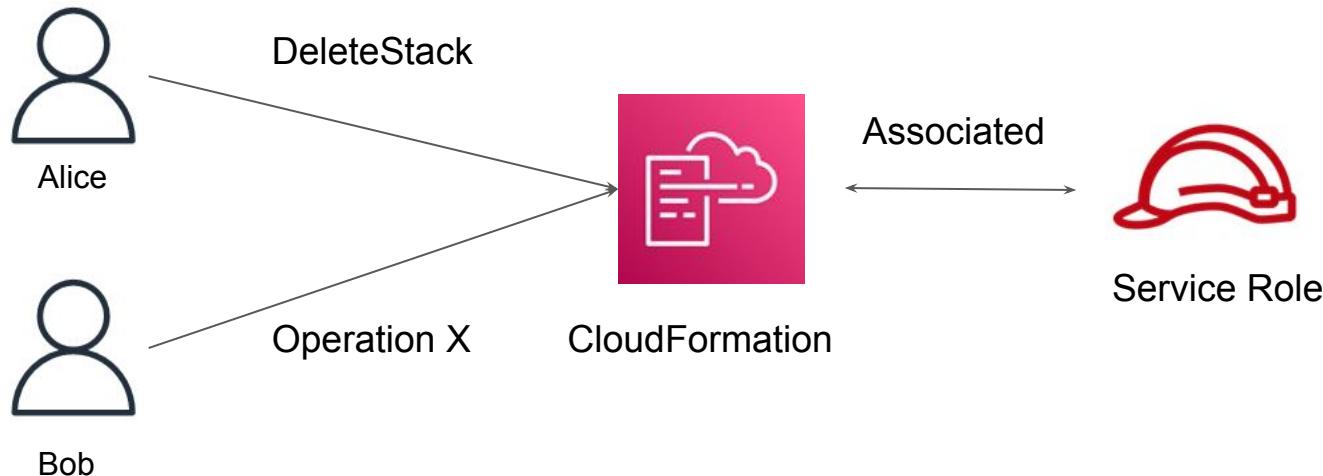


Sample PassRole Policy

```
{  
    "Version": "2012-10-17",  
    "Statement": [{  
        "Effect": "Allow",  
        "Action": [  
            "iam:GetRole",  
            "iam:PassRole"  
        ],  
        "Resource": "arn:aws:iam::<account-id>:role/EC2-roles-for-XYZ-*"  
    }]  
}
```

Important Pointer

Once the Role is associated with CloudFormation, other users that have permissions to operate on this stack will be able to use this role, even if they don't have permission to pass it. Ensure that this role grants least privilege.



OpsWorks

Automation

Introduction

AWS OpsWorks is a configuration management service that provides managed instances of Chef and Puppet.

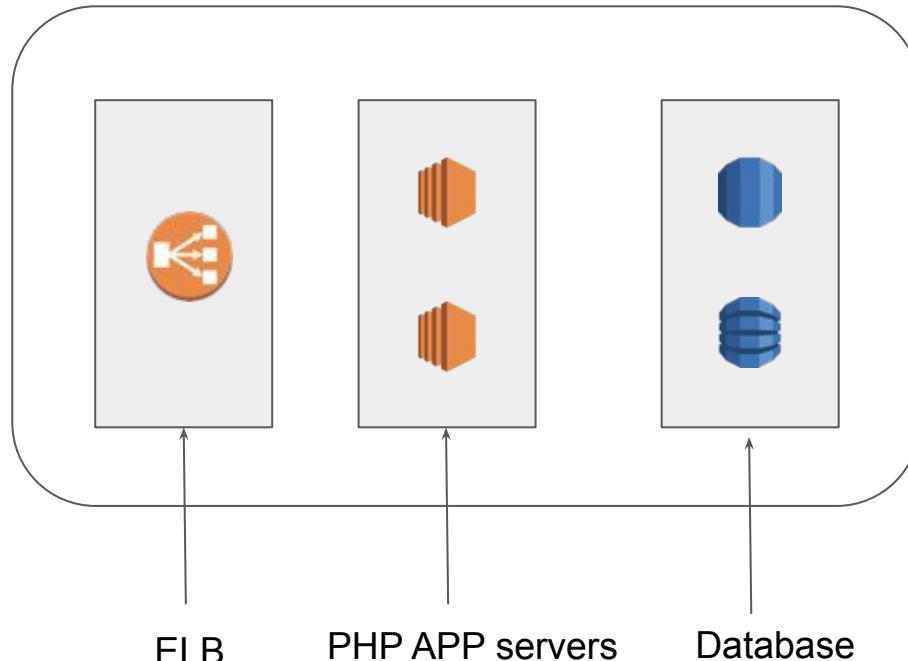
Integration of EC2 with the configuration management tool brings up great possibilities on how servers are configured, deployed and managed.

Example Use Case:

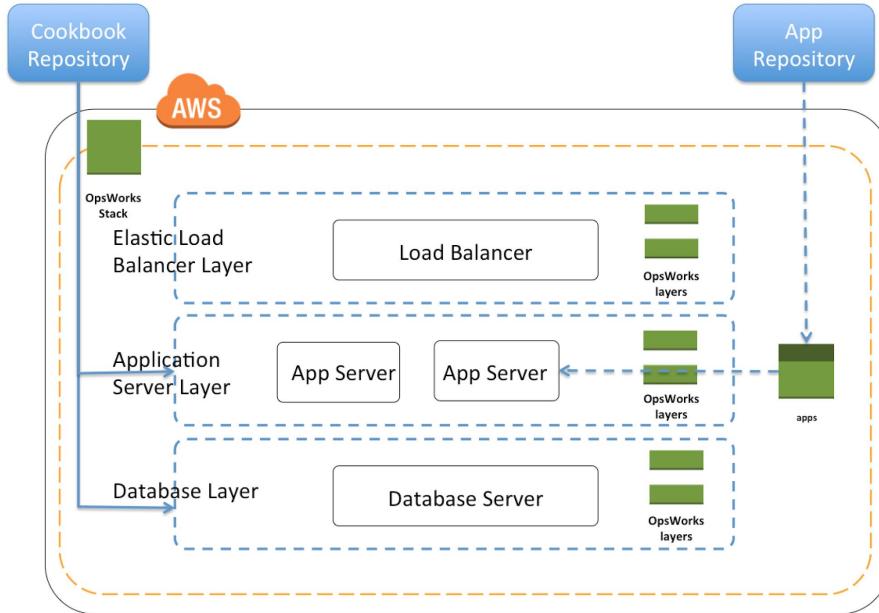
While an EC2 instance is getting launched, we want to install certain packages like Nginx, PHP-FPM and MySQL along with having custom SSH configuration file followed by restart of SSH server.

OpsWorks Concepts

OpsWorks Stack



Stacks & Layers



OpsWorks - Lifecycle Events

Deployments, yet again!

Getting Started

OpsWorks has “five” events that occur during the event lifecycle.

These events are :

1. Set up
2. Configure
3. Deploy
4. Undeploy
5. Shutdown

When an event occurs, it runs a set of chef recipes assigned to that event.

Getting Started

i) Setup Event:

- Event occurs when the started instance has finished booting up.
- Used for initial installation of software packages.

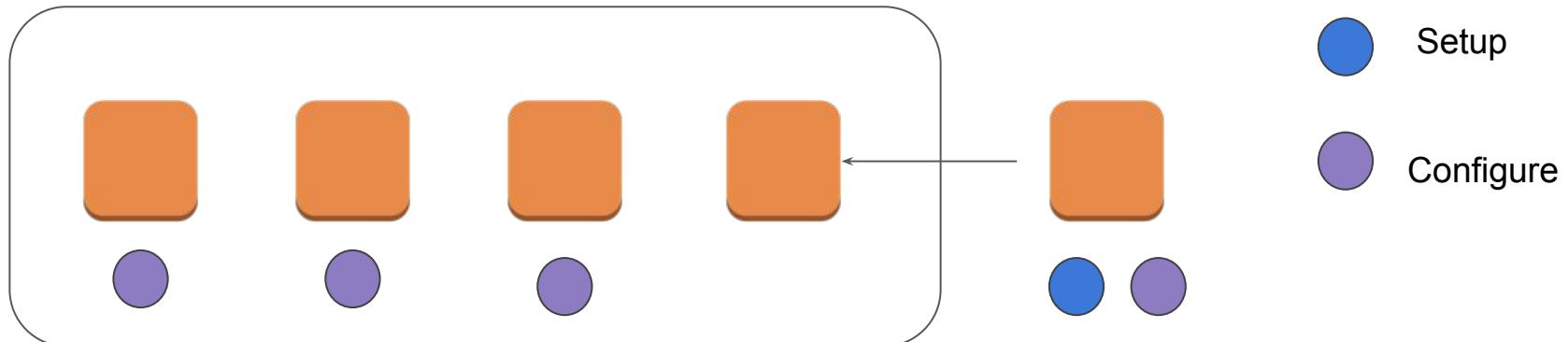
Example:

Installing PHP and Apache through layer recipes.

Configure Event

ii) Configure [Event occurs on all the stack's instances when]

- Events in configure are executed whenever instance enters or leaves online state.
- You associate or disassociate EIP of the instance.
- You attach or detach an ELB from the layer.



Deploy and Undeploy events

iii) Deploy

Deploy event allows us to manually define when we want to deploy a new version of app.

iv) Undeploy:

This event occurs when we delete the app or run the undeploy command to remove the app from the set of application servers.

ShutDown events

v) ShutDown

The event in this stage are executed when we inform OpsWorks to shut the instance down before the EC2 instances are terminated.

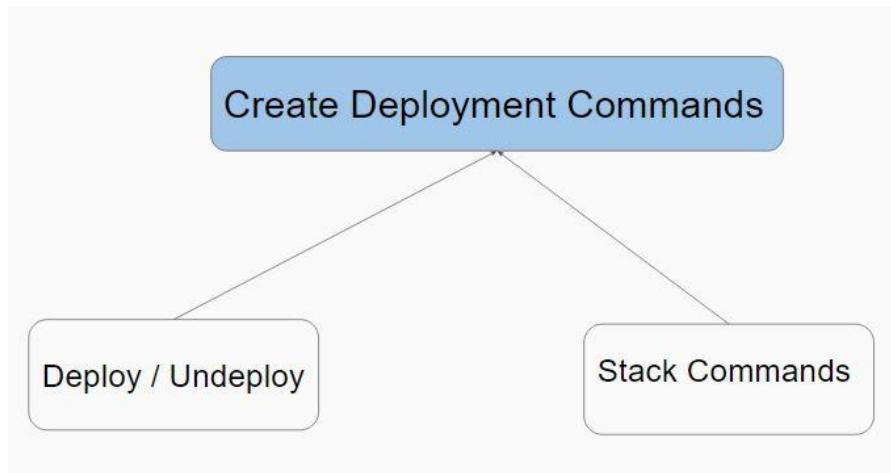
The default shutdown timeout if 120 seconds.

Create Deployment Commands

Support for Configuration Management

Create Deployment Commands

Create Deployment commands allows us to runs set of commands depending on use-case



Stack Commands

AWS OpsWorks provides set of stack commands which can be used to perform variety of operations on the stack's instances

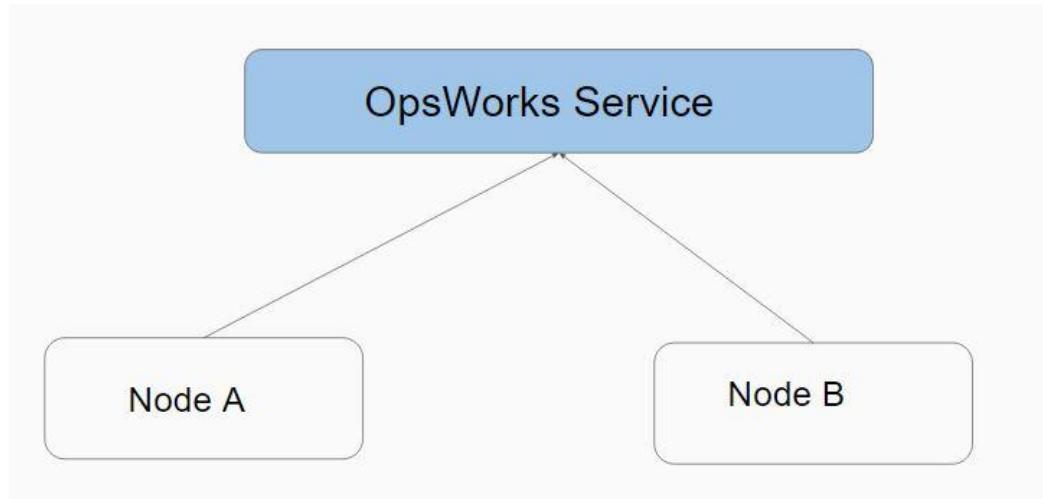
Update Custom Cookbook	Updates the instance's cookbook with current version from repo.
Execute Recipes:	Executes specified set of recipes on the instance.
Setup	Run the instance's setup recipes.
Configure	Run the instance's configure recipes.
Upgrade Operating System	Upgraded the OS to latest version (Linux Only)

OpsWorks Auto-Healing

Health Checking!

Create Deployment Commands

Every instance has an OpsWorks agent installed that communicates regularly with the service.



Auto-Healing Feature

If the agent does not communicate with the service for ~5 minutes, OpsWorks stack will consider the instance to have failed.

What will OpsWorks do after instance is marked as failed:

EBS Backed Volume: Stop the EC2 Instance → Start the EC2 instance

What it doesn't do

Many times due to corruption of file-system or EBS backed volume, the EC2 instance might fail to start after it has stopped.

This would show up as **start_failed** error.

In such situations, manual intervention is required.

Auto-Healing OS will also not upgrade the OS of the instance. For example if the default OS is updated at the stack level to CentOS, auto-healing feature will return with the same OS of the instance before it was healed. [It's not for performance, it's for failure]

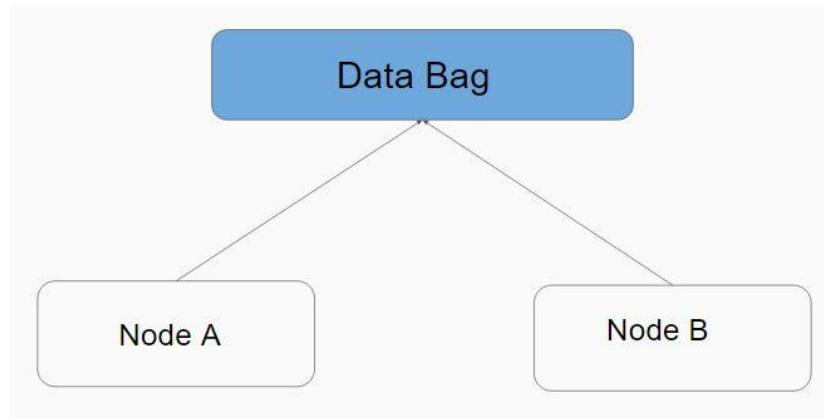
DataBags

Global Variables

Overview of DataBags

DataBag is basically a Chef concept and is basically a global variable.

Data bags are generally stored in a central location and can be searched.



OpsWorks Databags

OpsWorks stores data bags at various level which includes:

- At Stack Level
- At Layers
- Instances
- Application level

AWS SAM

Serverless Application Model

Overview of SAM

We can make use of SAM to define serverless application in simple and clear syntax.



MEET SAM.



USE SAM TO BUILD TEMPLATES THAT DEFINE
YOUR SERVERLESS APPLICATIONS.



DEPLOY YOUR SAM TEMPLATE
WITH AWS CLOUDFORMATION.

How things works ?

There are two important steps during the SAM process:

- Create a SAM template (JSON / YAML) that defines Lambda, API and others.
- Test, Upload and Deploy application using SAM CLI.

Info:

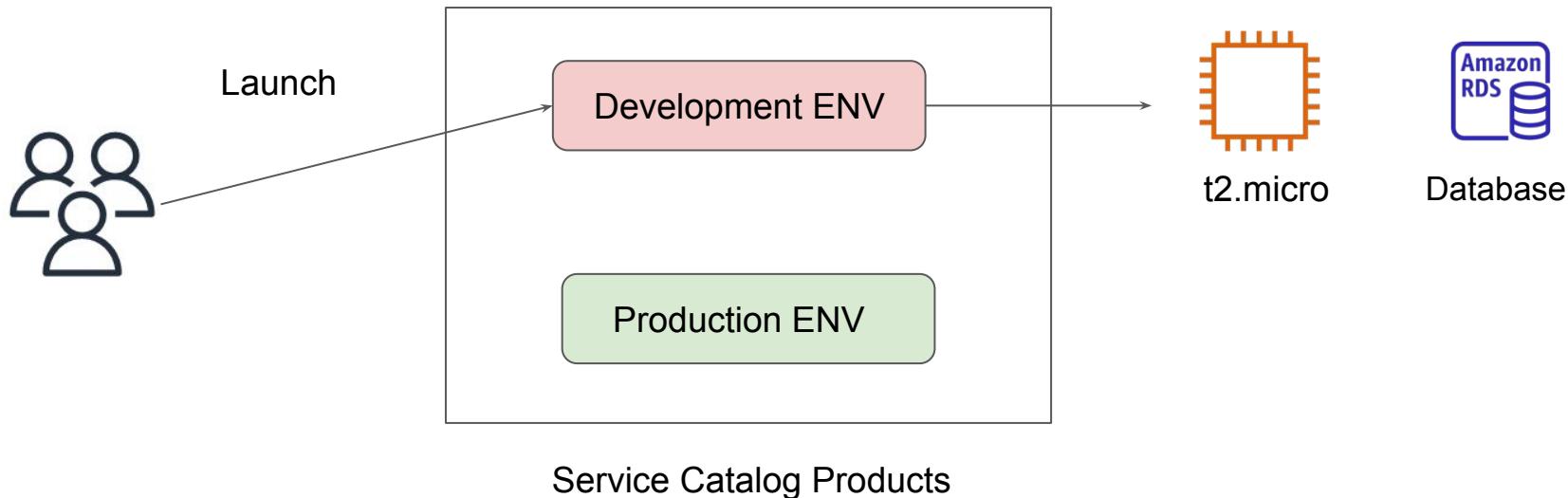
During deployment, SAM automatically translates application specs into CloudFormation Syntax.

AWS Service Catalog

Standardized Stack

Understanding the Workflow

AWS Service Catalog enables organizations to create and manage catalogs of IT services that are approved for use on AWS.



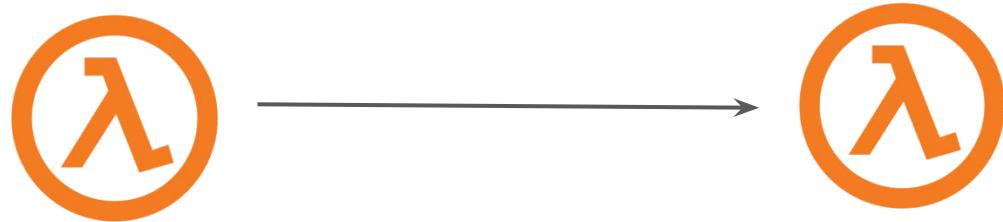
Step Functions

Coordinating across distributed components

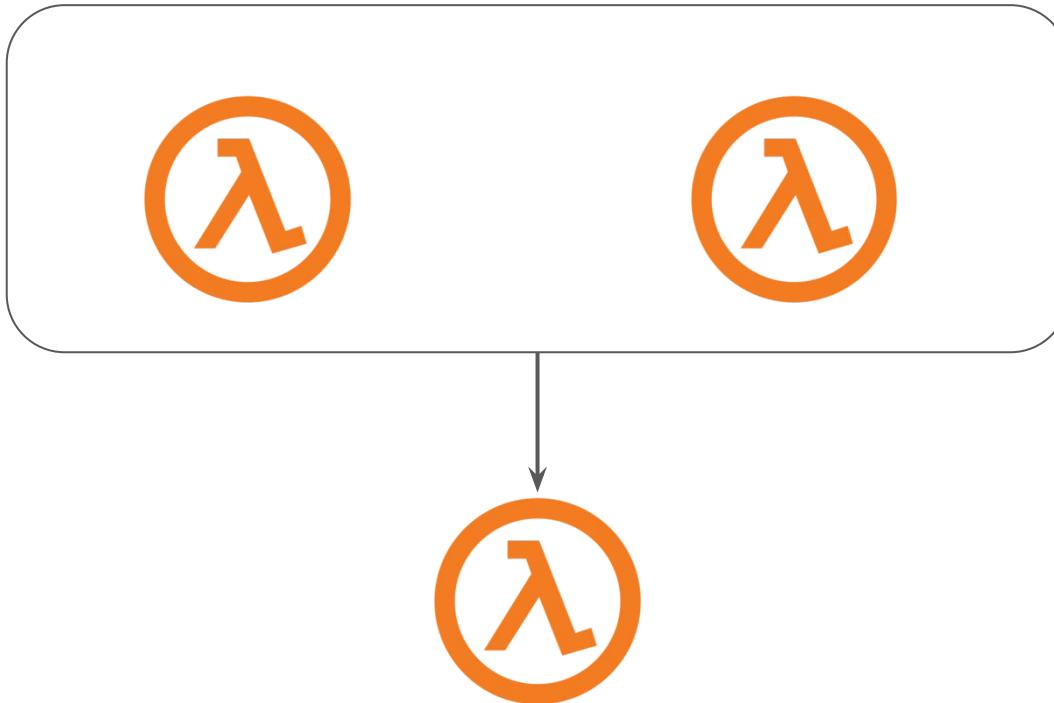
Overview of Step Functions

Step Functions are generally used as an orchestration for serverless functions.

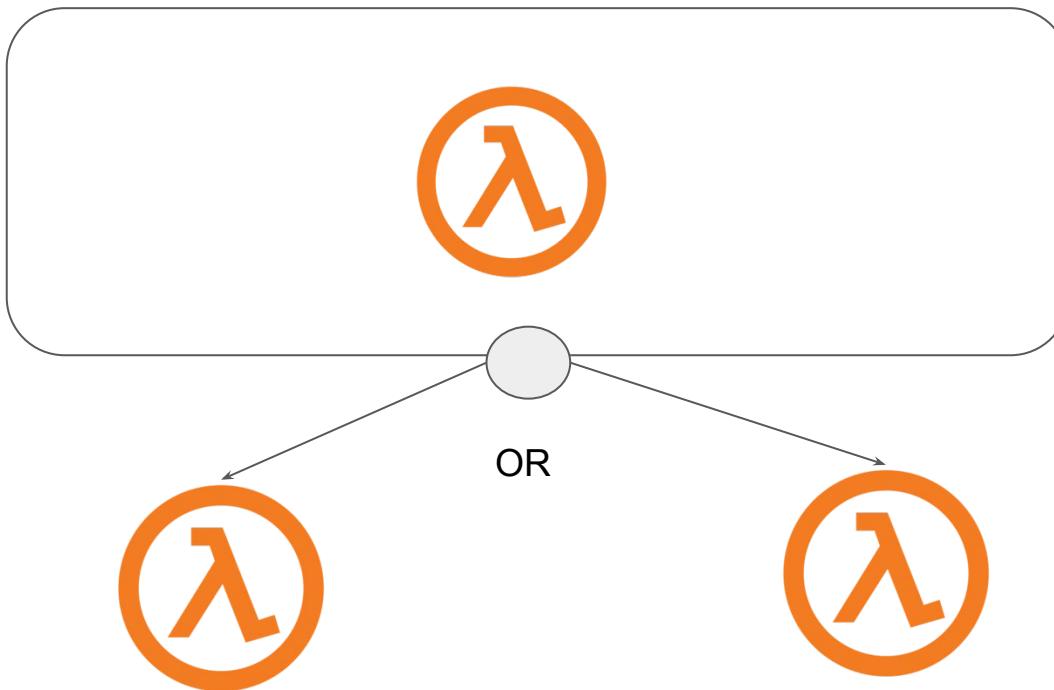
One of the question that comes when you use serverless is, how can we turn serverless into apps ?



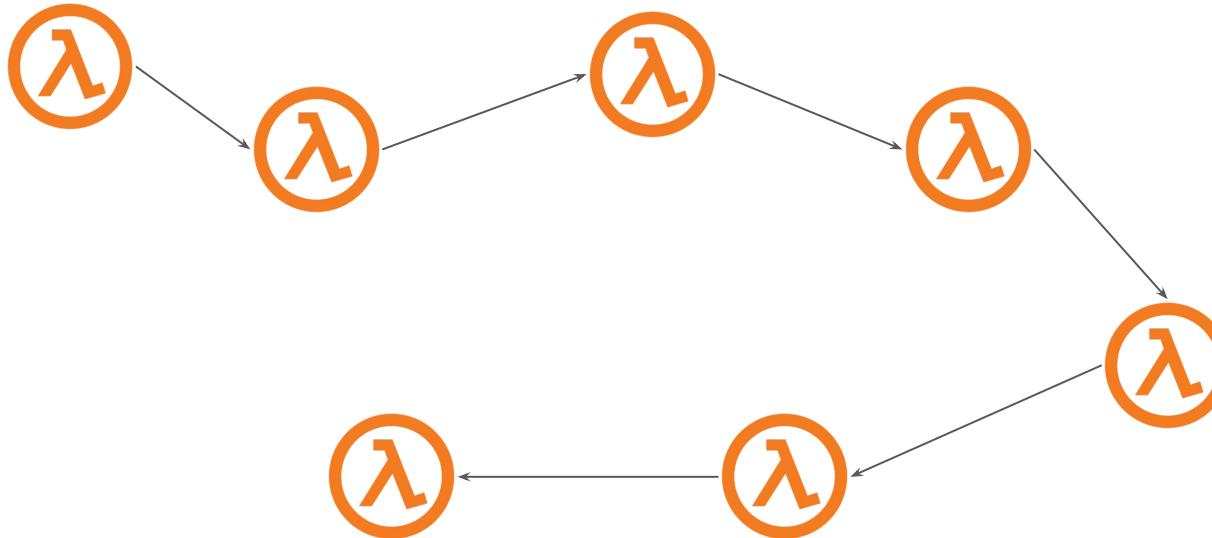
Running Functions in Parallel



Selecting Function Based on Data

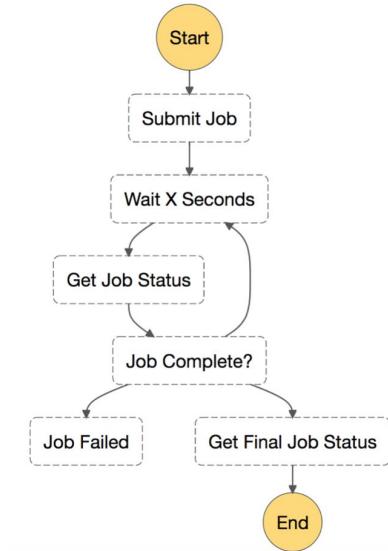


Coordinating Lambda Function



Overview of Step Functions

Step Functions makes it easy to coordinate the components of distributed application using visual workflow.



AWS License Manager

Let's understand Licensing

Getting Started

In Enterprises, managing software licenses sometimes becomes quite a hassle.

Organizations uses wide variety of software licenses:

- OS Level Licenses : Windows, RedHat
- Database Licenses : Oracle DB, Microsoft SQL
- Application Licenses: SAP
- Other 3rd Party Licenses

Challenges

In Cloud, new servers can be launched in click of a button.

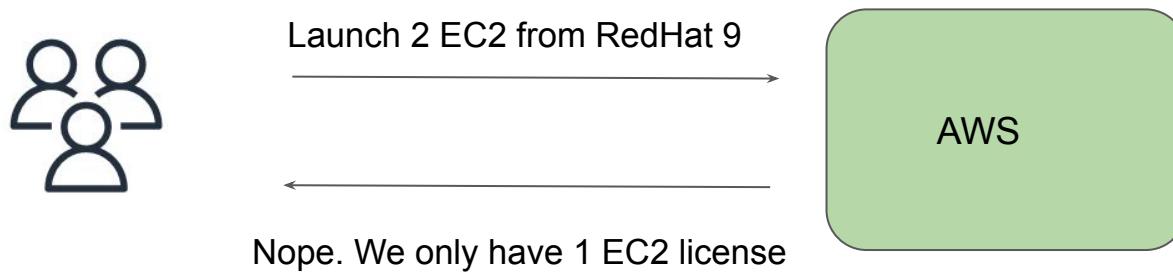
License Violation detected during audit can lead to heavy penalties.

Difficult to track licenses across multiple accounts.

Overview of AWS License Manager

AWS License Manager is a service which allows us to manage license from wide variety of software vendors across AWS and on-premise.

We can **enforce policies** for licenses based on various factors like CPU, sockets that will control the number of EC2 instance which can be launched.



RDS Read Replicas



Use Case : Bank

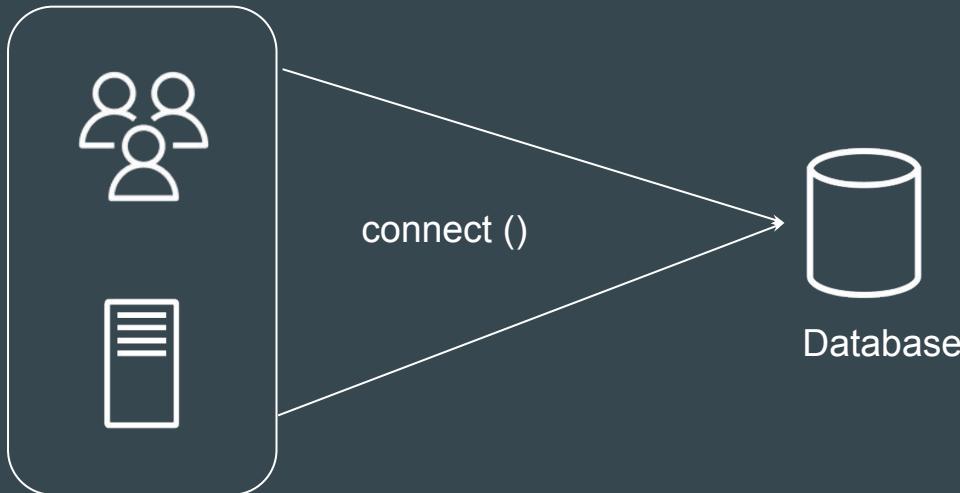
In bank, for different kind of work purpose, there are different people you might have to approach. For example :

- Cash Collector
- Cheque Counter
- Enquiry Counter



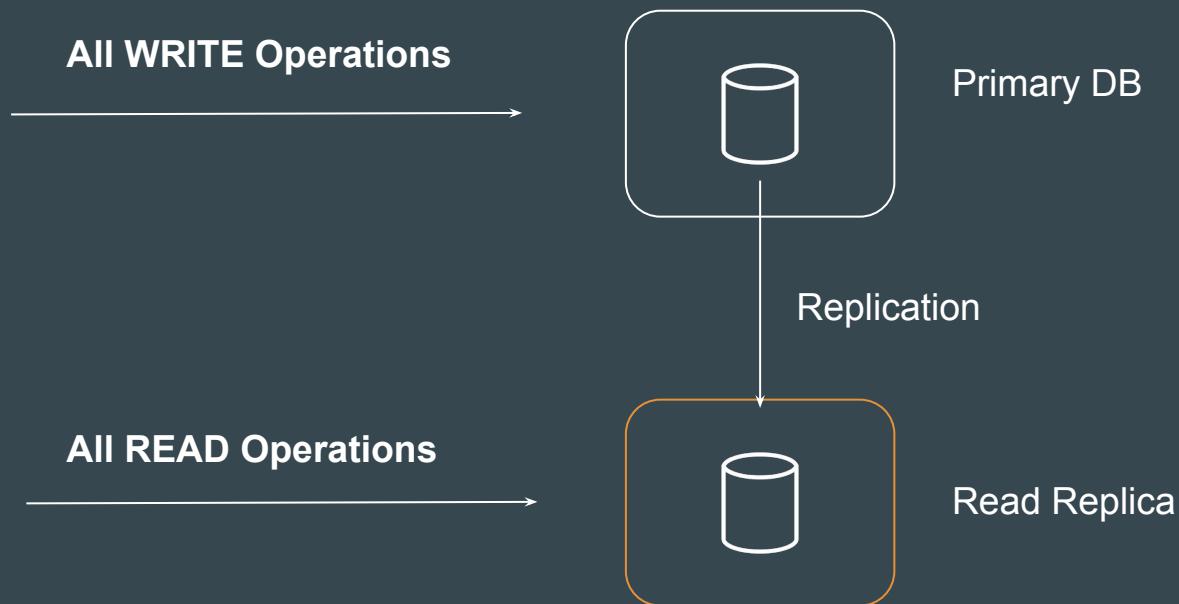
Database Way

Using a single database for all kind of activity will increase the database load and slow down the operations.



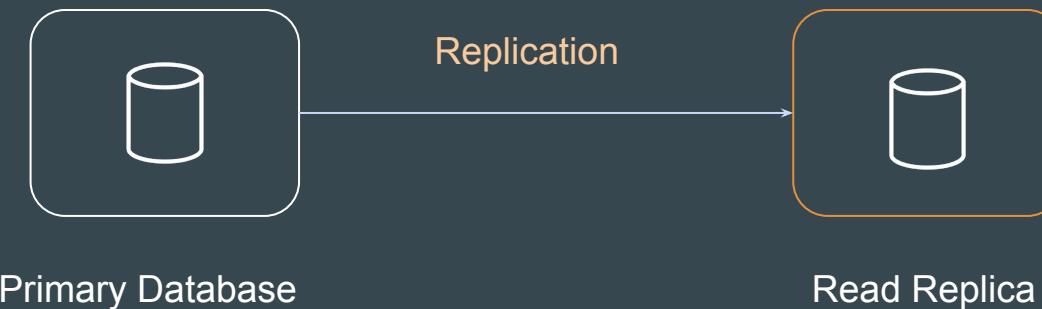
Improved Architecture - Read Replica

Read Replica allows customers to offload read requests or analytics traffic from the primary instance



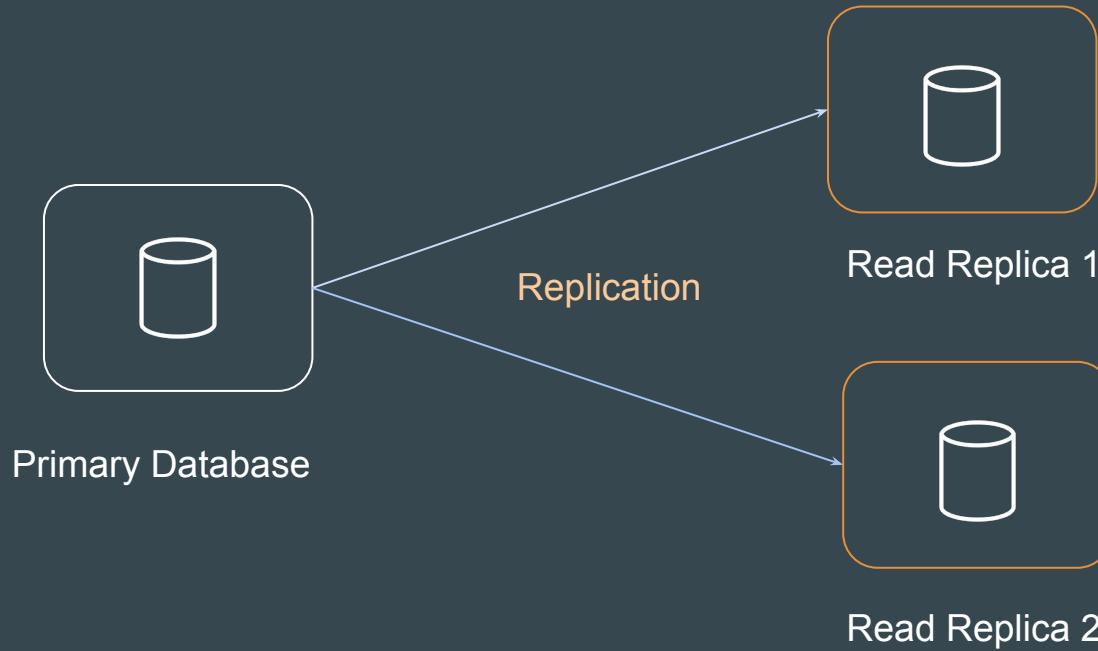
RDS Read Replica

RDS Read Replica feature allows customers to implement “Database Read Replica” functionality for RDS databases.



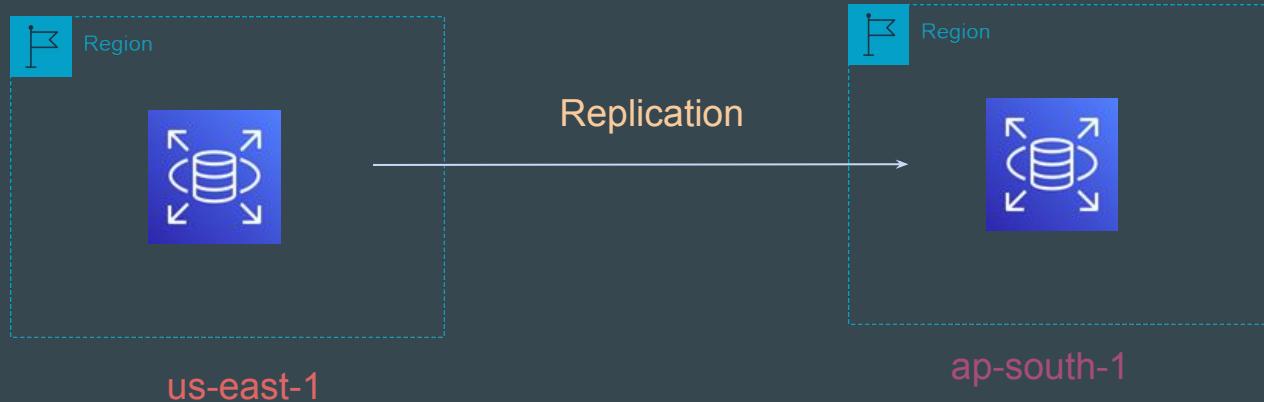
Pointers to Note - 1

You can create one or more replicas of a given source DB Instance and serve high-volume application read traffic.



Pointers to Note - 2

With Amazon RDS, you can create a read replica in a different AWS Region from the source DB instance.

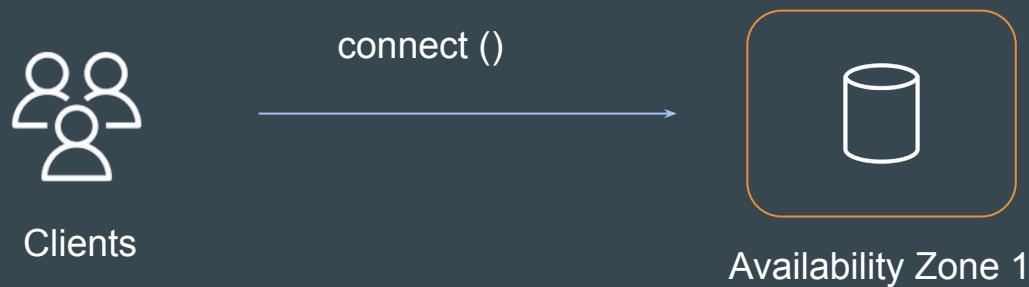


Amazon RDS Multi AZ Deployments



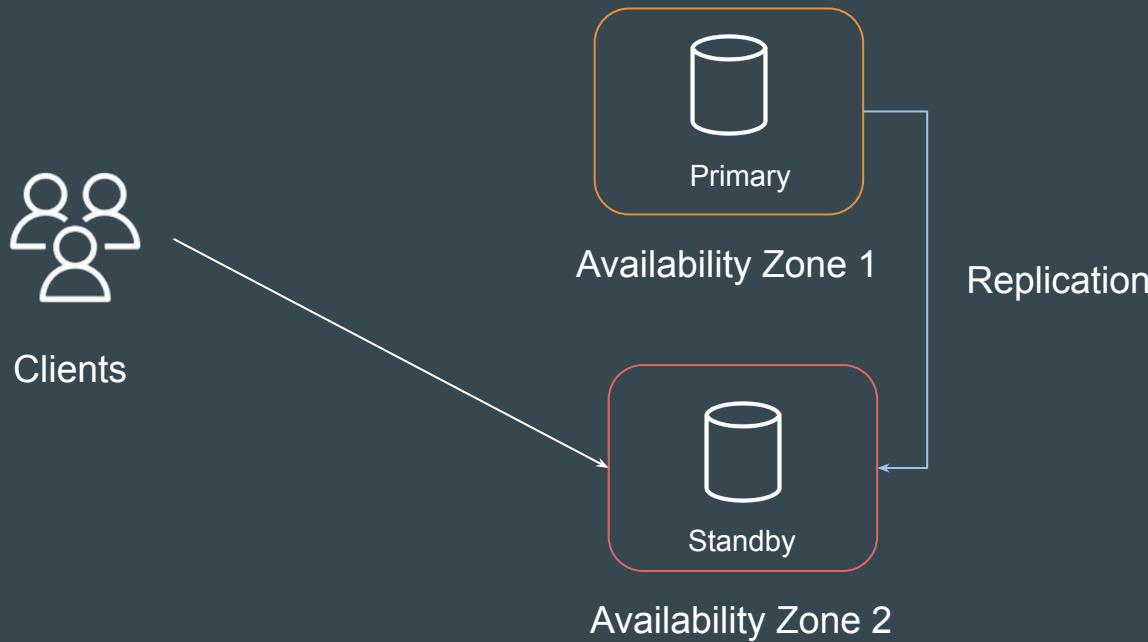
Understanding the Challenge

If database is running in a specific availability zone and if the AZ is down or unreachable then your entire application can be impacted.



Multi-AZ Architecture

In this approach, Amazon creates a standby DB instance and synchronously replicates data from the primary DB instance in a different availability zone.



Failover Condition

If a planned or unplanned outage of your DB instance results from an infrastructure defect, Amazon RDS automatically switches to a standby replica in another Availability Zone if you have turned on Multi-AZ.

- Loss of availability in primary Availability Zone
- Loss of network connectivity to primary
- Compute unit failure on primary
- Storage failure on primary

Failover times are typically 60–120 seconds.

Multi AZ Deployment Types



Types of Multi-AZ Deployments

There are two primary deployment types available in Multi-AZ based approach.



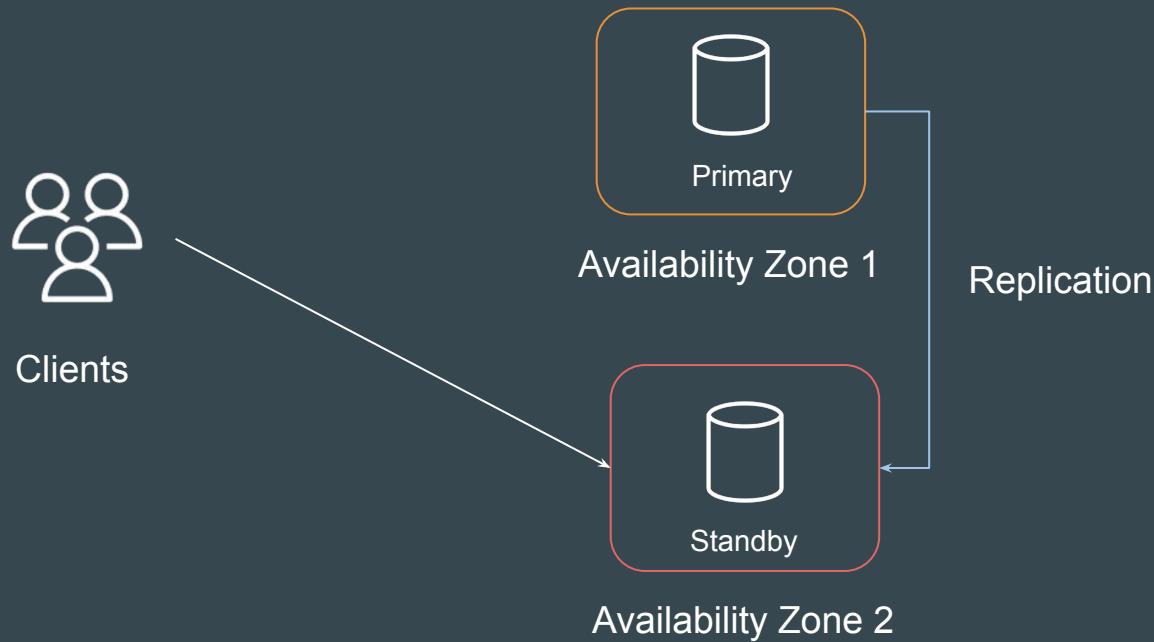
Multi-AZ Instance Deployment



Multi-AZ Cluster Deployment

Approach 1 - Multi-AZ Instance Deployment

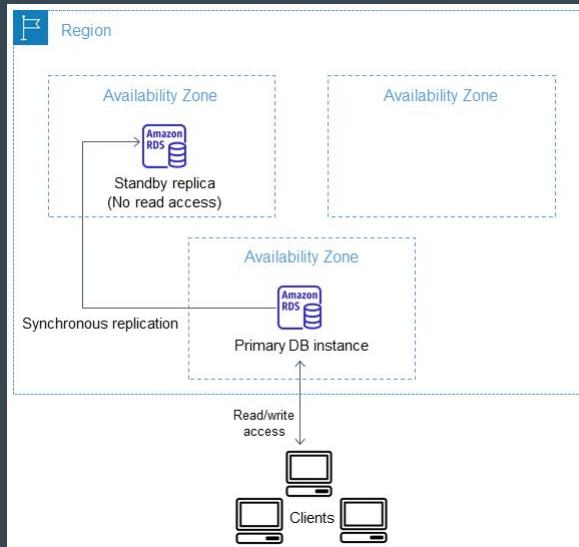
Referred to as RDS Multi-AZ with one standby



Approach 1 - Multi-AZ Instance Deployment

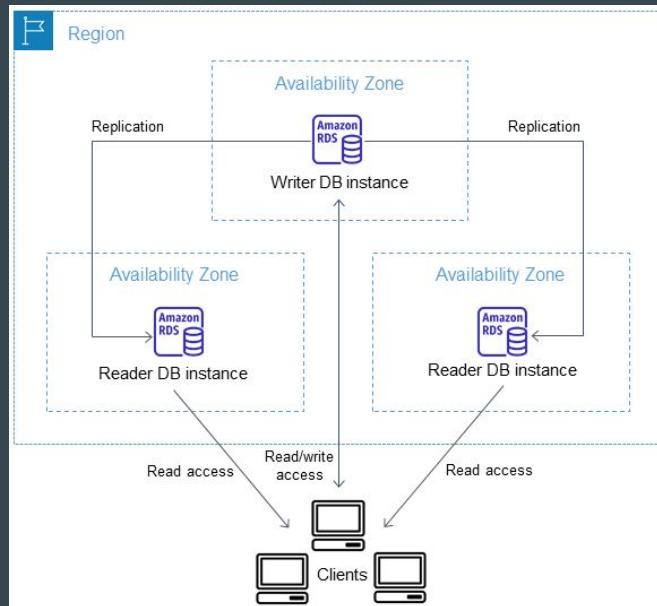
Referred to as RDS Multi-AZ with one standby

Cannot perform any operation on standby replica (including read)



Approach 2 - Multi-AZ Cluster Deployment

A Multi-AZ DB cluster has a writer DB instance and two reader DB instances in three separate Availability Zones in the same AWS Region.



Different Deployment Types

Feature	Single-AZ	Multi-AZ with 1 Standby	Multi-AZ with 2 readable Standby
Additional Read Capacity	None (only primary)	None (only primary)	2 standby DB instance
Automatic Failover Detection	None	Yes	Yes
Failover Duration	NA	New primary is available to serve workload in as quickly as 60 seconds	New primary is available to serve workload in typically under 35 seconds

RDS Event Notification

Back to Notifications!

RDS Event Notification

RDS Event Notification provides notification when a specific type of RDS event occurs.

These events are categorized into multiple categories like Availability, Configuration Change, Failure, Deletion, Low Storage and others.



Amazon Aurora

Closed Source Database

Overview of Database Offerings

Databases are generally divided into two types:

- Open Source Databases
- Commercial Databases

Commercial Offering does come with various aspects that are not found in the open source databases.

Open Source Databases

Commercial Databases

Introducing Aurora

Amazon Aurora is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases.

Amazon Aurora is up to five times faster than standard MySQL databases and three times faster than standard PostgreSQL databases.

It provides the security, availability, and reliability of commercial databases at 1/10th the cost

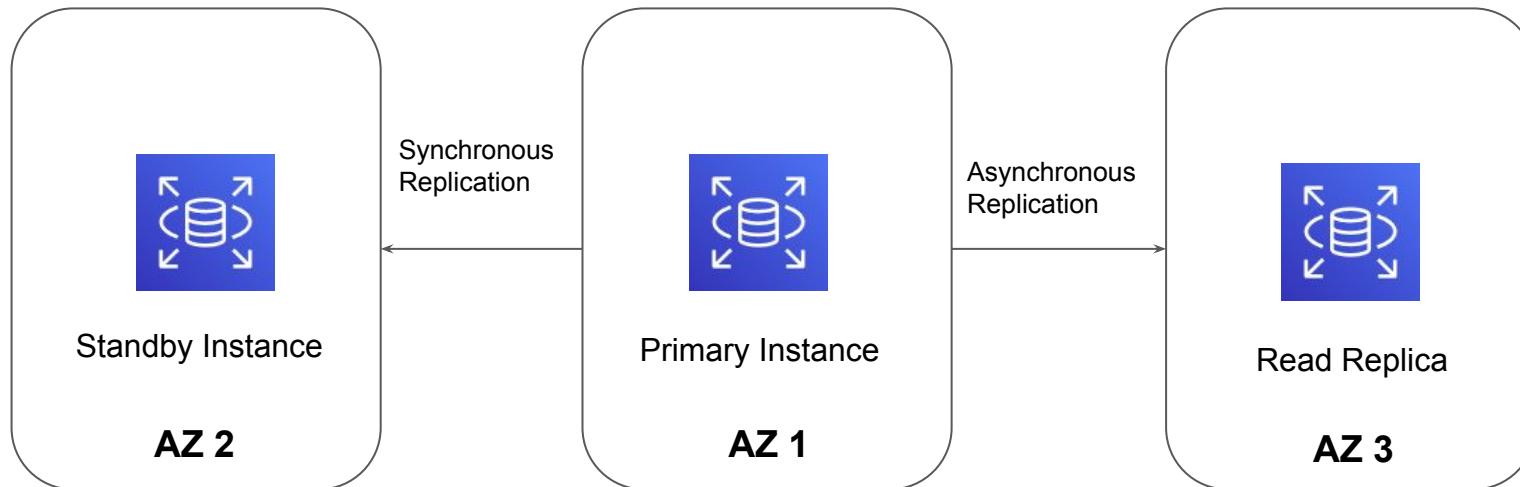
Performance /
Availability of
Enterprise Databases

Simplicity and Cost
Effectiveness of Open
Source Databases

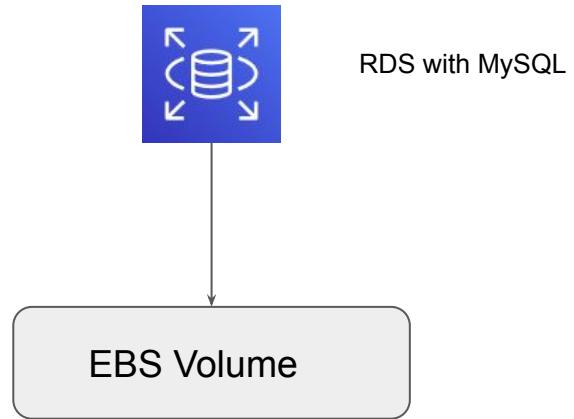
Amazon Aurora

RDS - Multi-AZ & Read Replica Architecture

In a typical setup, primary, standby and read replicas are three different instances in multiple availability zones. The underlying storage is EBS volume.



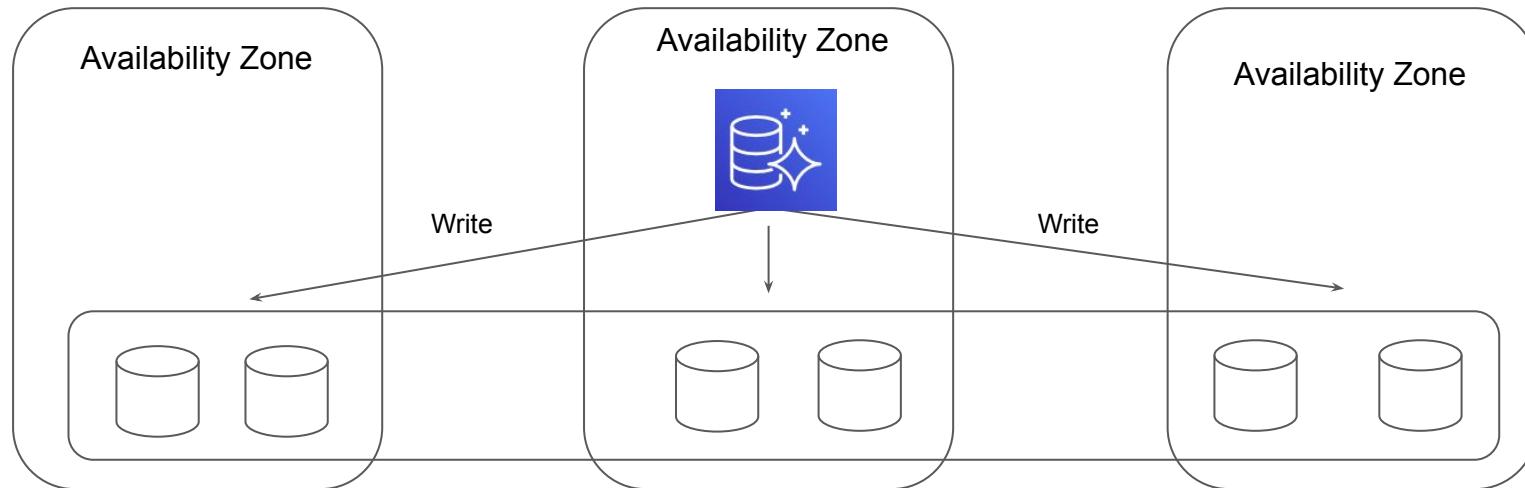
RDS with EBS Storage



Aurora Architecture

Two Primary Components: DB Instances + Storage Cluster Volume

Since Aurora and Storage Layer are independent, we can scale the storage easily.



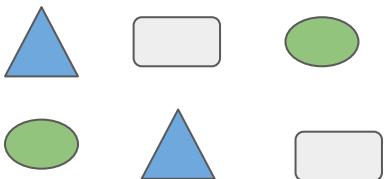
Overview of Storage Volume

Availability Zone 1

Availability Zone 2

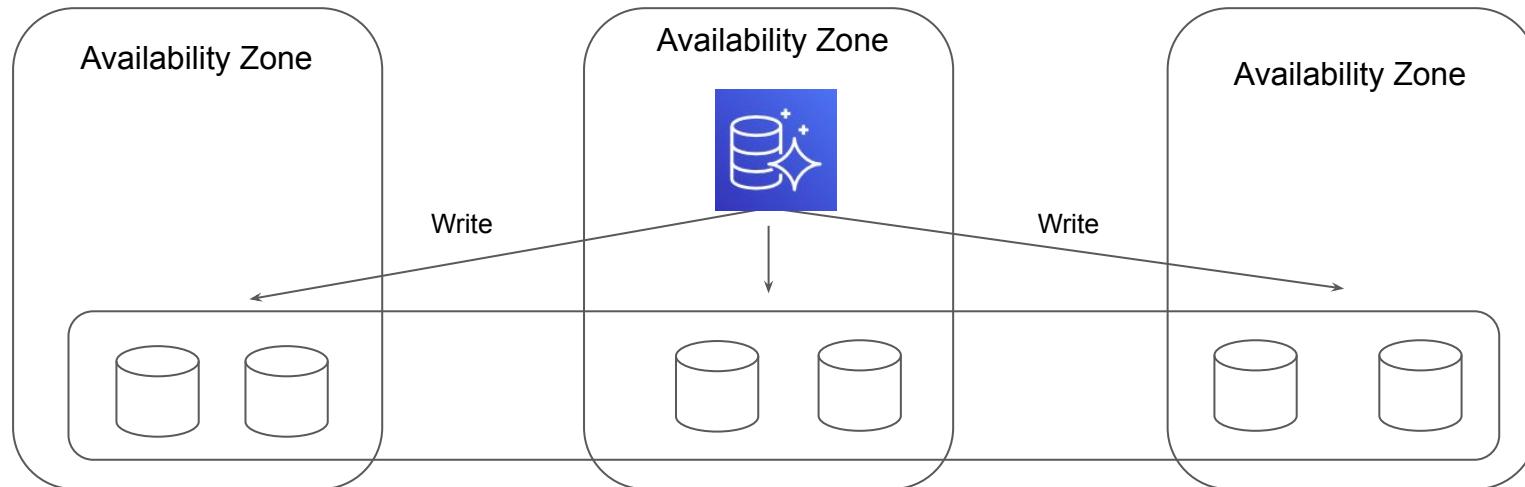
Availability Zone 3

Cluster Storage Volume



Scalability Aspect in Aurora

With this architecture, you can add a DB instance quickly because Aurora doesn't make a new copy of the table data. Instead, the DB instance connects to the shared volume that already contains all your data.



Scale at a Faster Pace

Availability Zone 1



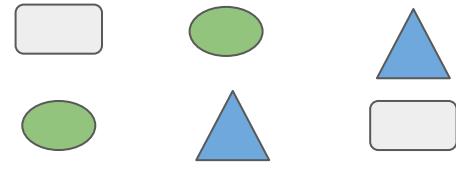
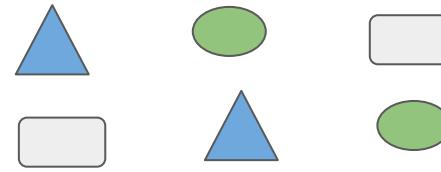
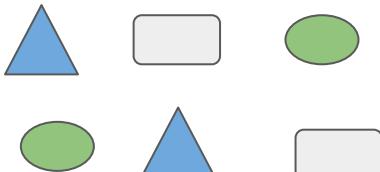
Availability Zone 2



Availability Zone 3



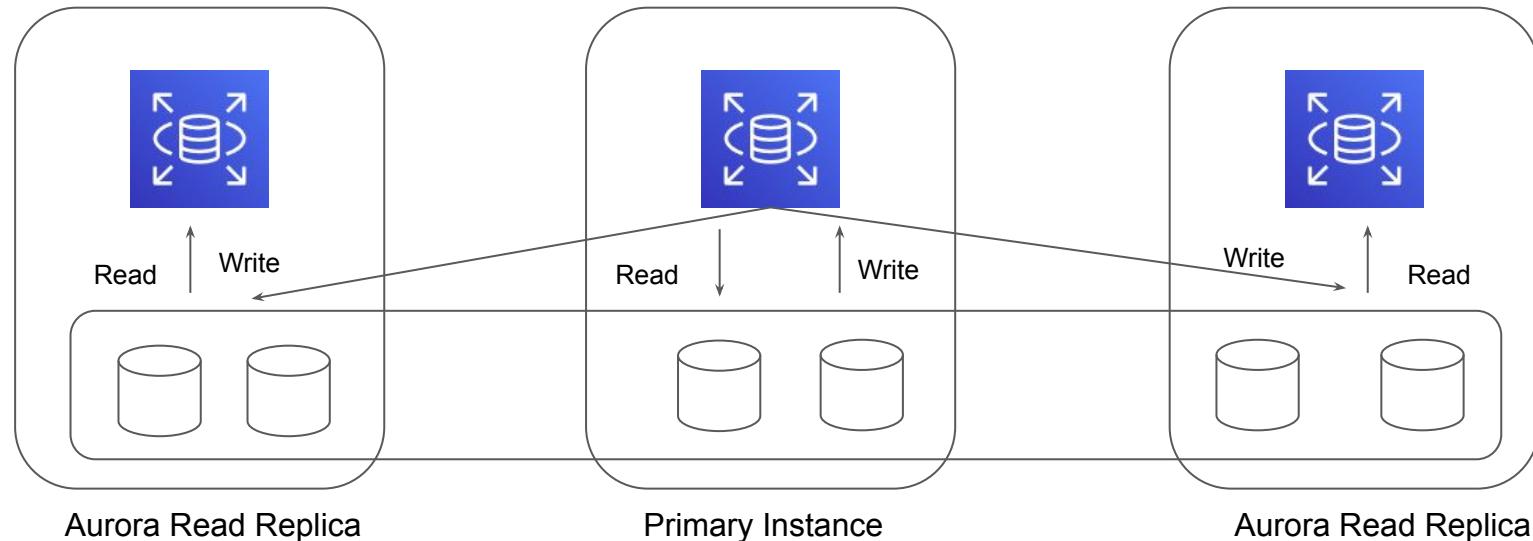
Cluster Storage Volume



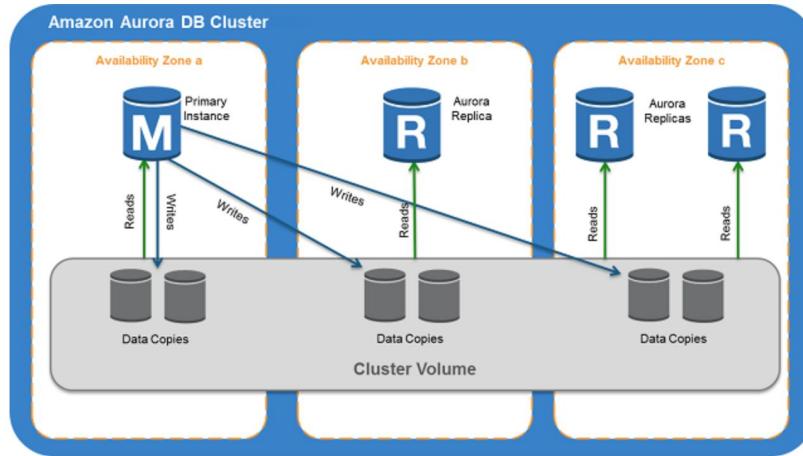
Aurora Architecture

Two Primary Components: DB Instances + Storage Cluster Volume

Since Aurora and Storage Layer are independent, we can scale the storage easily.



Aurora Architecture



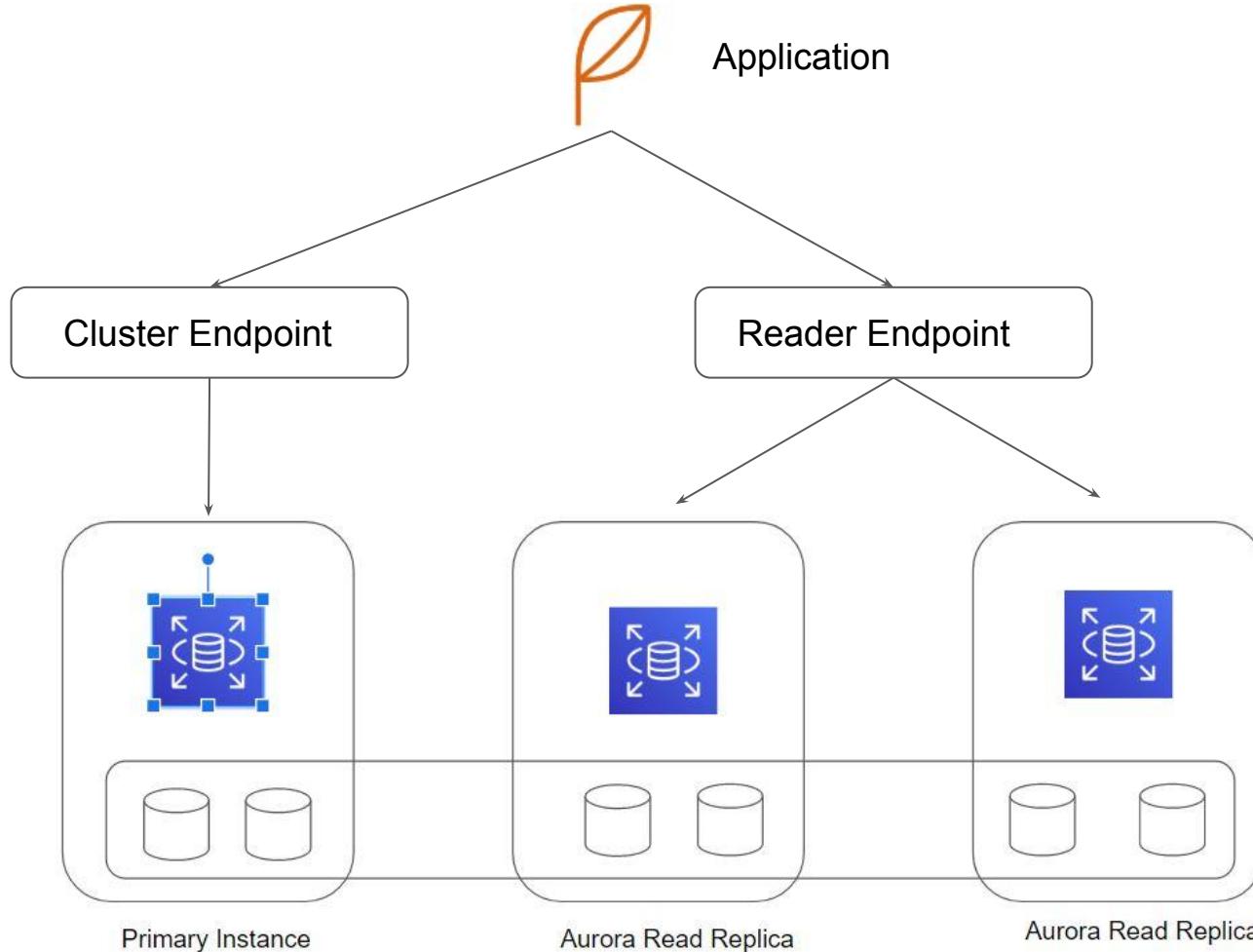
Aurora Endpoints

You can connect to Aurora Cluster through endpoints.

Endpoints is Aurora Specific URL consisting of host and port.

There are three primary types of endpoints available:

- Cluster Level Endpoints
- Reader Level Endpoints
- Instance Level Endpoints



Aurora Endpoints

Endpoint Types	Description
Cluster Level Endpoints	Connects to current primary DB instance in the cluster. Used for performing write operations.
Reader Level Endpoints	Built-In endpoints for Read Replicas. For Multiple Read Replicas, this endpoint will balance load among all read replicas.
Instance Endpoints	Allows connection directly to the instance.
Custom Endpoints	Ability to create custom endpoints for our own requirements.

Aurora Features

Aurora provides wide variety of interesting features. Some of these includes:

Global Databases	Serverless
Cross Region Replication	Auto-Scaling
BackTrack	IAM DB Authentication
Sharing DB Clusters	RDS Proxy

Aurora Architecture

Two Primary Components: DB Instances + Cluster Volume

Since Aurora and Storage Layer are independent, we can scale the storage easily.



Primary Instance

Aurora Read Replica

Aurora Read Replica

Aurora Global Database

Scalability Aspect

Overview of Global Database

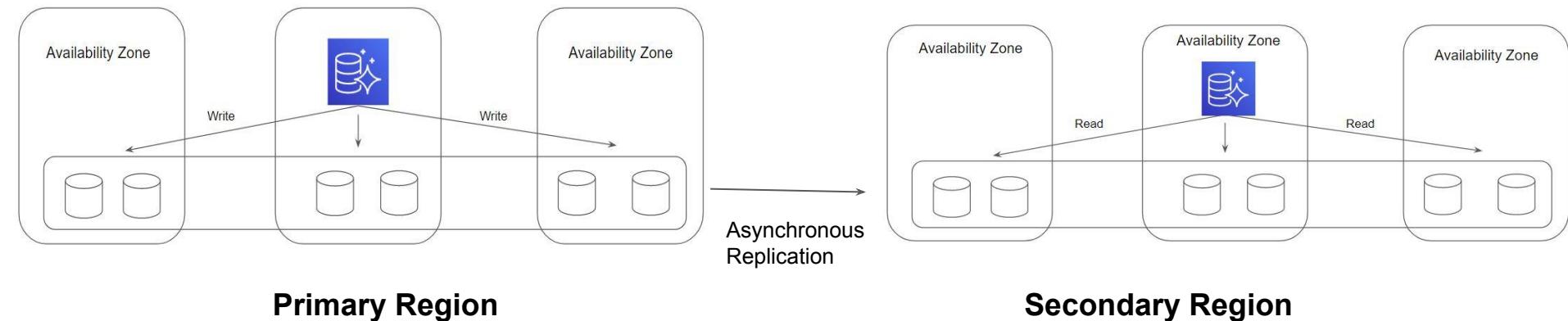
Aurora Global Database allows a single Amazon Aurora database to span multiple AWS regions.

It replicates your data with no impact on database performance, enables fast local reads with low latency in each region, and provides disaster recovery from region-wide outages.



Replication Approach

Data is replicated based on asynchronous replication between the storage layer of the two regions.



Important Pointers

Global Database does not support automated failover to the secondary region. This step is manual.

Not all instance types are supported. You can't use db.t2 or db.t3 instance classes.

Certain features like Backtrack are not supported.

Stopping and starting the DB clusters within the global database is not supported.

Aurora Scaling



Storage Scaling

Aurora storage **automatically scales** with the data in your cluster volume.

As your data grows, your cluster volume storage expands up to a maximum of 128 tebibytes (TiB) or 64 TiB

Even though an Aurora cluster volume can scale up in size to many tebibytes, you are only charged for the space that you use in the volume.

Instance Scaling

You can scale your Aurora DB cluster as needed by modifying the DB instance class for each DB instance in the DB cluster

Aurora supports several DB instance classes optimized for Aurora, depending on database engine compatibility.

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)

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

db.t3.small
2 vCPUs 2 GiB RAM Network: 2,085 Mbps

Include previous generation classes

Read Scaling

You can achieve read scaling for your Aurora DB cluster by creating up to 15 Aurora Replicas in a DB cluster that uses single-master replication.

Each Aurora Replica returns the same data from the cluster volume with minimal replica lag

As your read traffic increases, you can create additional Aurora Replicas and connect to them directly to distribute the read load for your DB cluster.

Aurora Auto Scaling with Aurora replicas

Aurora Auto Scaling dynamically adjusts the number of Aurora Replicas provisioned for an Aurora DB cluster using single-master replication based on the workload.

When the connectivity or workload decreases, Aurora Auto Scaling removes unnecessary Aurora Replicas.

You define and apply a scaling policy to an Aurora DB cluster. The scaling policy defines the minimum and maximum number of Aurora Replicas that Aurora Auto Scaling can manage.

Policy details

Policy name

A name for the policy used to identify it in the console, CLI, API, notifications, and events.

Policy name must be 1 to 256 characters.

IAM role

The following service-linked role is used by Aurora Auto Scaling.

Target metric

Only one Aurora Auto Scaling policy is allowed for one metric.

- Average CPU utilization of Aurora Replicas [View metric](#)
- Average connections of Aurora Replicas [View metric](#)

Target value

Specify the desired value for the selected metric. Aurora Replicas will be added or removed to keep the metric close to the specified value.

 %

► Additional configuration

Cluster capacity details

Configure the minimum and maximum number of Aurora Replicas you want Aurora Auto Scaling to maintain.

Minimum capacity

Specify the minimum number of Aurora Replicas to maintain.

 Aurora Replicas

Maximum capacity

Specify the maximum number of Aurora Replicas to maintain. Up to 15 Aurora Replicas are supported.

 Aurora Replicas[Cancel](#)[Add policy](#)

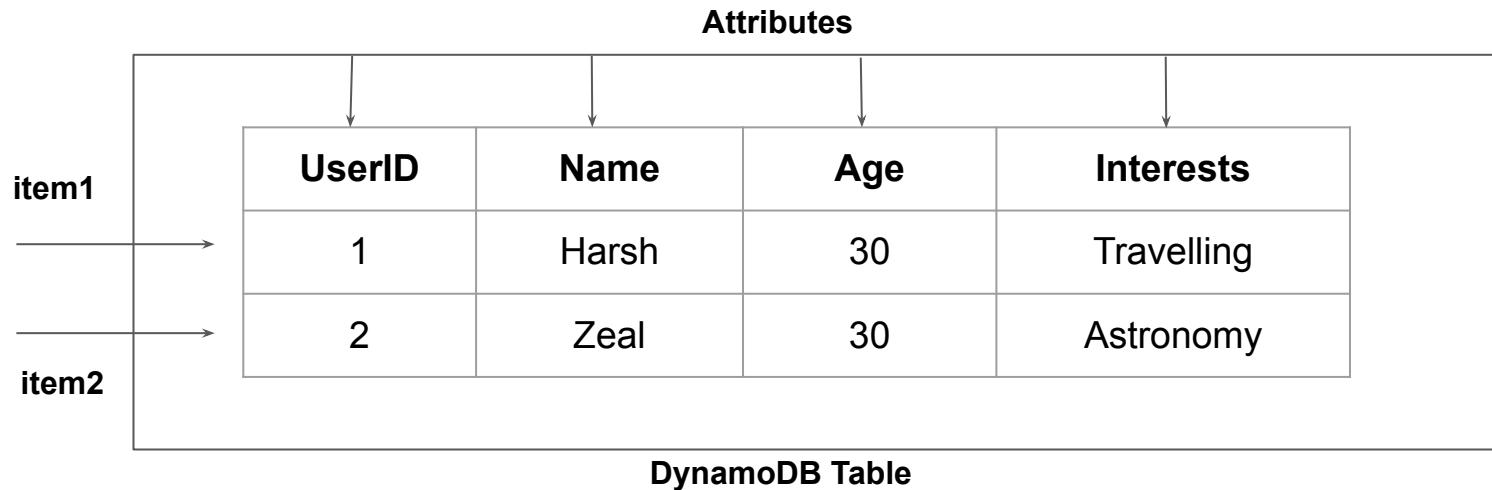
Core Components - DynamoDB

DynamoDB Basics

Understanding the Basics

In DynamoDB, tables, items, and attributes are the core components that you work with.

A Table is a collection of items, and each item is a collection of attributes.



Importance of Primary Key

Each item in the table has a unique identifier, or primary key, that distinguishes the item from all of the others in the table

Other than the primary key, the table is schemaless, which means that neither the attributes nor their data types need to be defined beforehand.

Primary Key



UserID	Name	Age	Interests
1	Harsh	30	Travelling
2	Zeal	30	Astronomy

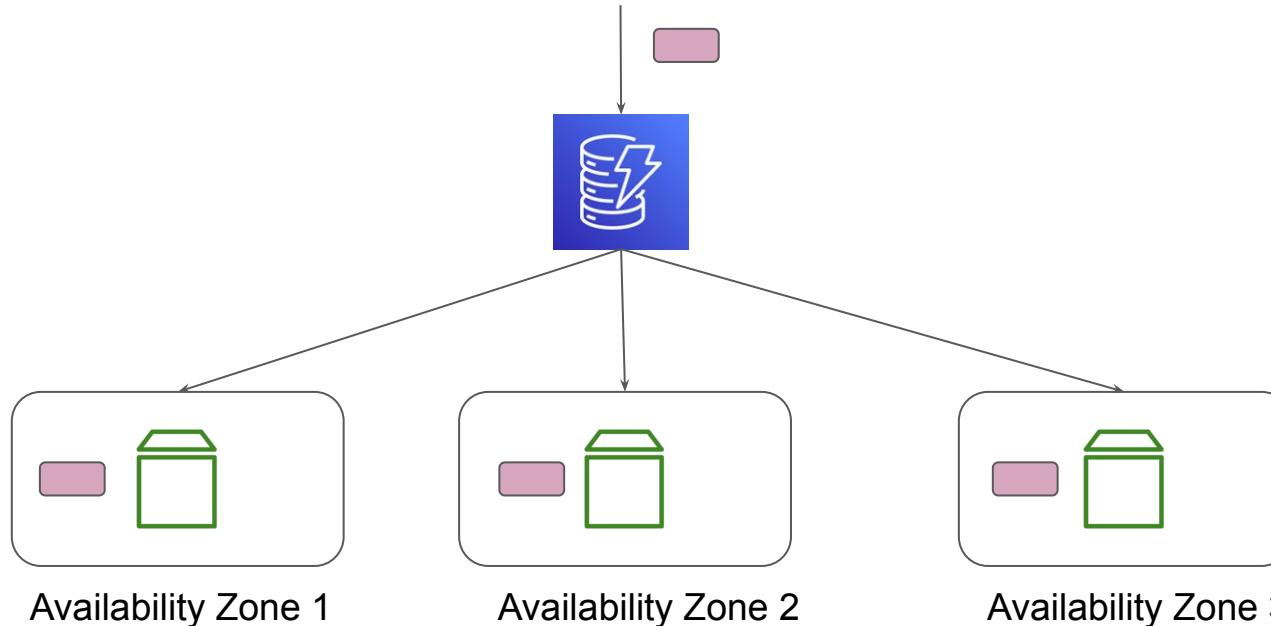
DynamoDB Table

Consistency Model

Important Storage Concept

Understanding Consistency Model

In DynamoDB, all of your data is stored on SSDs and is automatically replicated across multiple Availability Zones in an Amazon Region, providing built-in high availability and data durability.



Consistency Timeline

When your application writes data to a DynamoDB table and receives an HTTP 200 response (OK), the write has occurred and is durable.

The data is eventually consistent across all storage locations, usually within one second or less.

Eventual Consistency Reads

When you read data from a DynamoDB table, the response might not reflect the results of a recently completed write operation.

The response might include some stale data.

If you repeat your read request after a short time, the response should return the latest data.

Strong Consistency Reads

When you request a strongly consistent read, DynamoDB returns a response with the most up-to-date data, reflecting the updates from all prior write operations that were successful.

1. A strongly consistent read might not be available if there is a network delay or outage. In this case, DynamoDB may return a server error (HTTP 500).
2. Strongly consistent reads may have higher latency than eventually consistent reads.
3. Strongly consistent reads use more throughput capacity than eventually consistent reads.

Important Note

DynamoDB uses eventually consistent reads, unless you specify otherwise.

Read operations (such as GetItem, Query, and Scan) provide a ConsistentRead parameter. If you set this parameter to true, DynamoDB uses strongly consistent reads during the operation.

Example Command:

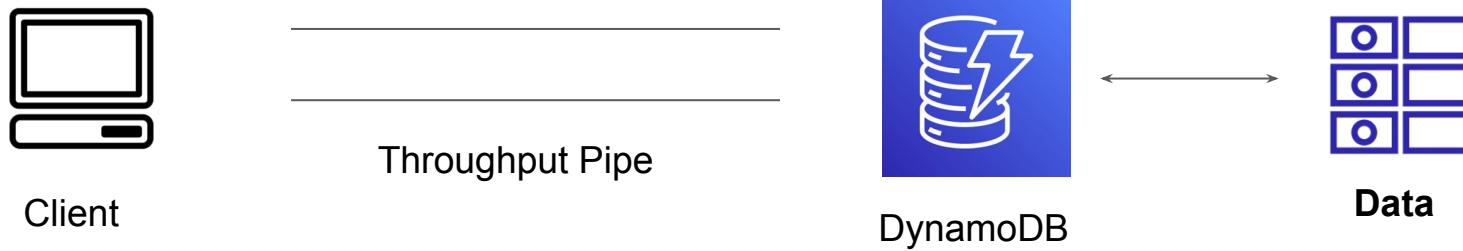
```
aws dynamodb get-item --table-name MusicCollection --key file://key.json  
--consistent-read
```

Read/Write Capacity Units

Managing Throughput

Throughput in DynamoDB

Throughput is the maximum amount of capacity that an application can consume from a table or index



Setting the Read & Write Capacity

We can specify throughput capacity in terms of read capacity units (RCUs) and write capacity units.

*

Read capacity

Auto scaling [Info](#)
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

On
 Off

Provisioned capacity units
5

Write capacity

Auto scaling [Info](#)
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

On
 Off

Provisioned capacity units
5

Read Request Unit

One read request unit represents one strongly consistent read request, or two eventually consistent read requests, for an item up to 4 KB in size.

If you need to read an item that is larger than 4 KB, DynamoDB needs additional read request units.

Item Size	Read Capacity Unit (Strong)	Read Capacity Unit (Eventual)
4 KB	1	1
8 KB	2	1
10 KB	3	2

Write Request Unit

One write request unit represents one write for an item up to 1 KB in size.

If you need to write an item that is larger than 1 KB, DynamoDB needs to consume additional write request units.

Item Size	Write Capacity Unit
1 KB	1
4 KB	4
10 KB	10

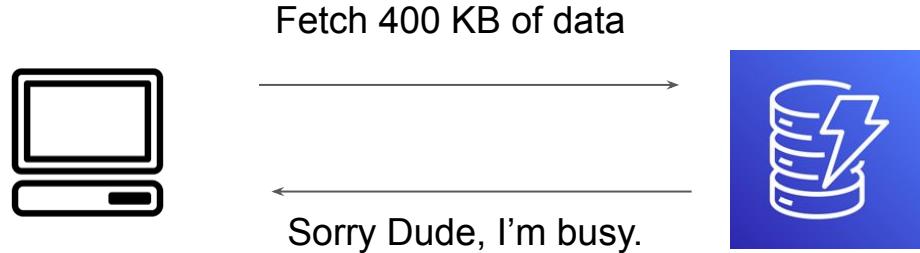
Capacity Modes in DynamoDB

Adjust Throughput Automatically

Understanding the Challenge

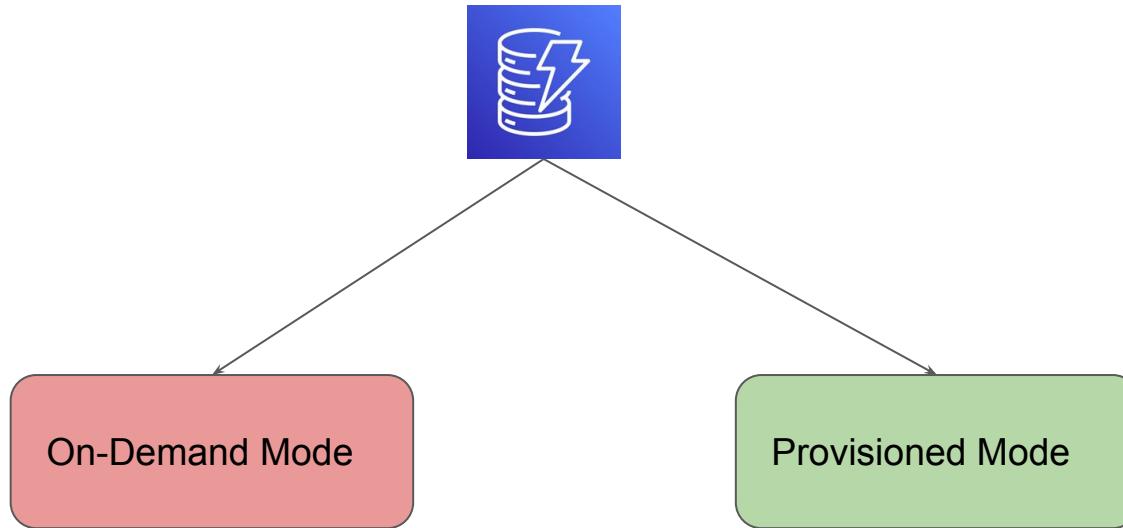
We can specify throughput capacity in terms of read capacity units (RCUs) and write capacity units

If your application exceeds your provisioned throughput capacity on a table or index, it is subject to request throttling.



Types of Capacity Modes

There are two primary capacity modes available in DynamoDB.



Provisioned Mode

If you choose provisioned mode, you specify the number of reads and writes per second that you require for your application.

You can use auto scaling to adjust your table's provisioned capacity automatically in response to traffic changes.

Read capacity

Auto scaling | [Info](#)
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

On
 Off

Minimum capacity units	Maximum capacity units	Target utilization (%)
1	10	70

Write capacity

Auto scaling | [Info](#)
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

On
 Off

Minimum capacity units	Maximum capacity units	Target utilization (%)
1	10	70

Recommended Traffic Patterns for Provisioned Mode

Provisioned mode is a good option if any of the following are true:

- You have predictable application traffic.
- You run applications whose traffic is consistent or ramps gradually.
- You can forecast capacity requirements to control costs.

On-Demand Mode

Amazon DynamoDB on-demand is capable of serving thousands of requests per second without capacity planning.

DynamoDB on-demand offers pay-per-request pricing for read and write requests so that you pay only for what you use.



On-Demand



Provisioned
Auto-Scaling

imgflip.com

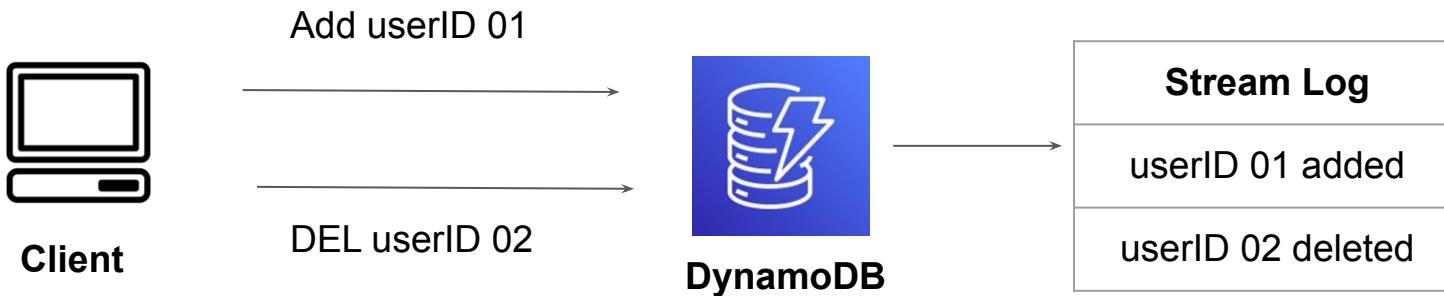
DynamoDB Streams

Stream Records Real-Time

Understanding the Basics

DynamoDB Streams captures a time-ordered sequence of item-level modifications in any DynamoDB table and stores this information in a log for up to 24 hours.

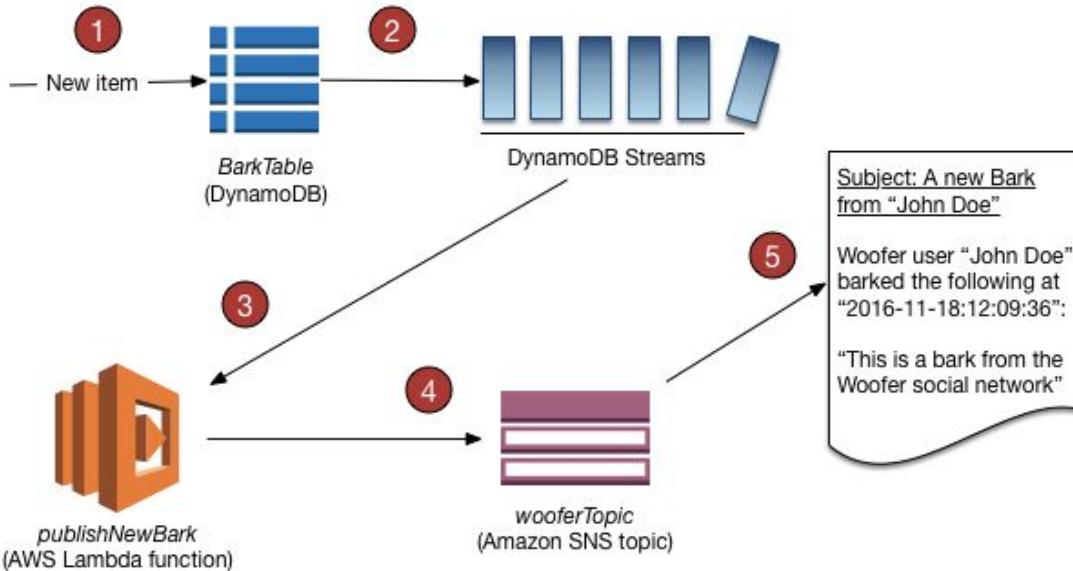
Applications can access this log and view the data items as they appeared before and after they were modified, in near-real time.



Sample Record Log in CloudWatch

```
▶ 2022-07-15T21:06:51.711+05:30 REMOVE
▶ 2022-07-15T21:06:51.711+05:30 DynamoDB Record: {
▶ 2022-07-15T21:06:51.711+05:30 "ApproximateCreationDateTime": 1657899411.0,
▶ 2022-07-15T21:06:51.711+05:30 "Keys": {
▶ 2022-07-15T21:06:51.711+05:30 "userID": {
▶ 2022-07-15T21:06:51.711+05:30 "S": "01"
▶ 2022-07-15T21:06:51.711+05:30 }
▶ 2022-07-15T21:06:51.711+05:30 },
▶ 2022-07-15T21:06:51.711+05:30 "OldImage": {
▶ 2022-07-15T21:06:51.711+05:30 "courseName": {
▶ 2022-07-15T21:06:51.711+05:30 "S": "AWS Certification Course"
▶ 2022-07-15T21:06:51.711+05:30 },
▶ 2022-07-15T21:06:51.711+05:30 "userID": {
▶ 2022-07-15T21:06:51.711+05:30 "S": "01"
```

A Sample Use-Case



Use-Cases

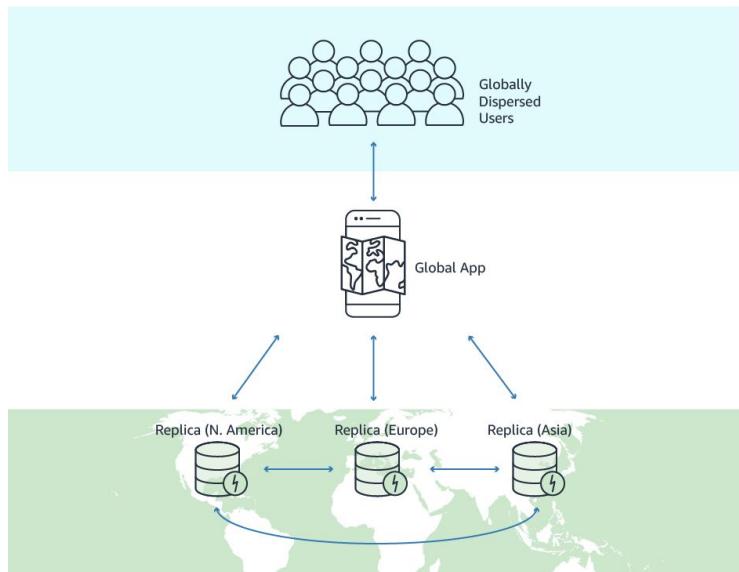
1. Allows setting up a relationship across multiple tables in which, based on the value of an item from one table, you update the item in a second table
2. Triggering an event based on a particular item change
3. Audit or Archive Data
4. Replicating Data Across Multiple Tables

DynamoDB - Global Table

Let's Replicate

Basics of Global Tables

Global tables feature provides us with a fully managed, multi-Region, and multi-active database that delivers fast, local, read and write performance for massively scaled, global applications.



Basic Terminology

A global table is a collection of one or more replica tables, all owned by a single AWS account.

A replica table is a single DynamoDB table that functions as a part of a global table. Each replica stores the same set of data items.

When an application writes data to a replica table in one Region, DynamoDB propagates the write to the other replica tables in the other AWS Regions automatically.

Important Pointers

In a global table, a newly-written item is usually propagated to all replica tables within seconds.

With a global table, each replica table stores the same set of data items. DynamoDB does not support partial replication of only some of the items.

Conflicts can arise if applications update the same item in different regions at about the same time. To ensure eventual consistency, DynamoDB global tables use a “last writer wins”

DynamoDB Accelerator (DAX)

Let's Accelerate Read Requests

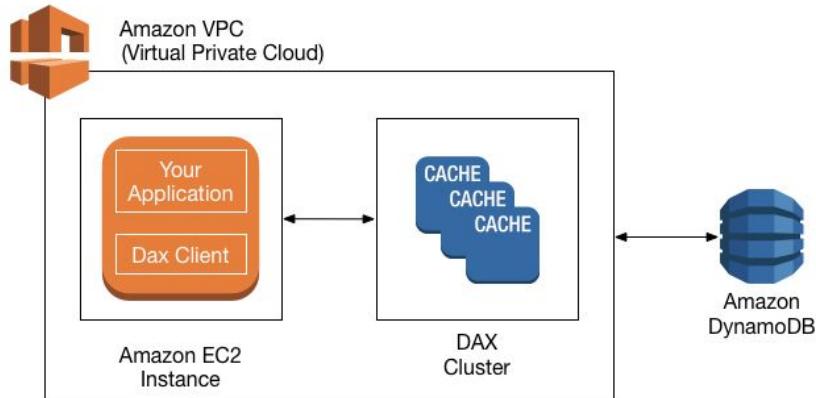
Understanding the Need

In most cases, the DynamoDB response times can be measured in single-digit milliseconds. However, there are certain use cases that require response times in microseconds.

For these use cases, DynamoDB Accelerator (DAX) delivers fast response times for accessing eventually consistent data.

Overview of the Feature

Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for Amazon DynamoDB that delivers up to a 10x performance improvement.



Use-Case for DAX

Applications that require the fastest possible response time for reads. Some examples include real-time bidding, social gaming, and trading applications.

Applications that read a small number of items more frequently than others

Applications that are read-intensive, but are also cost-sensitive.

Where it is not suitable for ?

DAX is not ideal for the following types of applications:

Applications that require strongly consistent reads (or that cannot tolerate eventually consistent reads).

Applications that do not require microsecond response times for reads, or that do not need to offload repeated read activity from underlying tables.

Applications that are write-intensive, or that do not perform much read activity.

AWS Backup



Understanding with Use-Case

AWS has lots of services where data can be stored.

For production environment, data backup is one of the critical task.

Taking backup at individual service level can take lot of time and require customization.



RDS



DynamoDB



S3

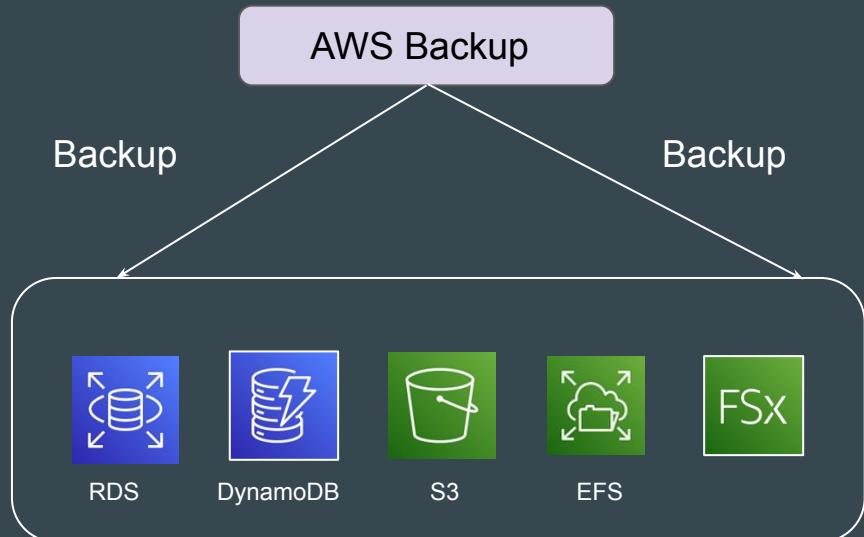


EFS



Introducing AWS Backup

AWS Backup is a fully-managed service that allows customers to configure backup policies in one central place.



Benefits of using AWS Backup

- Easily create backup rules for daily, monthly backups.
 - Backup Process is automated at a scheduled time.
 - Supports Cross-Region, Cross-Account Backups.
-
- AWS Backup can back up on-premises Storage Gateway volumes and VMware virtual machines
-
- Supports Retention Period that tells how long to store backup.

DR Techniques

Let's DR

How to design ?

There can be various disaster recovery design that we can implement, this directly depends on how quickly we want to recover from a disaster, in short RTO and RPO.

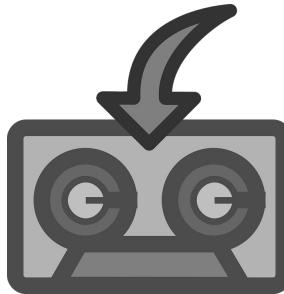
Broadly classified into 4 types :

- i) Backup & Restore
- ii) Pilot Light
- iii) Warm Standby
- iv) Multi-Site

Scenario 1 - Backup / Recovery

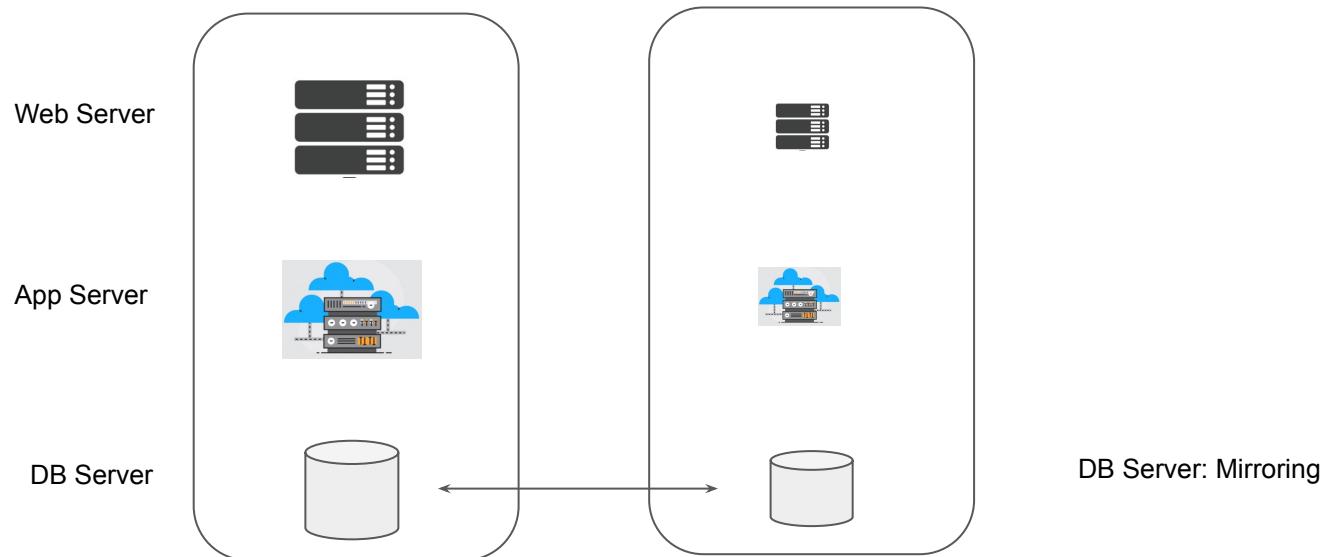
Based on simple / cost effective method which requires us to constantly take backup of our data and store it to service like S3 and restore when disaster strikes.

For on-premise server with huge amount of data typically in tens of terabytes, then can use technology like direct connect or import/export to backup their data to AWS.



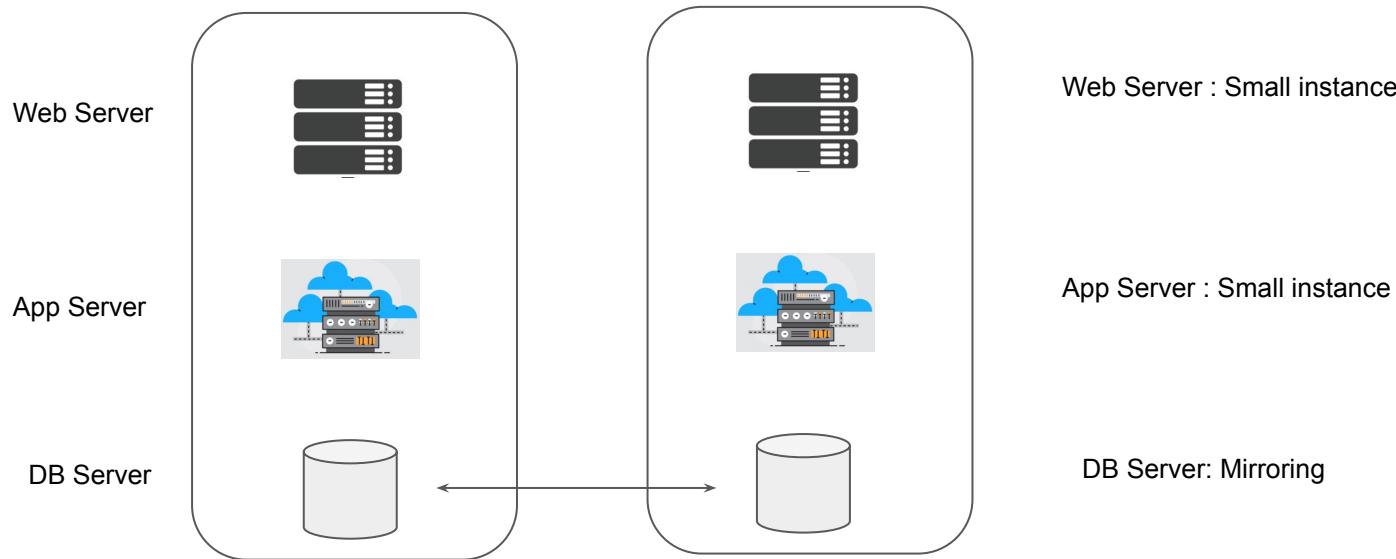
Scenario 2 - Pilot Light

Minimal version of server in stopped state or AMI present.



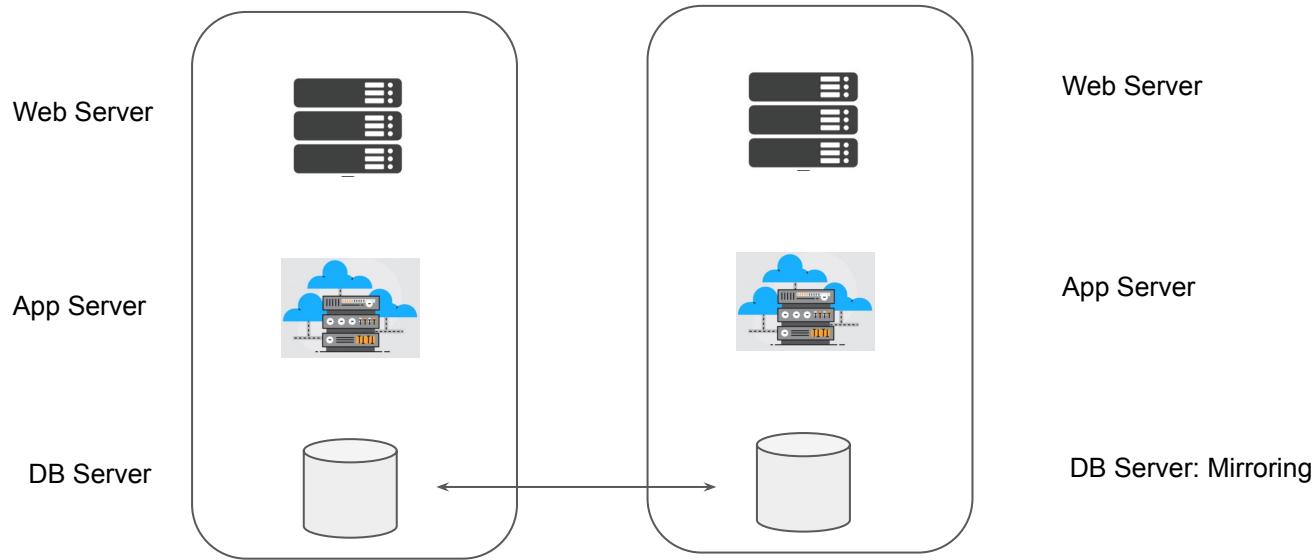
Scenario 3 - Warm Standby

- Server runs with minimal sizes.
- When disaster happens, the server are scaled up for production.



Scenario 4 - Multi-Site

- Complete 1 to 1 mirror of your production environment.



AWS Services for DR

We can use plethora of AWS services as part of the disaster recovery which includes :

AWS S3

AWS Glacier

Import / Export

Elastic Block Store (EBS)

AWS Storage Gateway

Direct Connect

RDS

VM Import / Export

AWS ElasticBeanStalk

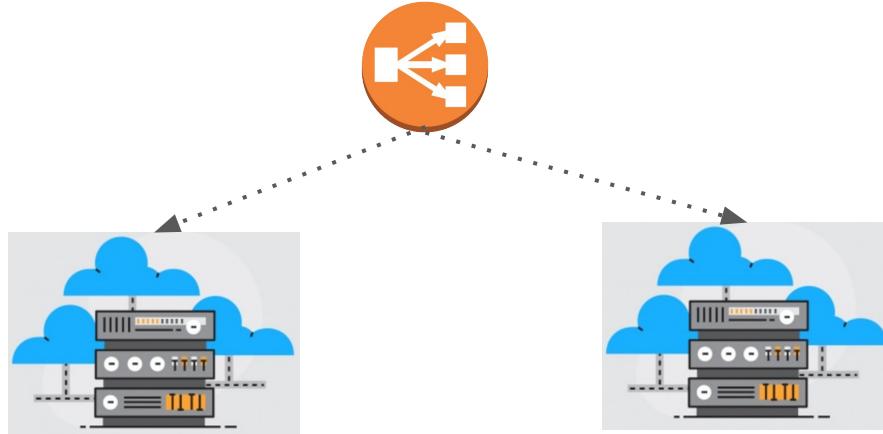
Route 53

RTO & RPO

Health should always be good

Everything comes at price

- High Availability Architecture is driven by your requirements.
- An highly available, multi-AZ, fault tolerant infrastructure is certainly possible, however there is cost associated with it.



Recovery Time Objective

- Recovery Time Objective (RTO) is the amount of time frame it takes for you to recover your infrastructure and business operations after disaster has struck.

Sample Example:

- If RTO is 3 hours, then one needs to invest quite good amount of money to make sure DR region is always ready in-case main region goes down due to disaster.

Recovery Point Objective

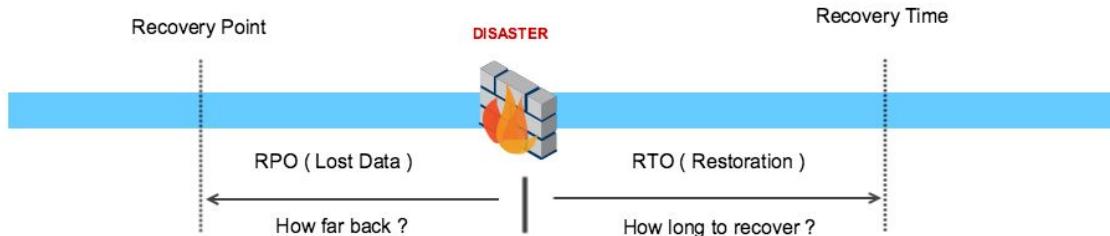
- Recovery Point Objective (RPO) is concerned with data and maximum tolerance period to which data can be lost.
- It helps in determining how well we should be designing the infrastructure.

Sample Example:

- If RPO is 5 hours for database, then we should be taking backup of database every five hours .

RTO vs RPO

- RTO is more broader scope and covers whole business and systems involved while RPO is more directly related to interval of backup to take to avoid data loss.



Advance Route53 Features

Interesting Features of Route53

Managed DNS Providers

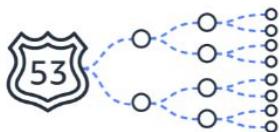
Generally a managed DNS server supports basic functionality like :

- Domain Registration
- GUI for putting DNS records
- Mapping & Resolving various DNS Records.
- WHOIS Management

The screenshot shows a domain management interface. On the left, a sidebar with a green header titled 'Details' contains links for Contacts, Nameservers, DNS Records, URL Forwarding, Email Forwarding, NS Registration, and Account Transfer. To the right, under 'Domain Details', is information for the domain kplabs.in. It includes:
- Domain name: kplabs.in
- Domain lock: Locked | [Unlock](#)
- Transfer Auth Code: [Show Code](#)
- Nameservers: [Edit Nameservers](#) (listing ns1.name.com, ns2.name.com, ns3.name.com, ns4.name.com)
- DNS hosted: Yes [Update DNS records](#)
- Registrar: name.com
- Website hosted: No
- Automatic Renewal: Enabled [Edit](#)
- Whois Privacy: N/A

Route53 does a lot more

- Support of Public and Private Hosted Zones.
- Routing - Weighted, Latency, Geolocation, Round Robin
- Health Checks & Monitoring
- Route53 Endpoints
- DNS Firewall



Traffic Flow



Hosted Zone



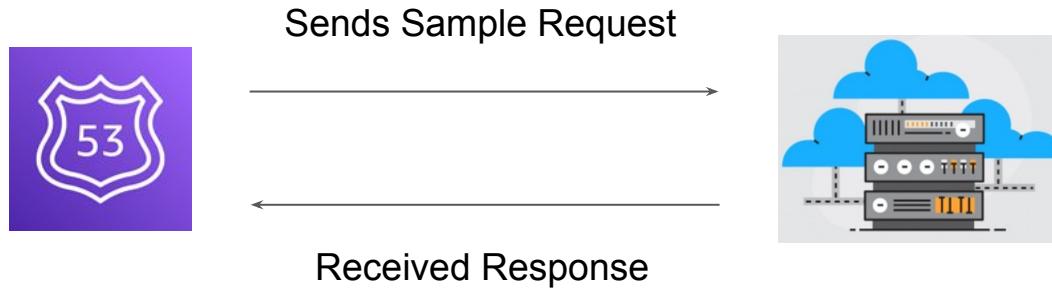
Health Checks

Route53 Health Checks

Back to Monitoring!

Overview of Health Checks

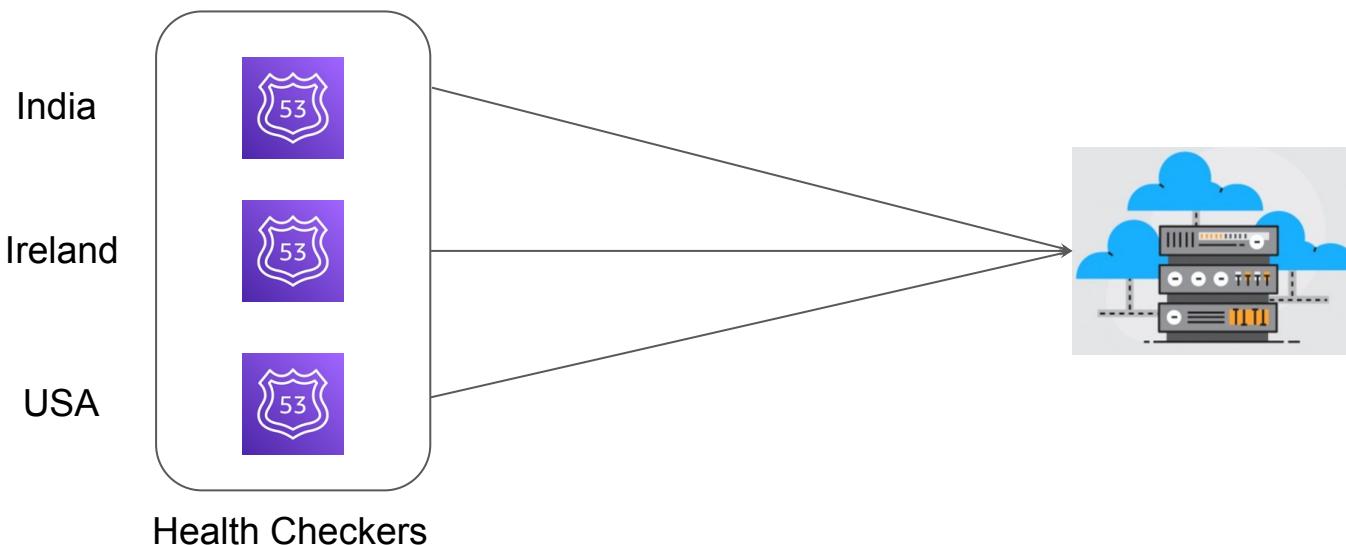
Amazon Route 53 health checks monitor the health and performance of your web applications, web servers, and other resources



Route53 Health Checkers

Route 53 has health checkers in locations around the world.

When you create a health check that monitors an endpoint, health checkers start to send requests to the endpoint that you specify to determine whether the endpoint is healthy.



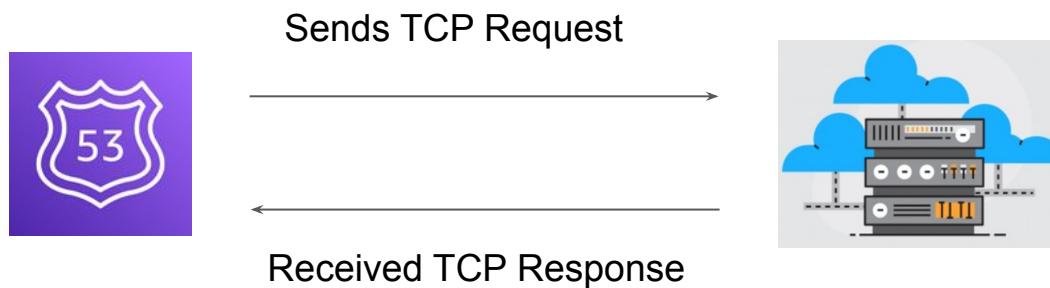
Types of Health Checks

Back to Monitoring!

Type of Health Checks

There are three primary type of Health Checks supported by Route53

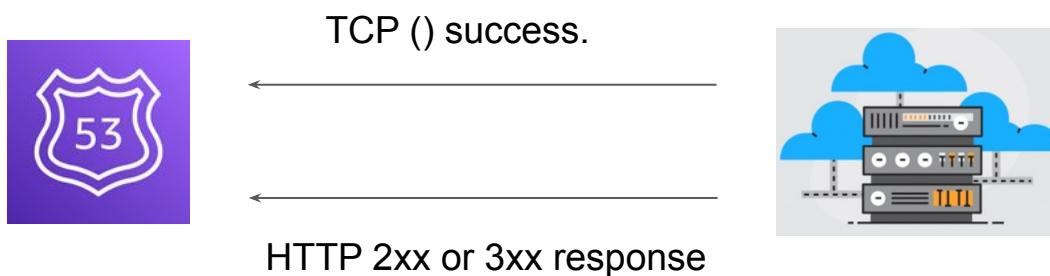
1. HTTP and HTTPS health checks
2. TCP health checks
3. HTTP and HTTPS health checks with string matching



Type 1 - HTTP/HTTPS

Two important factors as part of this health check:

1. Route 53 must be able to establish a TCP connection with the endpoint within four seconds.
2. In addition, the endpoint must respond with an HTTP status code of 2xx or 3xx within two seconds after connecting.



Type 2 - TCP

Route 53 must be able to establish a TCP connection with the endpoint within ten seconds.

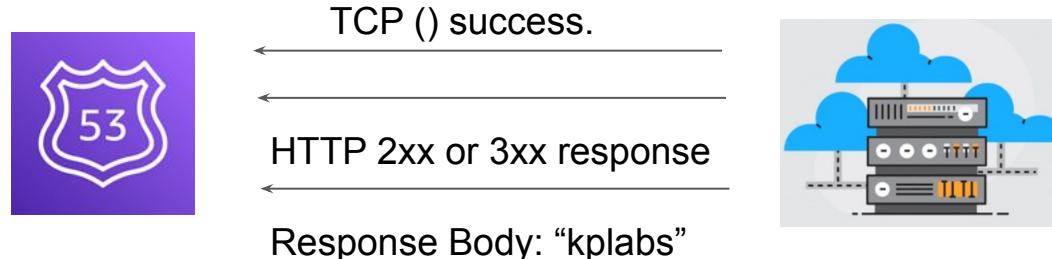


TCP () success.



Type 3 - HTTP/HTTPS with string matching

1. Route 53 must be able to establish a TCP connection with the endpoint within four seconds.
2. In addition, the endpoint must respond with an HTTP status code of 2xx or 3xx within two seconds after connecting.
3. Must receive response body within next two seconds containing a specific string.
4. The string must appear entirely in the first 5,120 bytes of the response body or the endpoint fails the health check.



Routing Policies

Great DNS Provider

Routing Policies

Routing Policies determine how Amazon Route53 responds to the queries.

There are various supported routing policies available in Route53.

Each policy supports a specific use-case.

- Simple
- Weighted
- Latency
- Failover
- Geolocation
- Multi-value answer



Simple Routing Policy

In simple routing, there is a plain one to one mapping between domain and host.

Example: blog.kplabs.internal A 128.199.241.125

Quick create record [Info](#) [Switch to wizard](#) [Add another record](#)

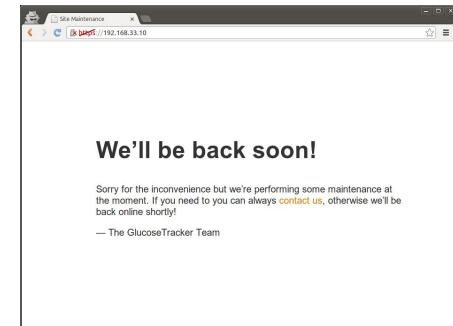
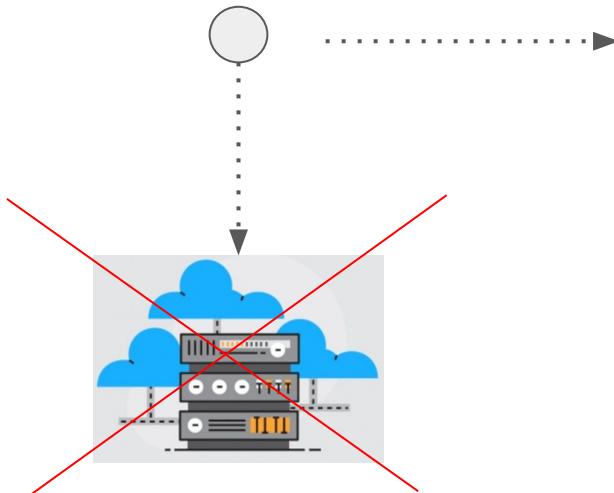
▼ Record 1

Record name Info blog.kplabs.internal	Record type Info A – Routes traffic to an IPv4 address and so...	Value Info 10.77.20.50 <small>Enter multiple values on separate lines.</small>
Valid characters: a-z, 0-9, ! " # \$ % & ' () * + , - / ; < = > ? @ [\] ^ _ ` { } . ~	<input checked="" type="radio"/> Alias	
TTL (seconds) Info 300	Routing policy Info Simple routing	
<input type="button" value="1m"/> <input type="button" value="1h"/> <input type="button" value="1d"/>	Recommended values: 60 to 172800 (two days)	

[Delete](#) [Cancel](#) [Create records](#)

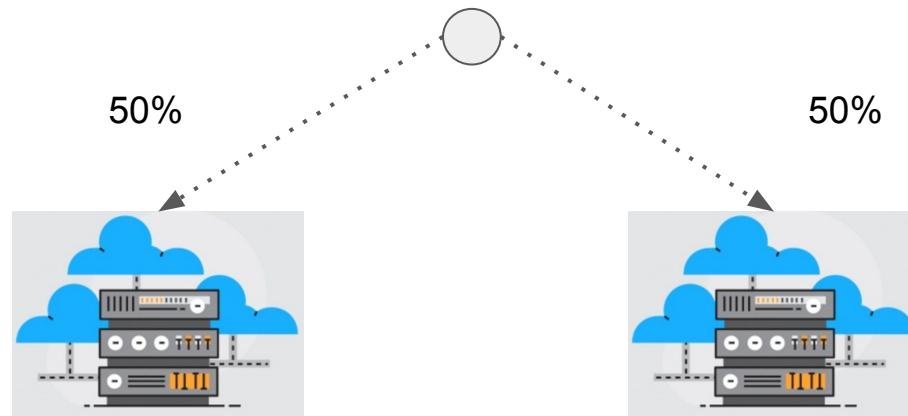
Failover Routing

Failover routing lets you route traffic to a resource when the resource is healthy or to a different resource when the first resource is unhealthy.



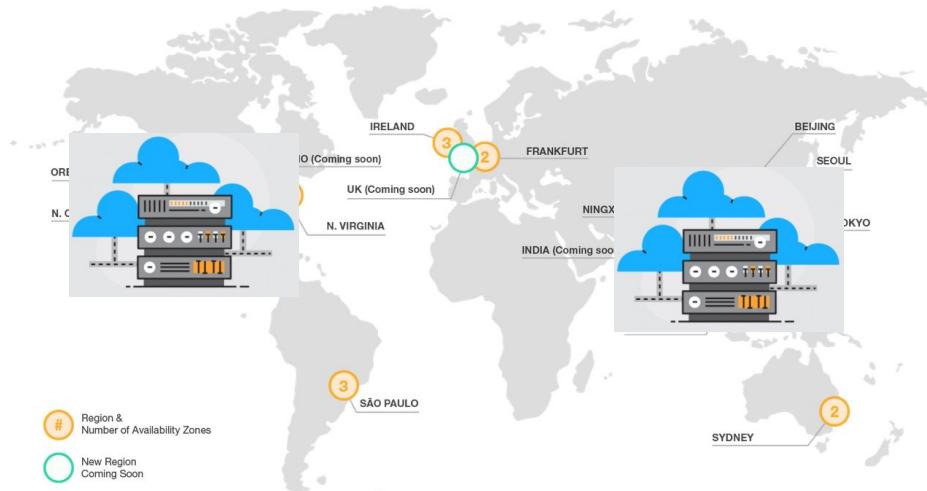
Weighted Routing

Weighted routing helps us to route the traffic to multiple resources in a proportion that we specify from our end.



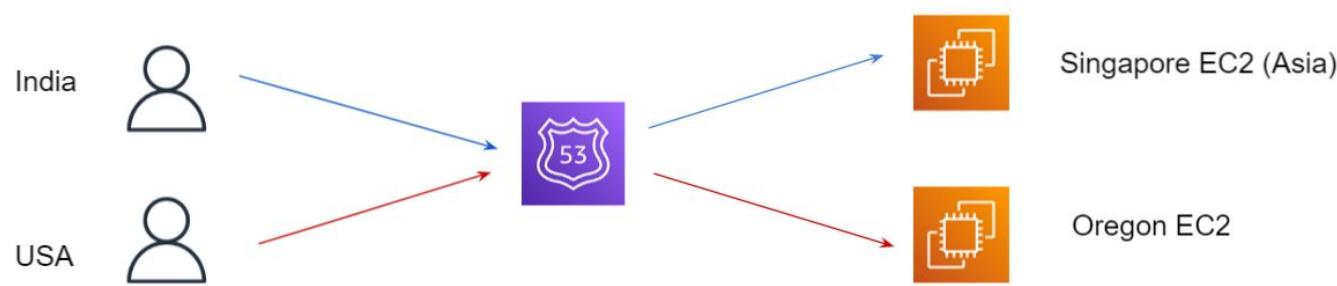
Latency Based Routing

If your application is hosted in multiple AWS regions, we can improve the performance for the users by serving their request from AWS region that provides lowest latency.



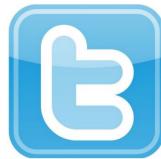
GeoLocation Routing

Geolocation routing allows us to choose different resources for different users based on different countries / continents.



Join us in our Adventure

Be Awesome



kplabs.in/twitter



kplabs.in/linkedin

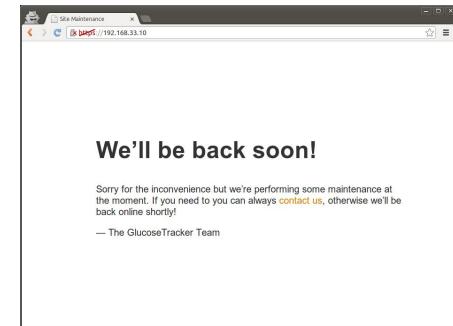
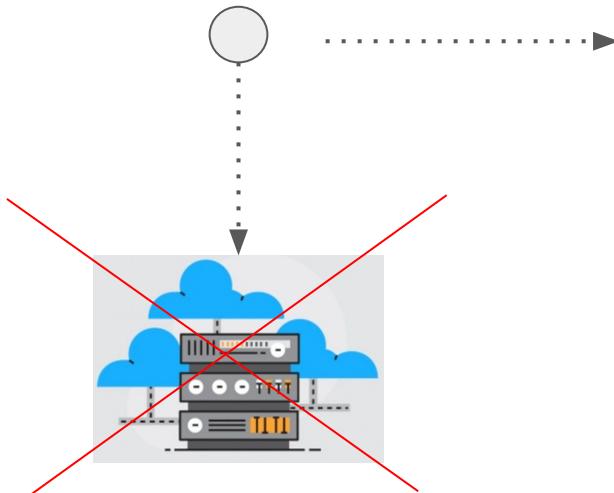
instructors@kplabs.in

Failover Routing Policy

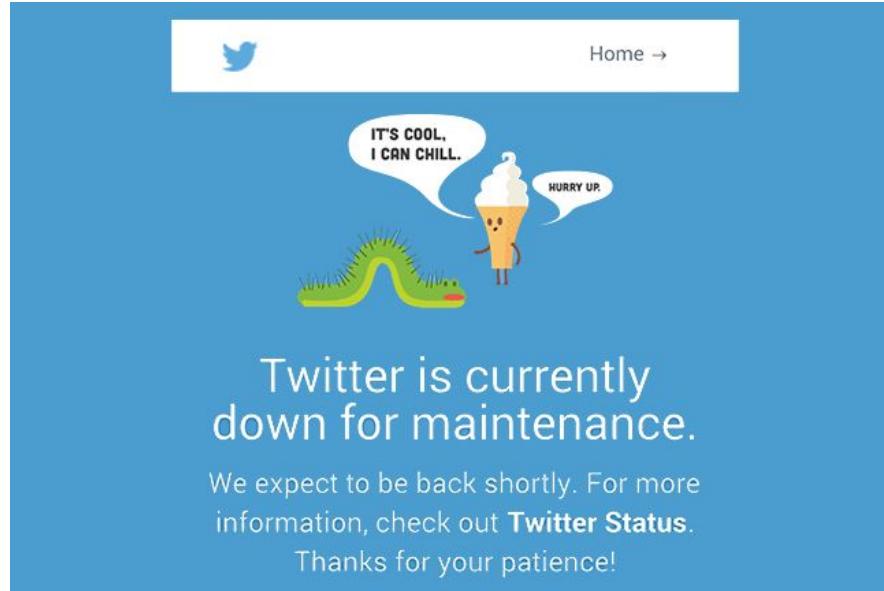
Back to Monitoring!

Failover Routing

Failover routing lets you route traffic to a resource when the resource is healthy or to a different resource when the first resource is unhealthy.



Maintenance Page



Relax and Have a Meme Before Proceeding

Before the war between Android and Apple started, the war between 0.5 and 0.7 users was the one that separate humans from each other.



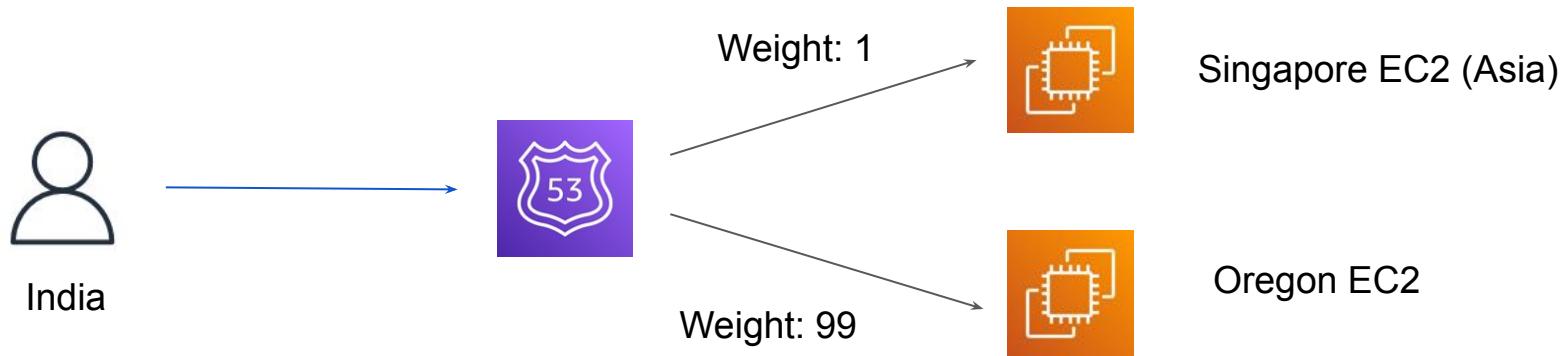
Weighted Routing

Route53 Routing Policy

Overview of Weighted Routing Policy

Weighted Routing allows us to specify the proportion in which traffic should be routed to the underlying servers.

If we want to send small portion of traffic to a new website theme, you can specify the weight of 1 and 99. The resource with 1 gets 1% of the traffic and other gets 99% of traffic.



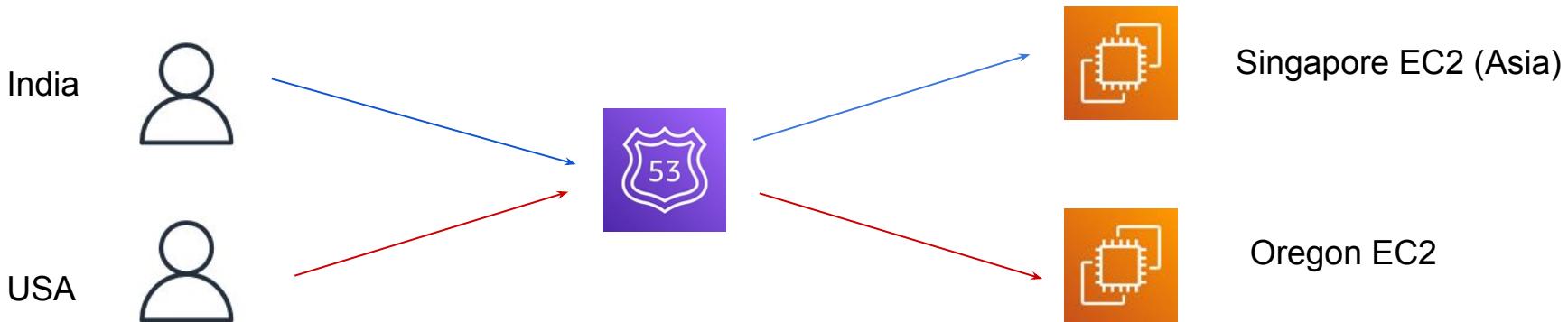
Geolocation Routing

Route53 Routing Policy

Overview of Geolocation Routing

Geolocation Routing allows us to choose resources based on the geographic location of the users

For example, you might want all queries from Asia to be routed to an ELB load balancer in the Singapore region.



Important Caution

Geolocation Routing works by mapping database to IP address.

The **results are not always accurate** as some ISP might not have any geolocation data associated with them, and some ISP might move the IP block to different country without notification.

For such cases, Route 53 allows us to have a default resource block associated with the routing policy.

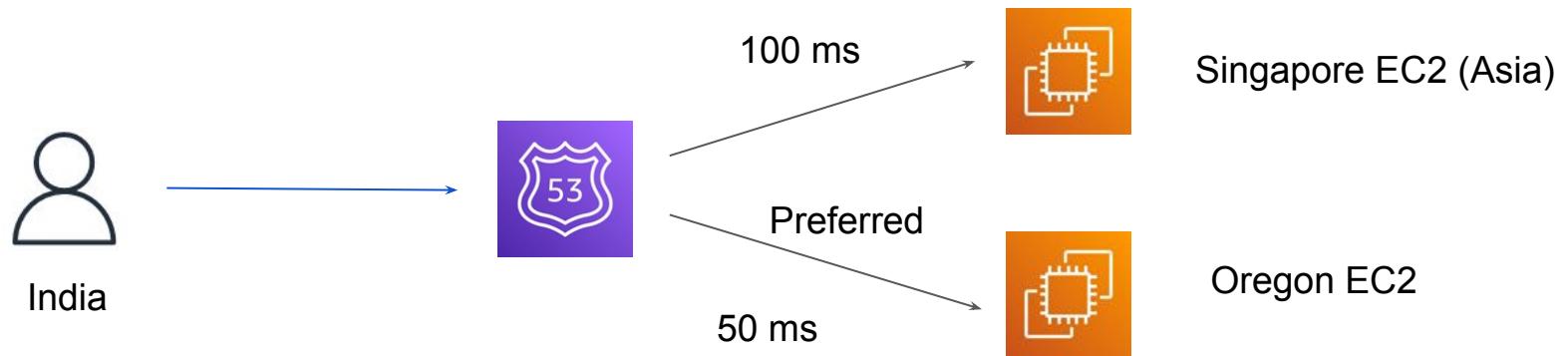
Latency Routing

Route53 Routing Policy

Overview of Latency Based Routing

If your application is hosted in multiple AWS regions, we can improve the performance for the users by serving their request from AWS region that provides lowest latency.

A request that is routed to Singapore today might be routed to India tomorrow.



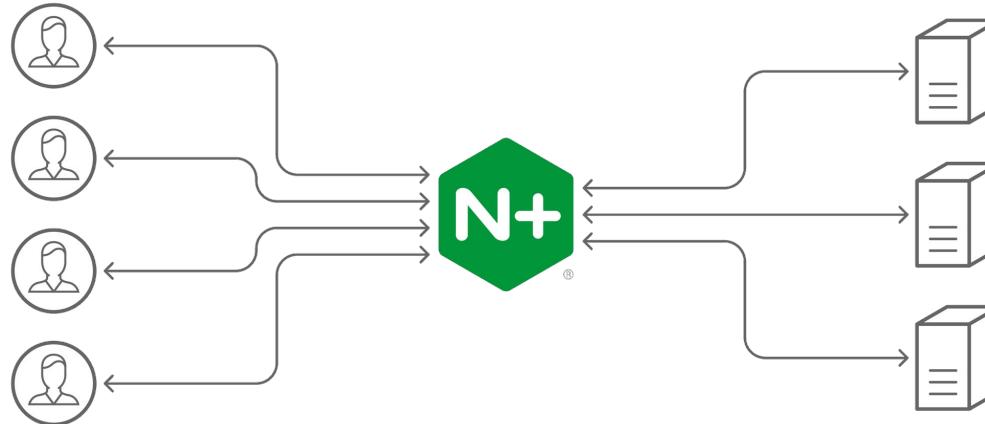
Load Balancing in AWS

Let's Load Balance Traffic in AWS

Basics of Load Balancing

There are multiple software and hardware based load balancing solutions available.

Some of the popular ones include Nginx, HA Proxy and others.



Challenges with Maintaining Load Balancing Solution

If you are using a load balancing solution, various responsibilities falls to customer.

Some of these include:

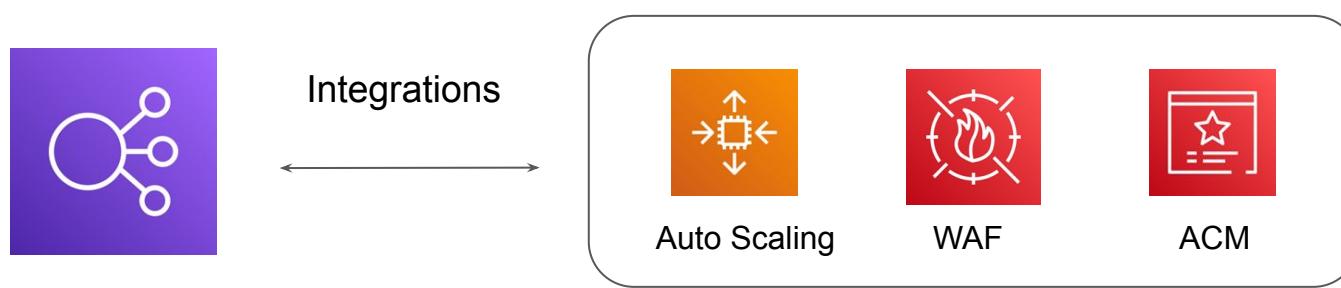
1. High-Availability of Load Balancers.
2. Security.
3. Performance.

Basics of Elastic Load Balancing Service

AWS offers managed load balancing solutions for wide variety of use-cases.

These solutions are offered under the Elastic Load Balancing feature.

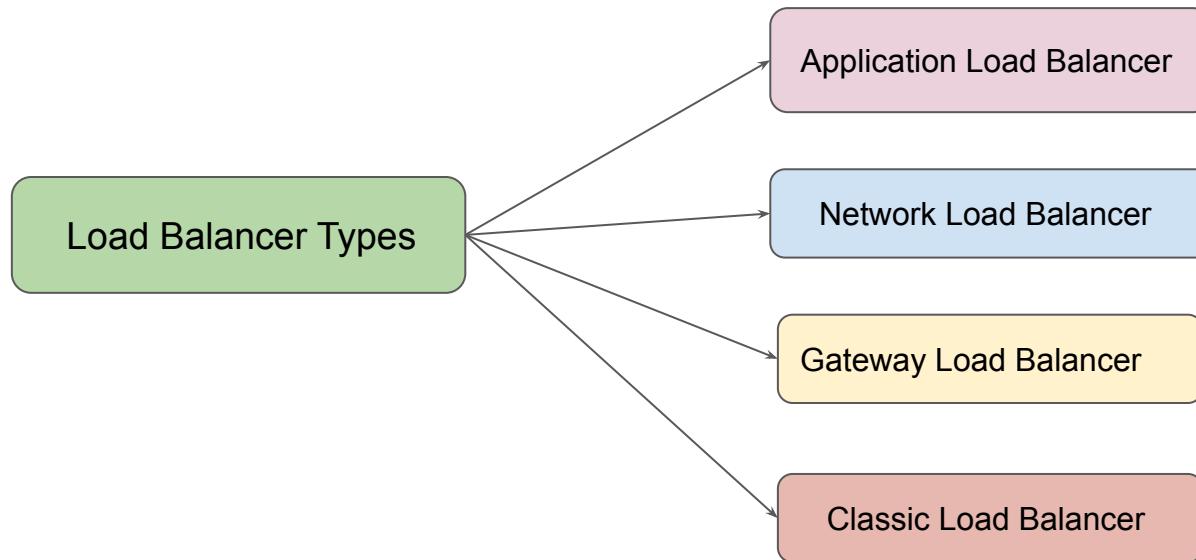
Tight integration with multiple AWS Services.



Elastic Load Balancing

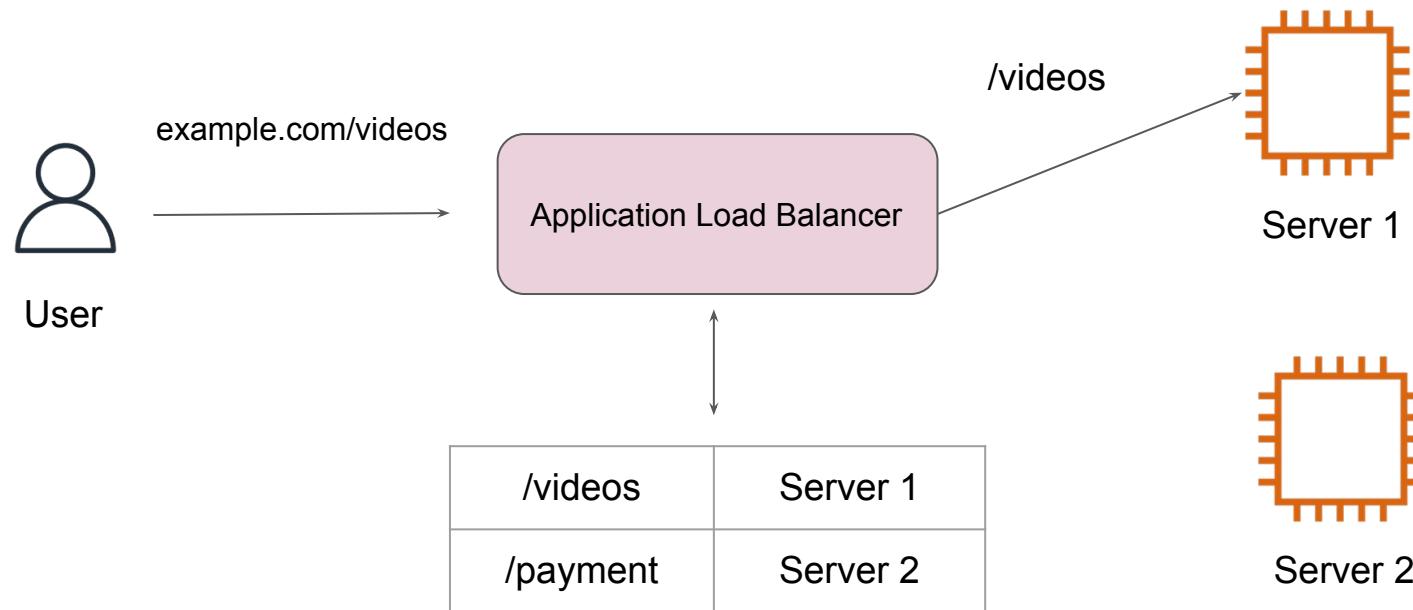
Types of Load Balancers

There are 4 primary type of Load Balancer offerings available.



Application Load Balancers

An Application Load Balancer makes routing decisions at the application layer (HTTP/HTTPS)

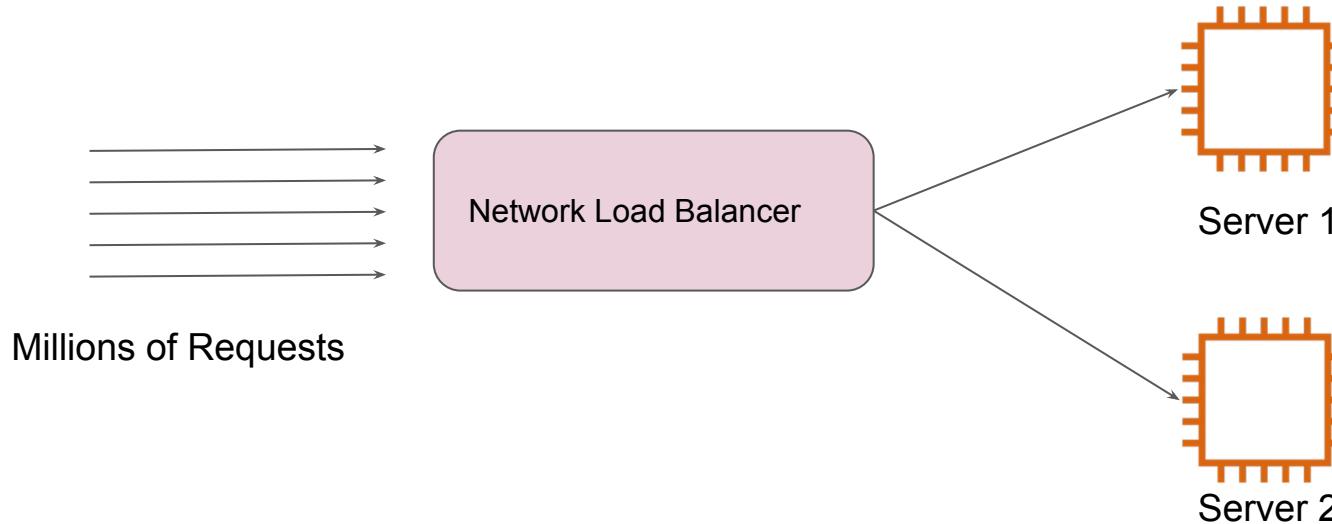


Network Load Balancers

A Network Load Balancer makes routing decisions at the transport layer (TCP/UDP/SSL).

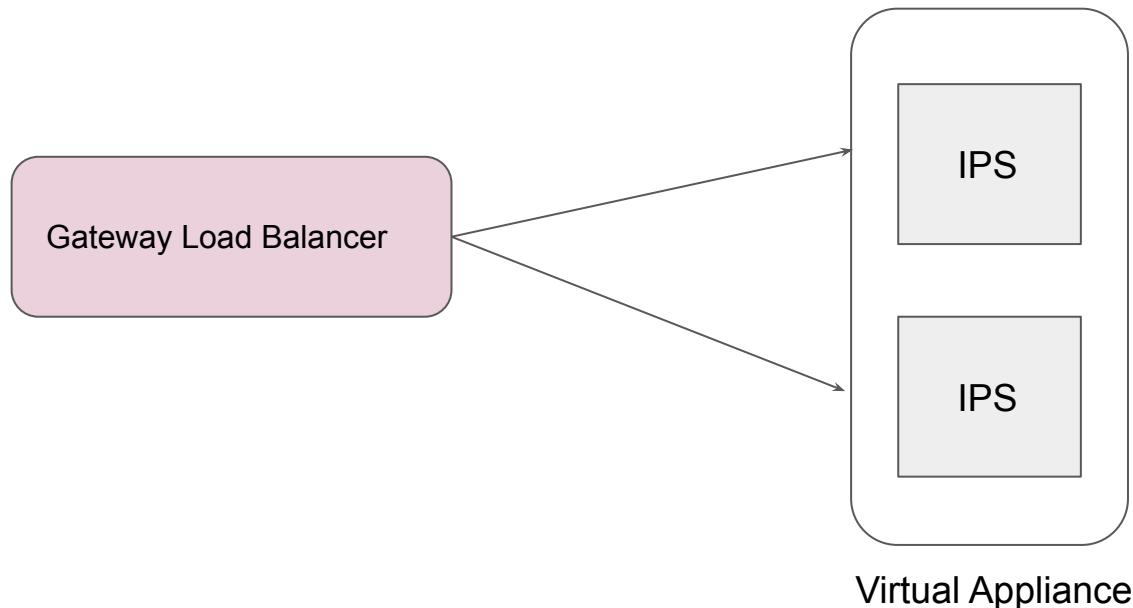
It can handle millions of requests per second.

Not all of the applications work on HTTP/HTTPS protocol.



Gateway Load Balancers

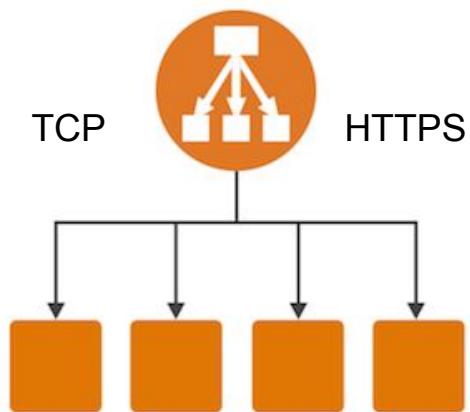
Gateway Load Balancers allow you to deploy, scale, and manage virtual appliances, such as firewalls, intrusion detection and prevention systems, and deep packet inspection systems



Classic Load Balancers

A Classic Load Balancer makes routing decisions at either the transport layer (TCP/SSL) or the application layer (HTTP/HTTPS).

Previous Generation Load Balancer and not recommended.



Summary Slide

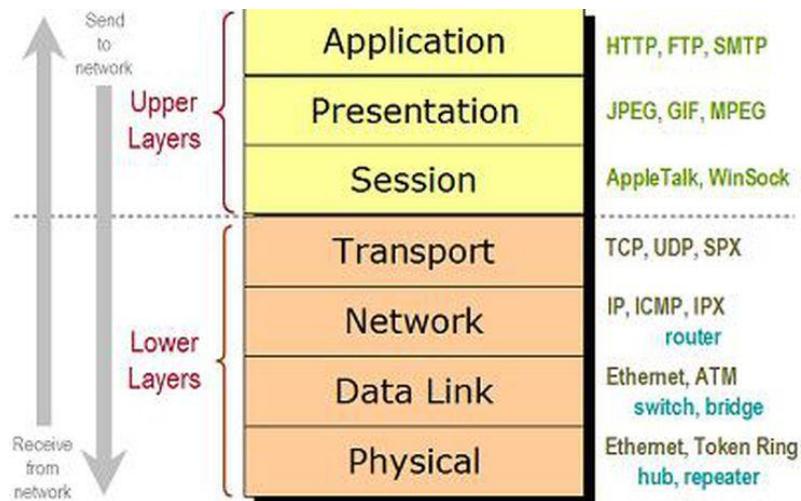
Load Balancer	Important Notes
Application Load Balancer	Use when you have websites/applications at L7 (HTTP/HTTPS)
Network Load Balancers	TCP and UDP based applications. Requirement to handle millions of requests per second. Ultra high performance.
Gateway Load Balancer	Use when you have virtual appliances: IDS/IPS Firewalls

OSI Model & Load Balancers

Revising Networking

Basics of OSI Model

The Open Systems Interconnection (OSI) model describes seven layers that computer systems use to communicate over a network. It



Load Balancer & OSI Layers

Each load balancer operates at a specific layer.

You will only be able to perform operations on requests based on Layer the ELB supports.

Feature	Application Load Balancer	Network Load Balancer	Gateway Load Balancer	Classic Load Balancer
Load Balancer type	Layer 7	Layer 4	Layer 3 Gateway + Layer 4 Load Balancing	Layer 4/7

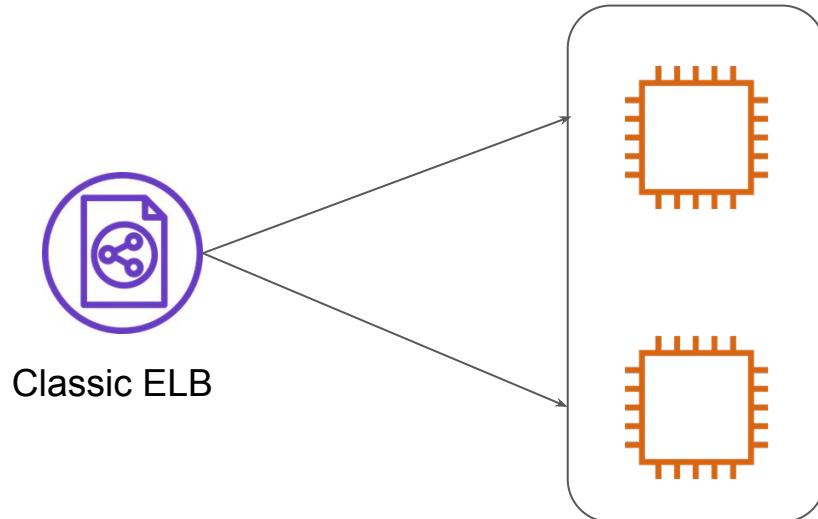
Classic Load Balancers

First generation Load Balancers

Understanding Classic Load Balancers

These are older generation of load balancers.

Provides basic set of features for HTTP, HTTPS, TCP and SSL protocols.



Limitation of Classic Load Balancers

- Does not support native HTTP/2 protocol.
- IP address as targets are not supported.
- Path based routing is not supported. (eg: /images should go to server 1 & /php to server 02)
- Many Many more

Application Load Balancers

Next generation load balancers

Basics of HTTP Headers

HTTP headers let the client and the server pass additional information with an HTTP request or response.

▶ GET http://demo-alb-137613815.us-east-1.elb.amazonaws.com/

Status	200 OK ⓘ
Version	HTTP/1.1
Transferred	196 B (35 B size)
Request Priority	Highest

▼ Response Headers (161 B)

- ⓘ Connection: keep-alive
- ⓘ Content-Length: 35
- ⓘ Content-Type: text/plain; charset=utf-8
- ⓘ Date: Thu, 21 Jul 2022 16:49:49 GMT
- ⓘ Server: awselb/2.0

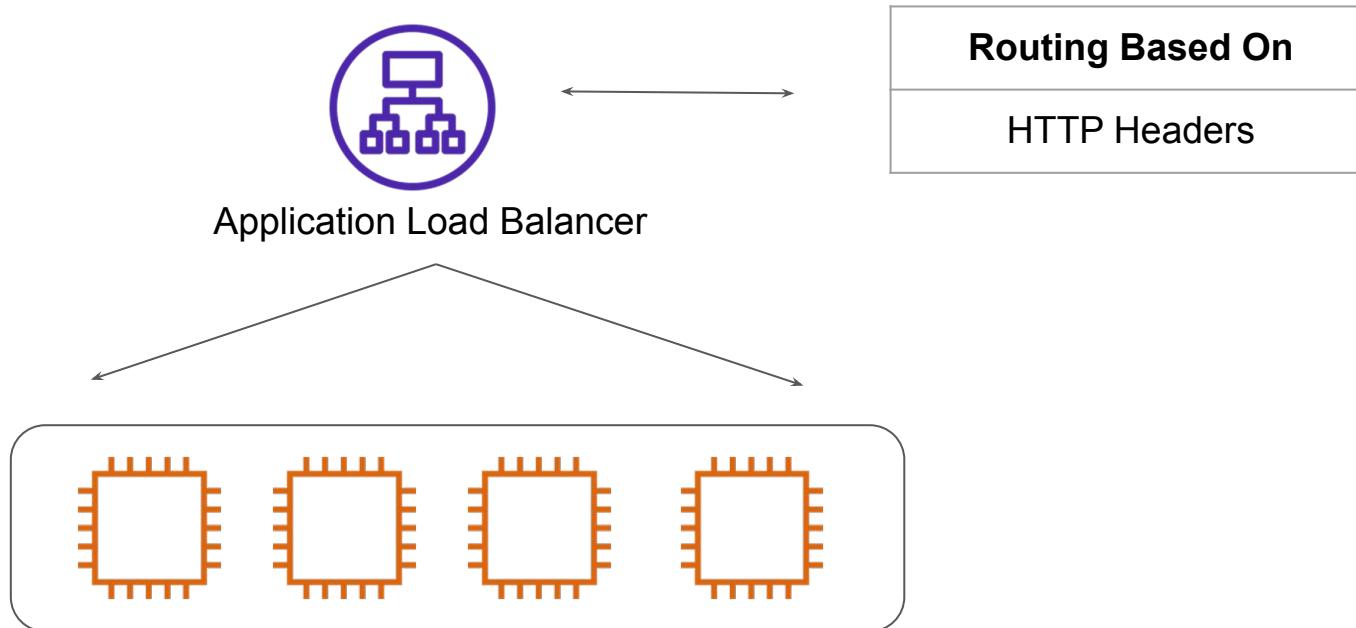
▼ Request Headers (380 B)

- ⓘ Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
- ⓘ Accept-Encoding: gzip, deflate
- ⓘ Accept-Language: en-US,en;q=0.5
- ⓘ Connection: keep-alive
- ⓘ Host: demo-alb-137613815.us-east-1.elb.amazonaws.com
- ⓘ Upgrade-Insecure-Requests: 1

ⓘ User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:102.0) Gecko/20100101 Firefox/102.0

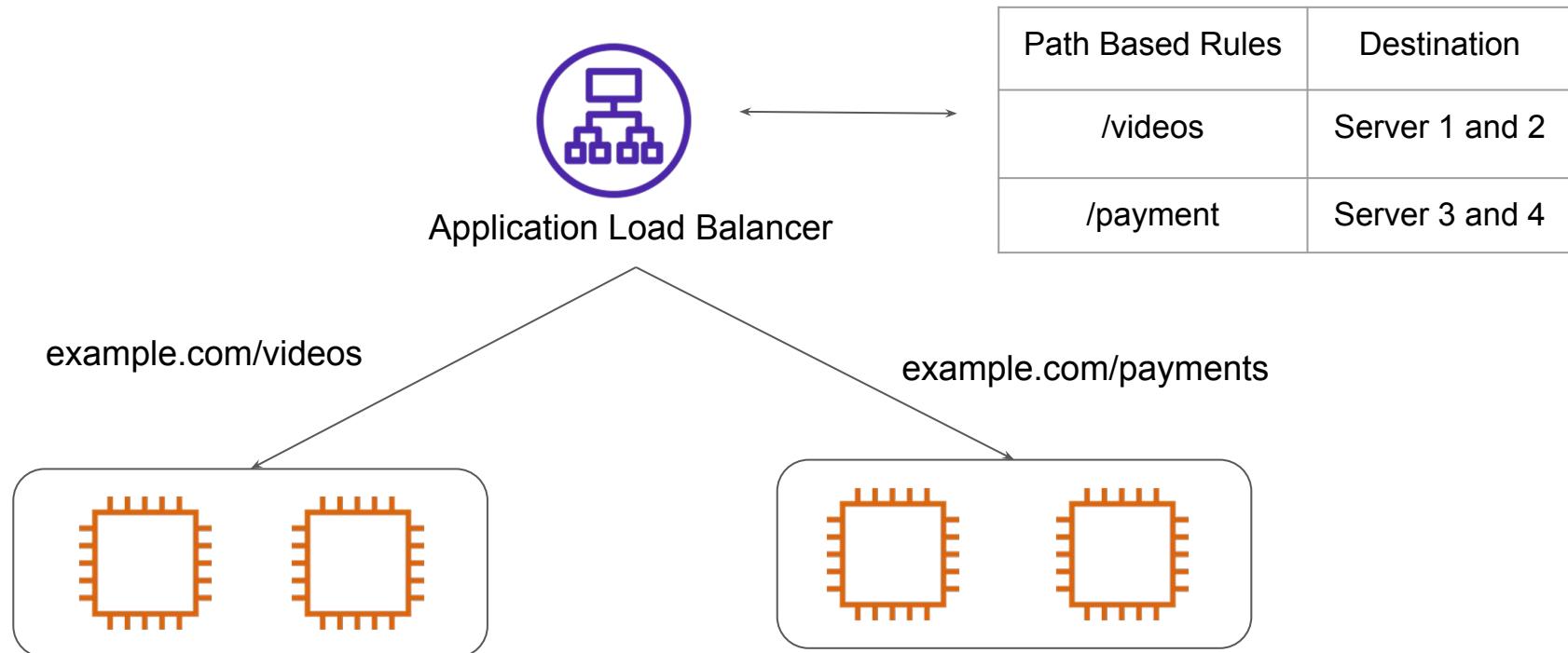
Understanding ALB

Application Load Balancer functions at Application layer and support both HTTP & HTTPS



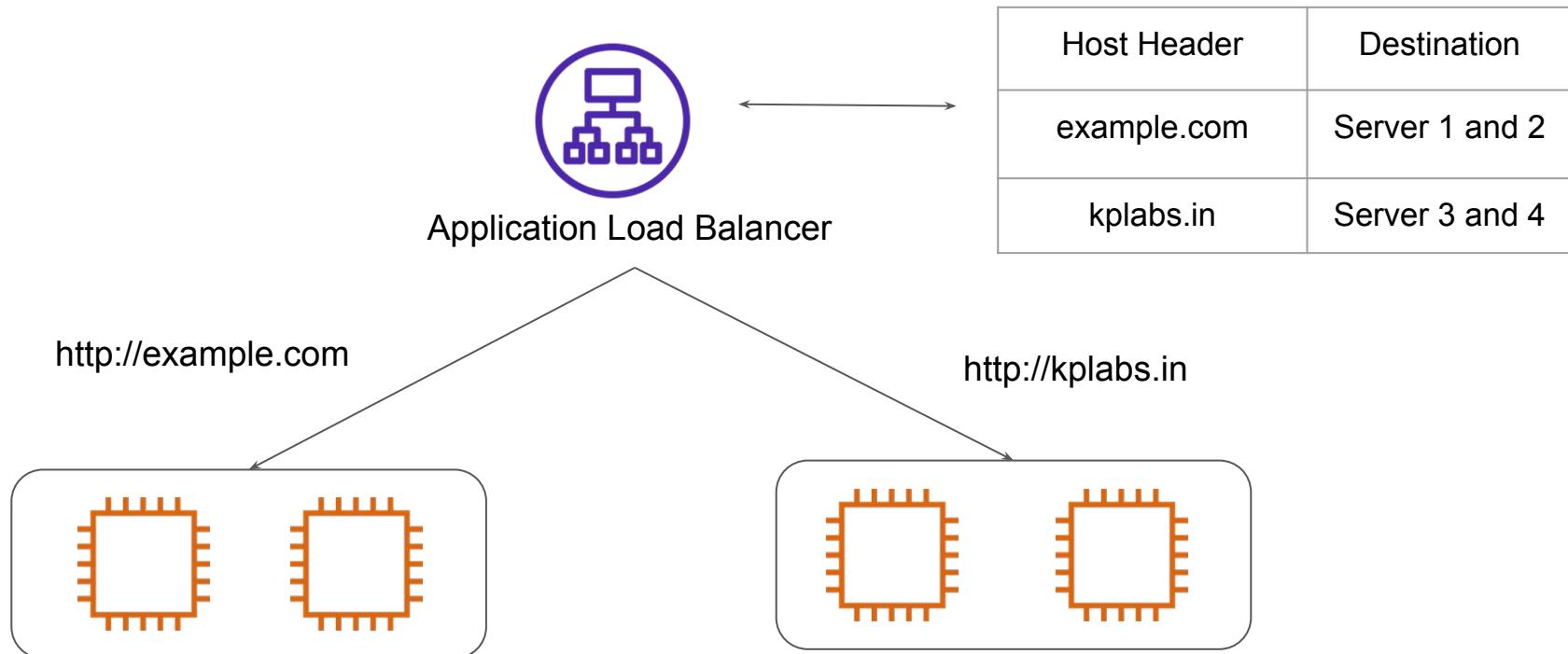
Path Based Routing

The requests are routed based on the URI path.



Routing Using Host Headers

The requests are routed based on the Host Header

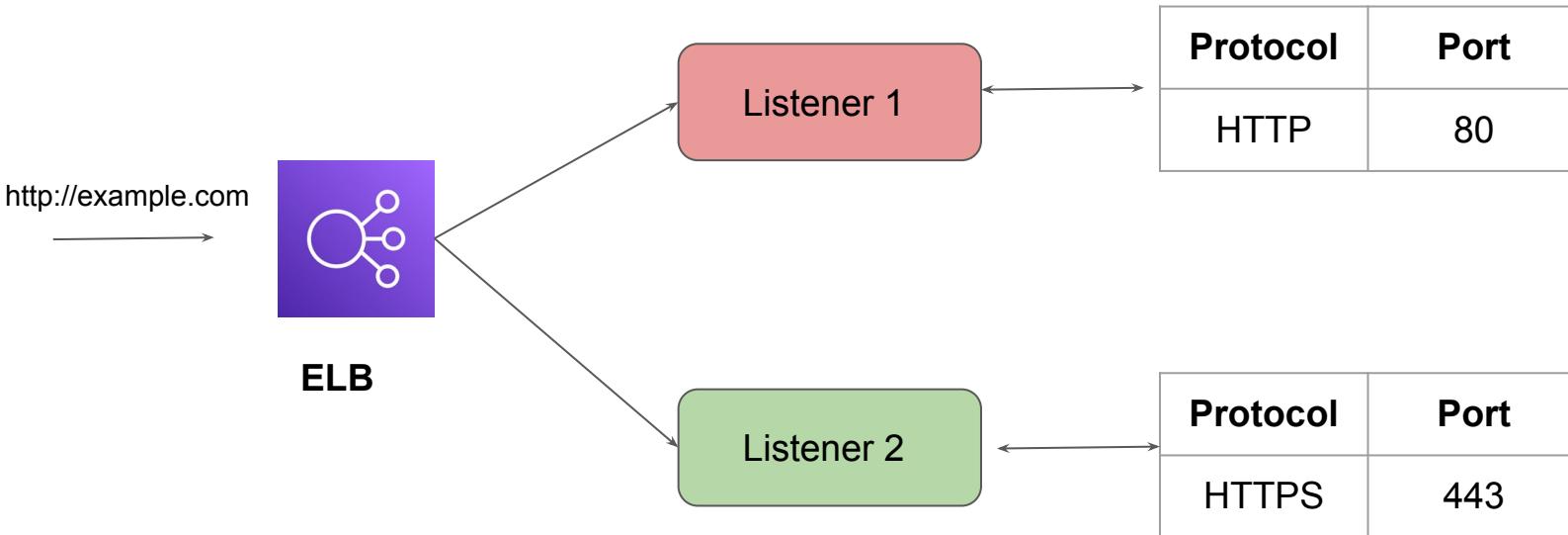


Listener & Target Groups

Next generation load balancers

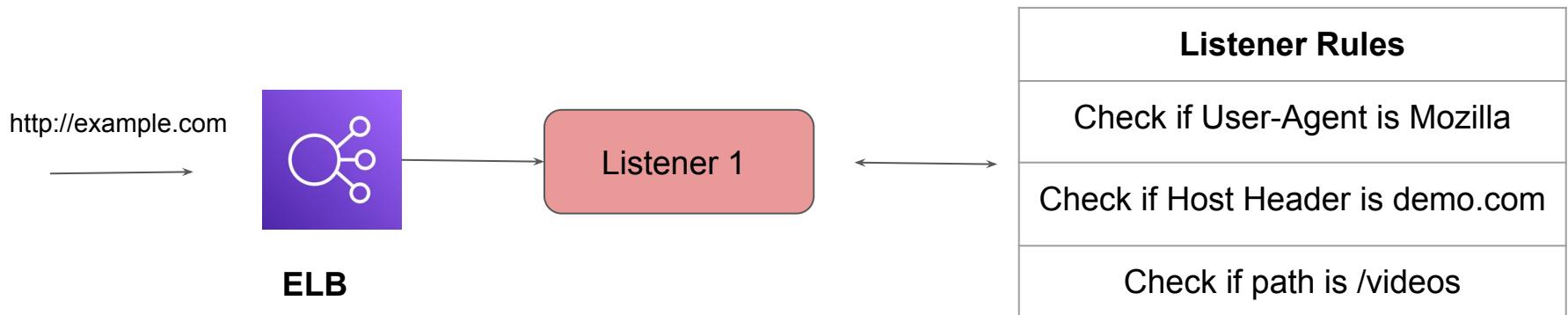
Understanding Listeners

A **listener** is a process that checks for connection requests, using the protocol and port that you configure.



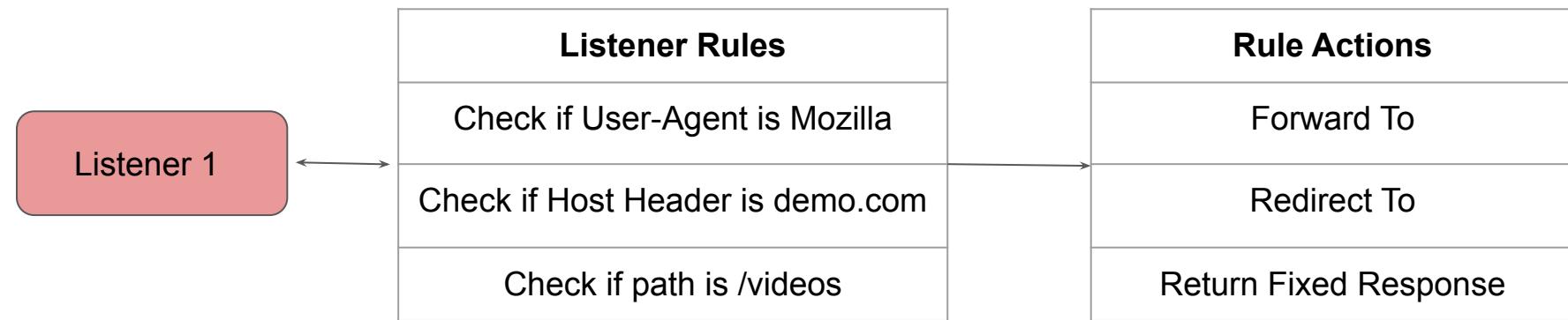
Listener Rules

Each listener has a rule based on which an action is taken based on a request.



Listener Rule Actions

If a request matches a specific rule, what action you want to perform on that request is determined in the Rule Actions.



demo-alb | HTTP:80 (4 rules)

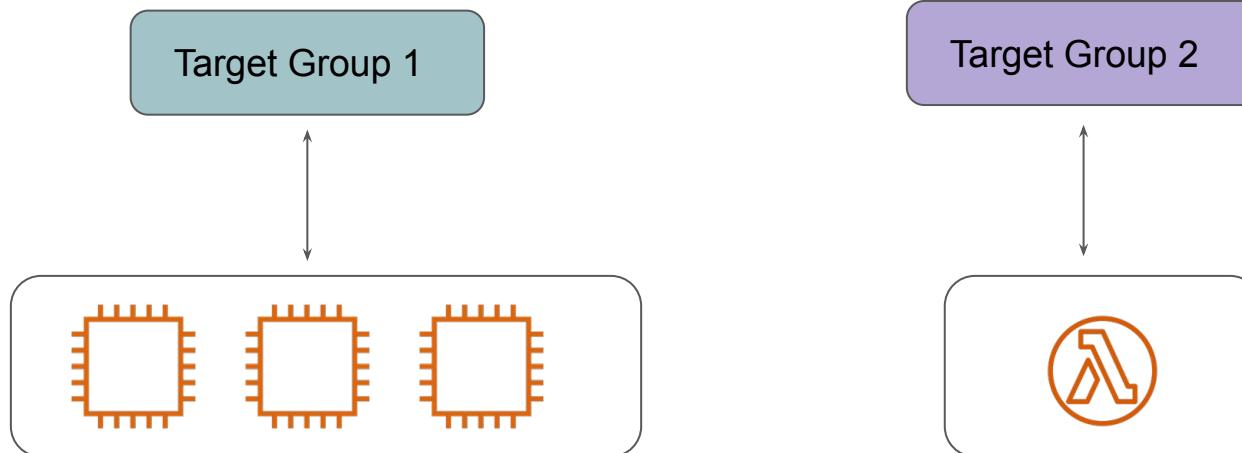
- ▶ Rule limits for condition values, wildcards, and total rules.

1 arn...93720 ▾	IF ✓ Http header User-Agent is *curl*	THEN Return fixed response 200 Content-Type: text/plain Response body: Hi curl! (less...)
2 arn...c9bc6 ▾	IF ✓ Http header User-Agent is *Mozilla*	THEN Return fixed response 200 Content-Type: text/plain Response body: Hey Mozilla! You have great addons! (less...)
3 arn...fdb85 ▾	IF ✓ Http header User-Agent is *wget*	THEN Return fixed response 200 Content-Type: text/plain Response body: Hi There wget! I detected you. (less...)

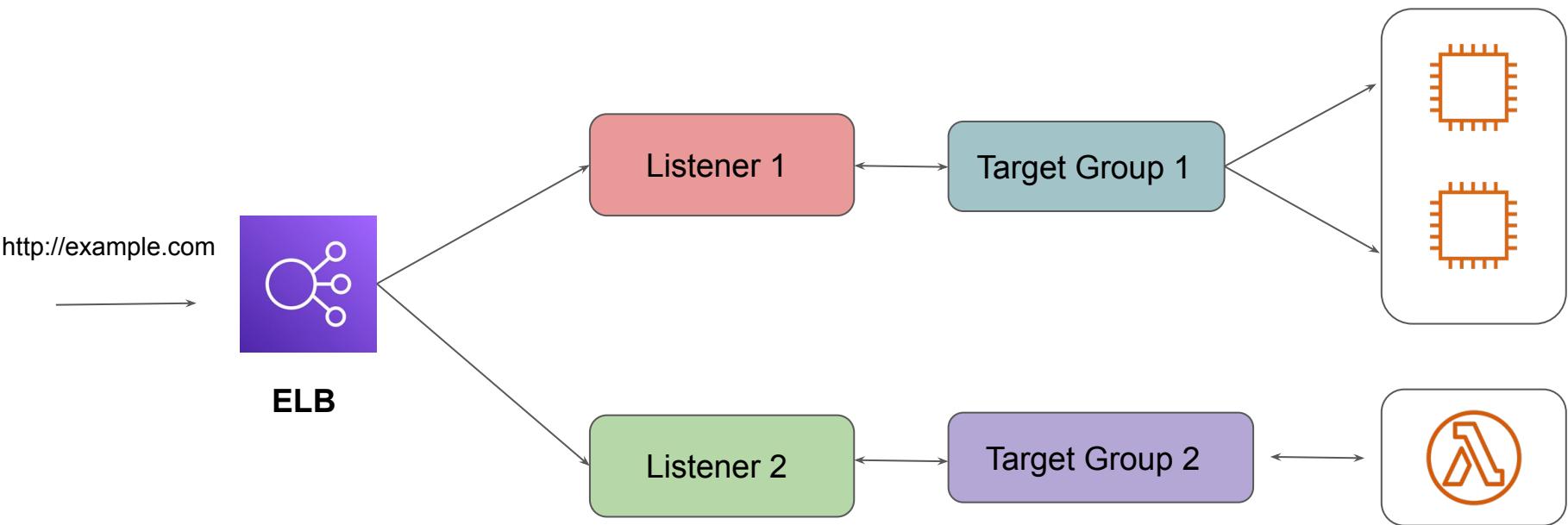
Understanding Target Groups

Target group is used to route requests to one or more registered targets.

These targets can be EC2 instances, Lambda Functions, and others.



Overall Workflow



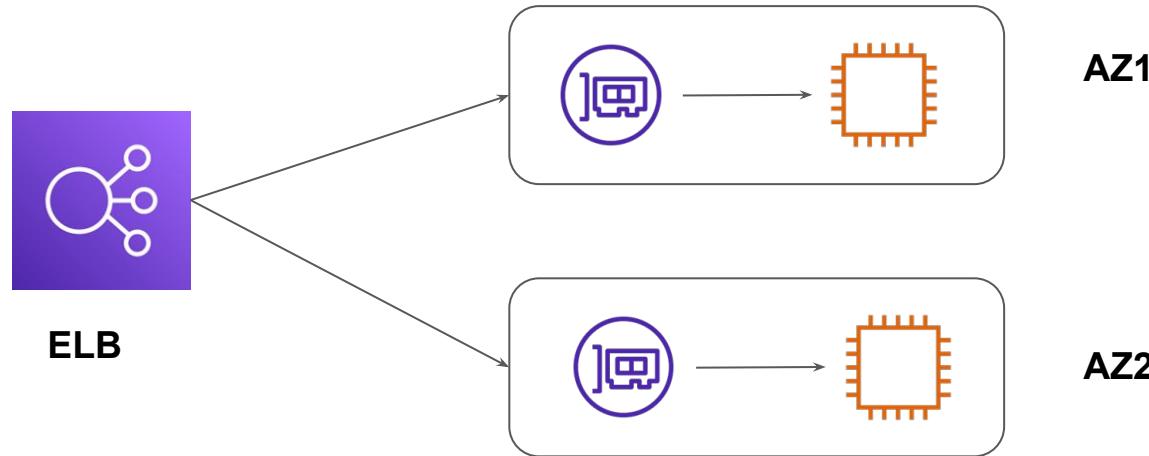
Availability Zones and ELB nodes

ELB Interfaces

Availability Zones and ELB nodes

When you enable an Availability Zone for your load balancer, Elastic Load Balancing creates a load balancer node in the Availability Zone.

If you register targets in an Availability Zone but do not enable the Availability Zone, these registered targets do not receive traffic.



Recommendations

With an Application Load Balancer, it is a requirement that you enable at least two or more Availability Zones. If one Availability Zone becomes unavailable or has no healthy targets, the load balancer can route traffic to the healthy targets in another Availability Zone.

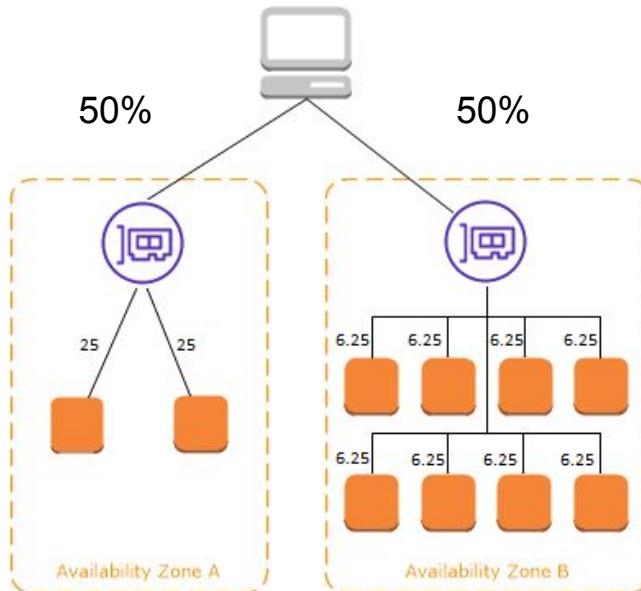
After you disable an Availability Zone, the targets in that Availability Zone remain registered with the load balancer. However, even though they remain registered, the load balancer does not route traffic to them.

Cross Zone Load Balancing

Interesting Learning

Understanding the Challenge

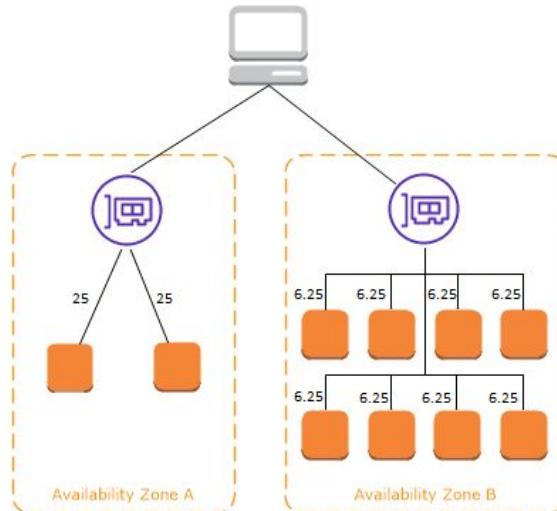
If Cross Zone Load Balancing is disabled, each load balancer node distributes traffic only across the registered targets in its Availability Zone.



Cross Zone Load Balancing Disabled

Each of the two targets in Availability Zone A receives 25% of the traffic.

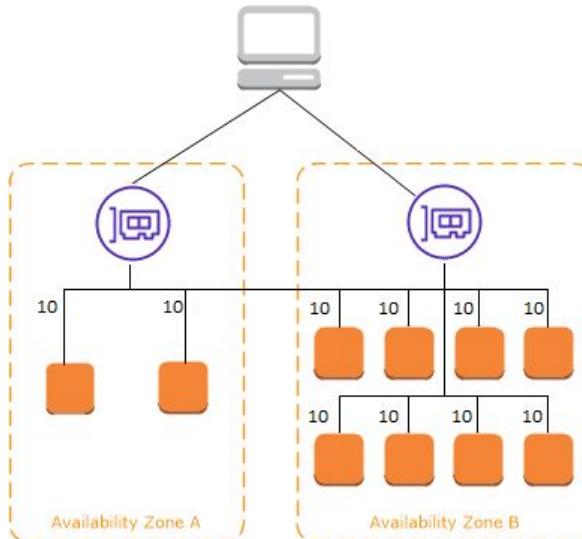
Each of the eight targets in Availability Zone B receives 6.25% of the traffic.



Cross Zone Load Balancing

When cross-zone load balancing is enabled, each load balancer node distributes traffic across the registered targets in all enabled Availability Zones.

If cross-zone load balancing is enabled, each of the 10 targets receives 10% of the traffic.



Important Pointers

With Application Load Balancers, cross-zone load balancing is always enabled.

With Network Load Balancers and Gateway Load Balancers, cross-zone load balancing is disabled by default. After you create the load balancer, you can enable or disable cross-zone load balancing at any time.

ELB Access Logs

Who is Visiting Us?

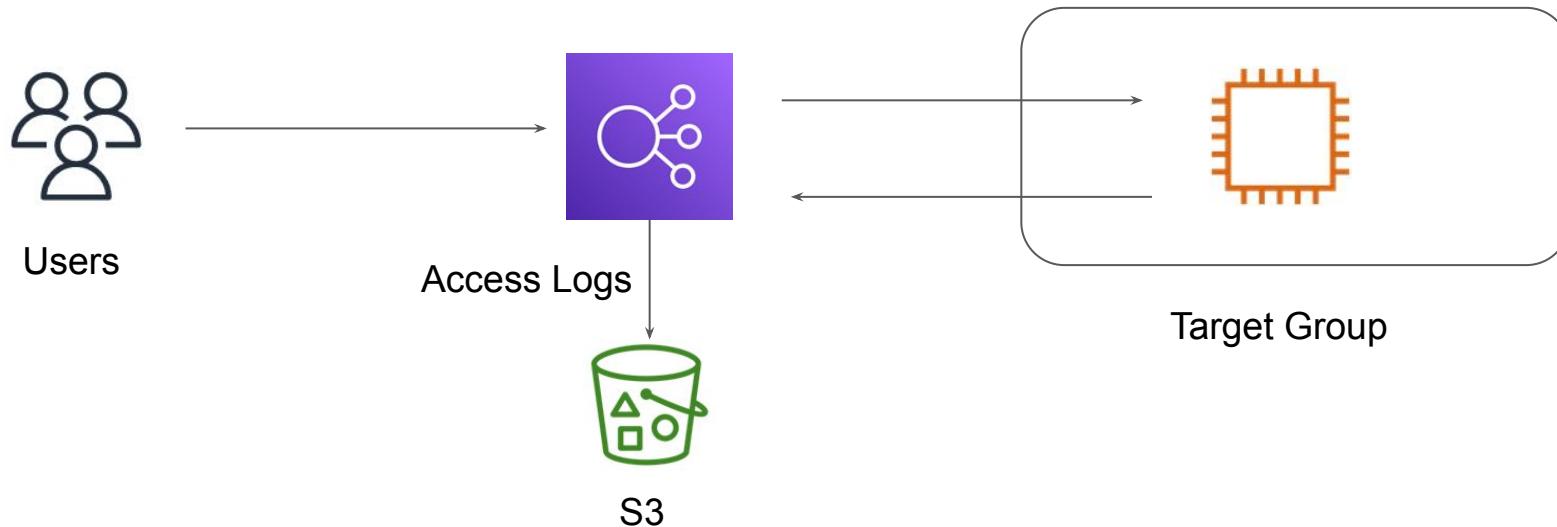
Overview of Access Logs

An access log is a list of all the requests for individual files that people have requested from a Web site

```
[root@ip-172-26-7-135 nginx]# tail -f access.log
128.14.133.58 - - [03/Sep/2021:04:43:10 +0000] "GET / HTTP/1.1" 200 82 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36" "-"
104.149.165.66 - - [03/Sep/2021:04:45:03 +0000] "HEAD /robots.txt HTTP/1.0" 404 0 "-" "-" "-"
92.118.160.57 - - [03/Sep/2021:05:02:42 +0000] "GET / HTTP/1.0" 200 82 "-" "NetSystemsResearch studies the availability of various services across the internet. Our website is netsystemsresearch.com" "-"
114.119.154.115 - - [03/Sep/2021:05:05:14 +0000] "GET /topic/blockchain/ HTTP/1.1" 404 153 "-" "Mozilla/5.0 (Linux; Android 7.0;) AppleWebKit/537.36 (KHTML, like Gecko) Mobile Safari/537.36 (compatible; PetalBot;+https://webmaster.petalsearch.com/site/petalbot)" "-"
135.125.244.48 - - [03/Sep/2021:05:11:08 +0000] "POST / HTTP/1.1" 405 559 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.129 Safari/537.36" "-"
135.125.244.48 - - [03/Sep/2021:05:11:08 +0000] "GET /.env HTTP/1.1" 404 555 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.129 Safari/537.36" "-"
109.49.235.11 - - [03/Sep/2021:05:12:32 +0000] "GET / HTTP/1.1" 200 82 "-" "Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36" "-"
185.53.90.24 - - [03/Sep/2021:05:20:55 +0000] "GET http://icanhazip.com/ HTTP/1.1" 200 82 "-" "Go-http-client/1.1" "-"
114.119.154.11 - - [03/Sep/2021:05:28:51 +0000] "GET /topic/graphic-design/ HTTP/1.1" 404 153 "-" "Mozilla/5.0 (Linux; Android 7.0;) AppleWebKit/537.36 (KHTML, like Gecko) Mobile Safari/537.36 (compatible; PetalBot;+https://webmaster.petalsearch.com/site/petalbot)" "-"
199.168.150.161 - - [03/Sep/2021:05:39:07 +0000] "GET / HTTP/1.1" 302 145 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/61.0.3163.100 Safari/537.36" "-"
```

ELB Access Logs

Elastic Load Balancing provides access logs that capture detailed information about requests sent to your load balancer.



Important Pointers for Access Logs - Part 1

Access logging is an optional feature of Elastic Load Balancing that is disabled by default

Elastic Load Balancing logs requests on a best-effort basis. AWS recommend that you use access logs to understand the nature of the requests, not as a complete accounting of all requests.

Important Pointers for Access Logs - Part 2

The bucket and your load balancer must be in the same Region.

Bucket Policy should be designed so that AWS Account must be able to write to your bucket.

Elastic Load Balancing publishes a log file for each load balancer node every 5 minutes.

Relax and Have a Meme Before Proceeding



alcohol
@Mandac5

What is an extreme sport?



allison
@amazaleax

Doing your homework while the
teacher is collecting it

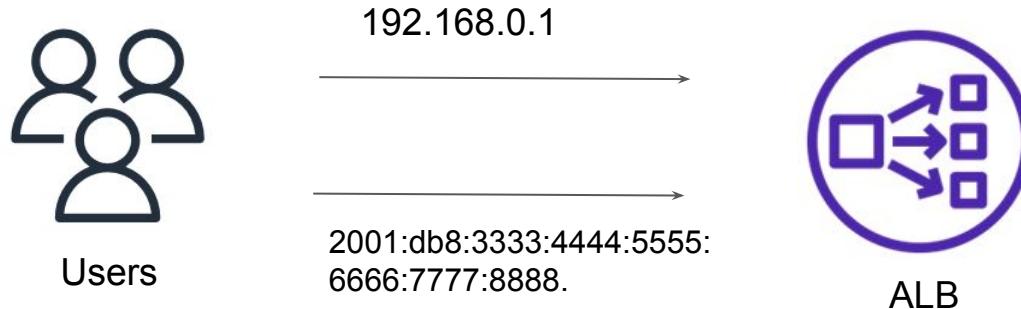
Dualstack IP Address Type for ELBs

Enable IPv6 for ELBs

IP Address Type Support

ELB Supports two address types:

- i) IPv4
- ii) Dualstack (includes both IPv4 and IPv6 addresses)



Important Pointer

To use IPv6 addresses, the virtual private cloud (VPC) where you launch your ELB must have subnets with associated IPv6 CIDR blocks

IPv6 addresses can be associated only with internet-facing Application Load Balancers and Network Load Balancers.

Internal Application Load Balancers, Classic Load Balancers, and Network Load Balancers do not support IPv6 addresses.

Launch Templates

Launching EC2 The Easy Way

Understanding the Challenge

When you launch an EC2 instance, there are various configurations that needs to be set.

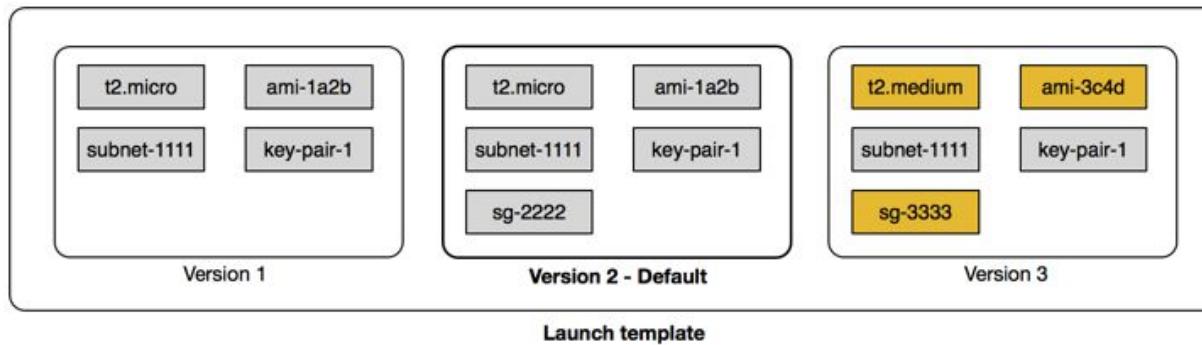
Some of the common configuration includes:

- AMI ID
- Instance Type
- Security Group
- Key Pair
- Storage
- IAM Role
- VPC

Everytime when you intend to launch instance, going through process is time consuming,

Introduction to Launch Templates

Launch templates enable you to store launch parameters so that you do not have to specify them every time you launch an instance.

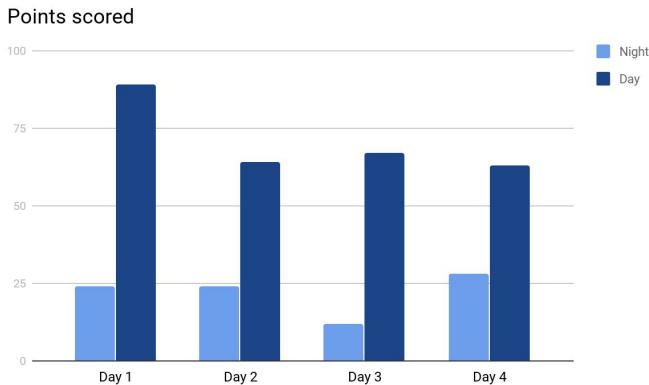


EC2 Auto Scaling

Up and Down, Round and Round

Understanding Scalability

- Scalability is the ability of a system to change in size depending on the needs.
- Infrastructure should scale to support changing in traffic patterns.



Launch and Remove Servers Based on Load

What if new servers automatically get launched on high load?

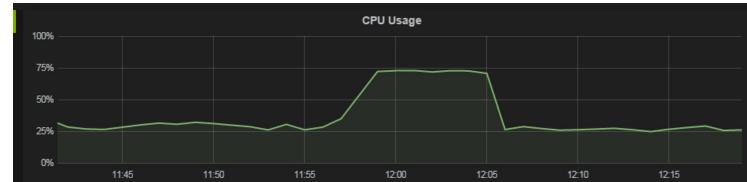
Simple Scaling Policy:

Base: 2 servers

Scalable :

If average CPU utilization > 60% ; add two more instance

If average CPU utilization < 30% ; remove two instance

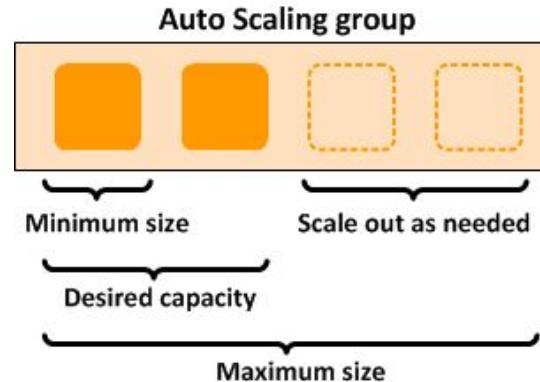


Overview of EC2 Auto-Scaling

Amazon EC2 Auto Scaling helps you maintain application availability and allows you to automatically add or remove EC2 instances according to conditions you define.

Example Scenario:

- Minimum: 2 EC2 instance
- Maximum: 10 EC2 instance
- Threshold: 50% of CPU



Multiple Types of Scaling

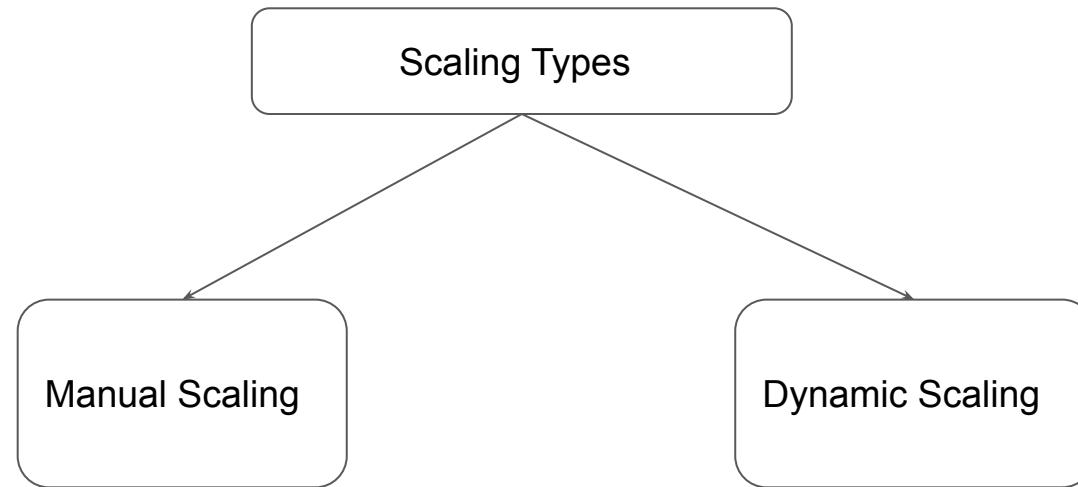
Type of Scaling	Description
Scheduled Scaling	<p>Servers are scaled based on a specific schedule.</p> <p>For example, every week the traffic to your web application starts to increase on Wednesday, remains high on Thursday, and starts to decrease on Friday</p>
Dynamic Scaling	<p>Follow the demand curve for scaling activities.</p> <p>CPU Utilization higher than 90%</p>
Predictive Scaling	<p>Predictive Scaling has machine learning algorithms that detect changes in daily and weekly patterns, automatically adjusting their forecasts.</p>

Dynamic Scaling

Overview

2 Types of Scaling

There are two primary types of scaling approaches that are available:



Dynamic Scaling

When you configure dynamic scaling, you define how to scale the capacity of your Auto Scaling group in response to changing demand.

Scaling Policy Types	Descriptions
Target tracking scaling	Increase or decrease the current capacity of the group based on a target value for a specific metric.
Step scaling	Increase or decrease the current capacity of the group based on a set of scaling adjustments, known as step adjustments, that vary based on the size of the alarm breach.
Simple scaling	Increase or decrease the current capacity of the group based on a single scaling adjustment.

Simple Scaling Policy

With simple scaling policy, you can configure a specific number of instances to be added when a threshold reaches certain value.

Policy type
Simple scaling

Scaling policy name
simple-scaling

CloudWatch alarm
Choose an alarm that can scale capacity whenever:
higher-60 [Create a CloudWatch alarm](#) 
breaches the alarm threshold: CPUUtilization >= 60 for 1 consecutive periods of 300 seconds for the metric dimensions:
AutoScalingGroupName = asg-manual-scaling

Take the action
Add 3 capacity units

And then wait
300 seconds before allowing another scaling activity

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

Step Scaling Policy

In step scaling, the adjustment of the current capacity of instances vary based on the size of the alarm breach.

Policy type
Step scaling

Scaling policy name
step-up

CloudWatch alarm
Choose an alarm that can scale capacity whenever:
higher-60 [Create a CloudWatch alarm](#) [C](#)
breaches the alarm threshold: CPUUtilization ≥ 60 for 1 consecutive periods of 300 seconds for the metric dimensions:
AutoScalingGroupName = asg-manual-scaling

Take the action
Add [▼](#)

1 capacity units [▼](#) when 60 \leq CPUUtilization < 70

3 capacity units [▼](#) when 70 \leq CPUUtilization < +infinity [X](#)

[Add step](#)

Target Tracking Policy

With target tracking scaling policies, you select a scaling metric and set a target value.

The scaling policy adds or removes capacity as required to keep the metric at, or close to, the specified target value.

Policy type
Target tracking scaling

Scaling policy name
Target Tracking Policy

Metric type
Average CPU utilization

Target value
50

Instances need
300 seconds warm up before including in metric

Disable scale in to create only a scale-out policy

Create

Use Case of Thermostat

A thermostat is a component which senses the temperature of a physical system and performs actions so that the system's temperature is maintained near a desired setpoint.

Example:

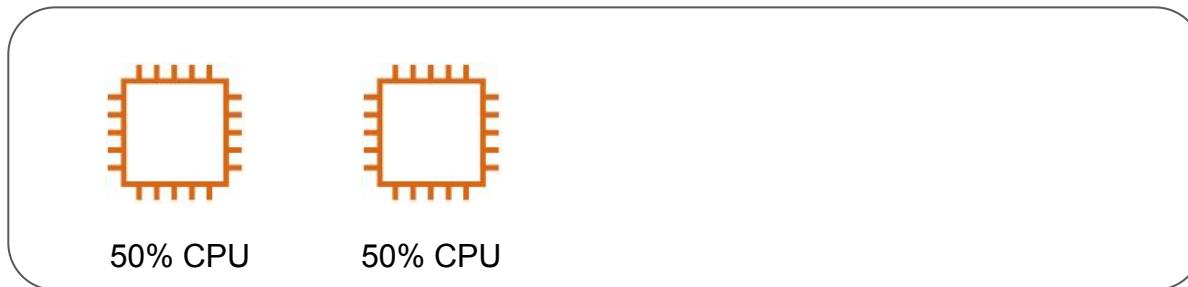
- Desired = 24
- Current = 18



Example Target Tracking Policy

Metric Type = CPU Utilization

Target Value = 50%

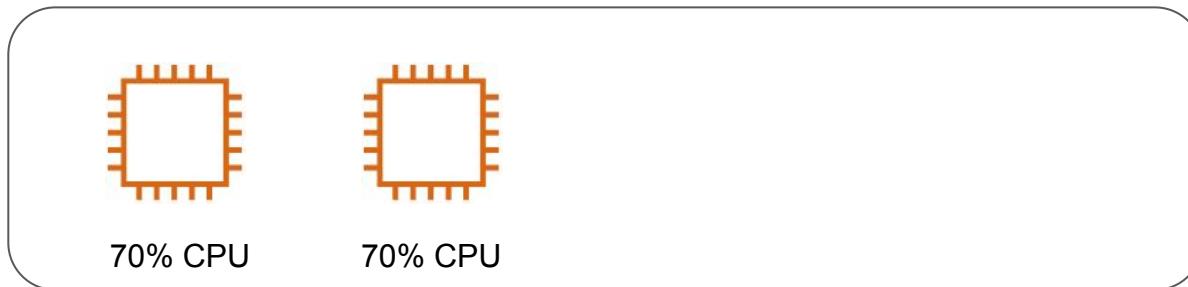


Example Target Tracking Policy

Metric Type = CPU Utilization

Target Value = 50%

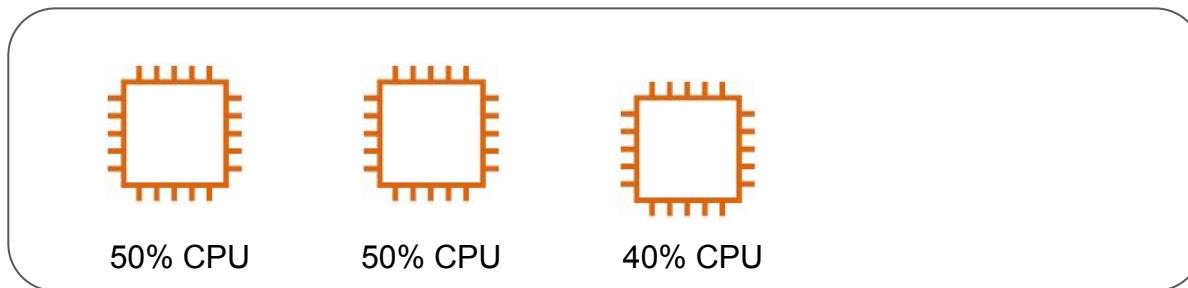
Actual Value = 70%



Example Target Tracking Policy

Metric Type = CPU Utilization

Target Value = 50%

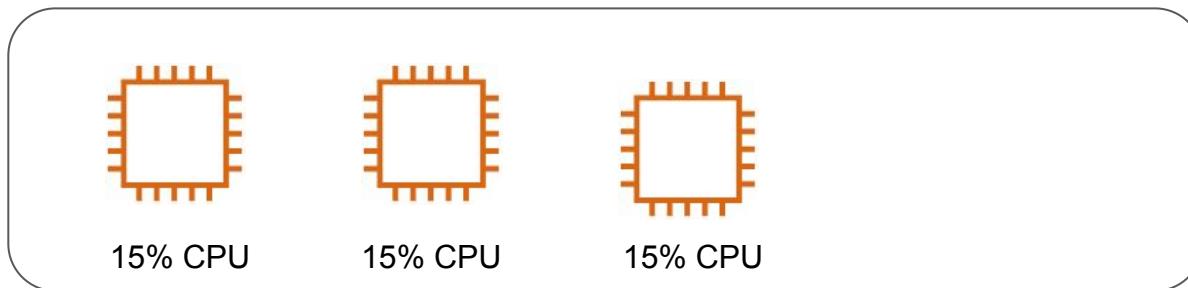


Example Target Tracking Policy

Metric Type = CPU Utilization

Target Value = 50%

Actual Value = 15%

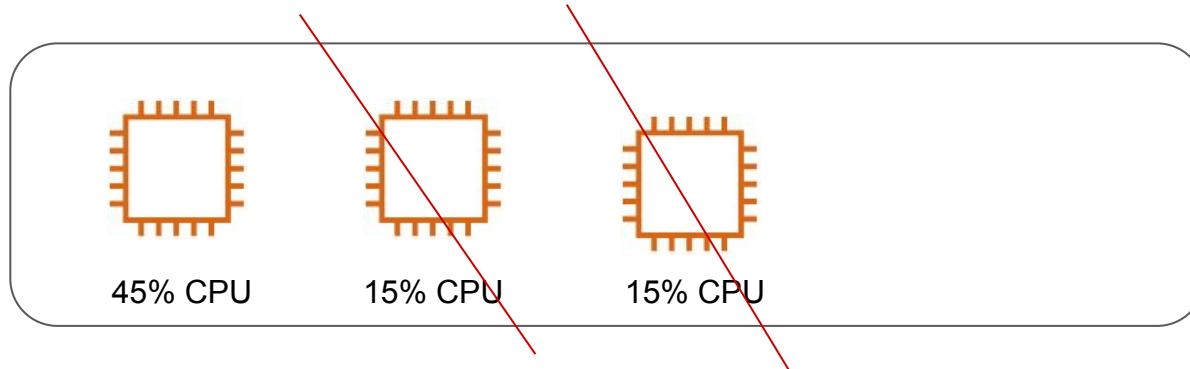


Example Target Tracking Policy

Metric Type = CPU Utilization

Target Value = 50%

Actual Value = 45% (average)



Scheduled Scaling

Overview and Practical

Overview of Scheduled Scaling

Scheduled scaling allows you to set your own scaling schedule.

For example, let's say that every week the traffic to your web application starts to increase on Wednesday, remains high on Thursday, and starts to decrease on Friday.

Scaling actions are performed automatically as a function of time and date.

Relax and Have a Meme Before Proceeding

That stupid walk you do when someone's mopping a floor and you know you're gonna walk over it but you want them to see how sorry you are to be walking over it so you make yourself look like you're walking over hot lava.



Auto-Scaling LifeCycle Hooks

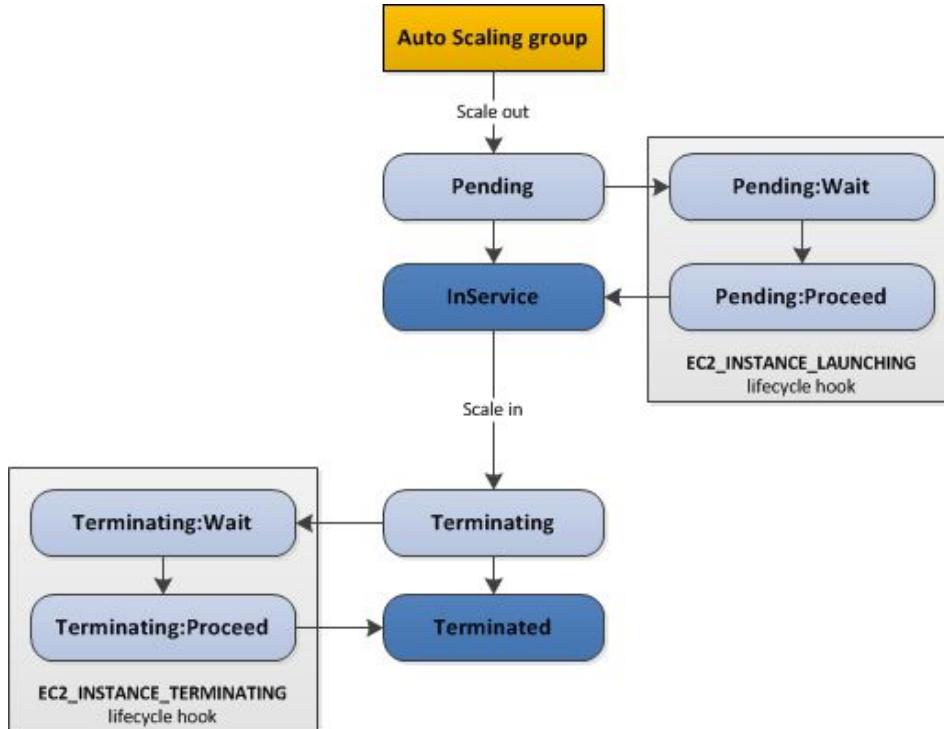
Challenges and Structure

Overview of LifeCycle Hooks

Auto-Scaling Lifecycle hooks allows us to have control over instance launch and termination state within auto-scaling group.

Sample Use-Case:

- You have EC2 instance which is scheduled to be terminated.
- You want to backup all it's logs to S3 and run some deregistration scripts.
- Terminate instance once 2nd steps is completed.



Step 1: Instance Awaiting Termination

Auto Scaling Group: sample-asg-group

Activity History

Scaling Policies Instances Monitoring Notifications Tags Scheduled Actions Lifecycle Hooks

Filter: Any Status ▾ Filter scaling history... 1 to 3 of 3 History Items

Status	Description	Start Time	End Time
Waiting for Terminate Lifecycle Action	Terminating EC2 instance: i-05bc139026835d753	2019 January 24 19:18:30 UTC+5:30	
Successful	Launching a new EC2 instance: i-0c931af7dde9441fb	2019 January 24 19:06:55 UTC+5:30	2019 January 24 19:14:05 UTC+5:30
Successful	Launching a new EC2 instance: i-05bc139026835d753	2019 January 24 18:54:17 UTC+5:30	2019 January 24 18:54:50 UTC+5:30

Step 2: Confirmation from Automation

```
[root@ip-172-31-34-10 ~]# aws autoscaling complete-lifecycle-action --lifecycle-hook-name SampleTerminateHook --auto-scaling-group-name sample-asg-group --lifecycle-action-result CONTINUE --instance-id i-05bc139026835d753 --region us-east-1
[root@ip-172-31-34-10 ~]#
Broadcast message from root@ip-172-31-34-10
(unknown) at 14:36 ...

```

The system is going down for power off NOW!
Connection to 54.160.201.201 closed by remote host.
Connection to 54.160.201.201 closed.

Step 3: Go Ahead!

Auto Scaling Group: sample-asg-group

Details Activity History Scaling Policies Instances Monitoring Notifications Tags Scheduled Actions Lifecycle Hooks

Filter: Any Status ▾ Filter scaling history... × 1 to 3 of 3 History Items

Status	Description	Start Time	End Time
Successful	Terminating EC2 instance: i-05bc139026835d753	2019 January 24 19:18:30 UTC+5:30	2019 January 24 20:08:21 UTC+5:30
Successful	Launching a new EC2 instance: i-0c931af7dde9441fb	2019 January 24 19:06:55 UTC+5:30	2019 January 24 19:14:05 UTC+5:30
Successful	Launching a new EC2 instance: i-05bc139026835d753	2019 January 24 18:54:17 UTC+5:30	2019 January 24 18:54:50 UTC+5:30

Basics of API



Understanding the Challenge

Book Distributor maintains the list of available books in it's backend systems.

Operator has access to Backend system to check the availability.

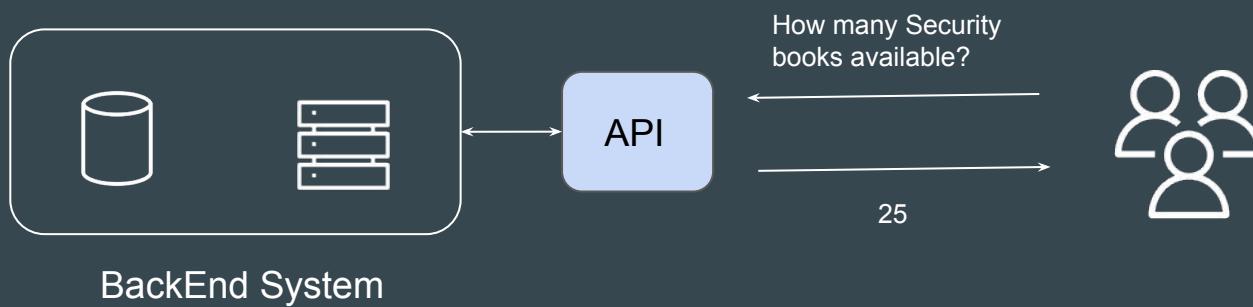
Clients they connect to Operator via Phone call / Chat option



API Based Approach

The book distributor could provide an API to check stock availability.

APIs let you open up access to your resources while maintaining security and control.



Simple Use-Case

James wants to build a weather report application.

OpenWeatherMap is an online service that provides global weather data via API.

He decided to connect his application to OpenWeatherMap API to fetch the latest reports and populate it in application.



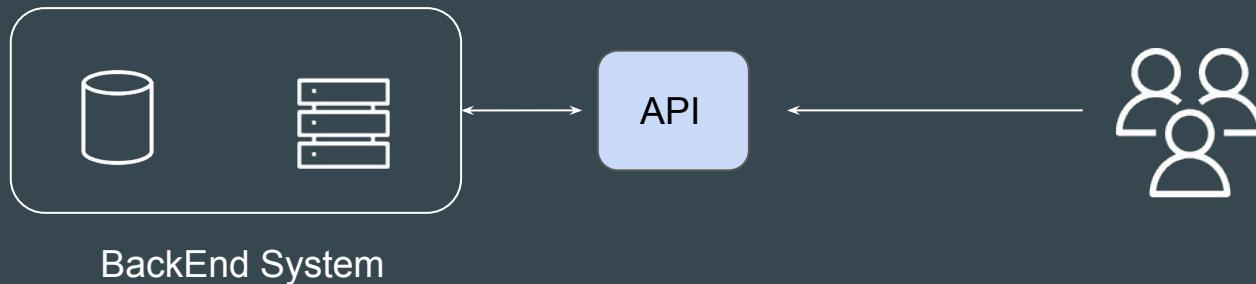
API Gateway



Introduction to Topic

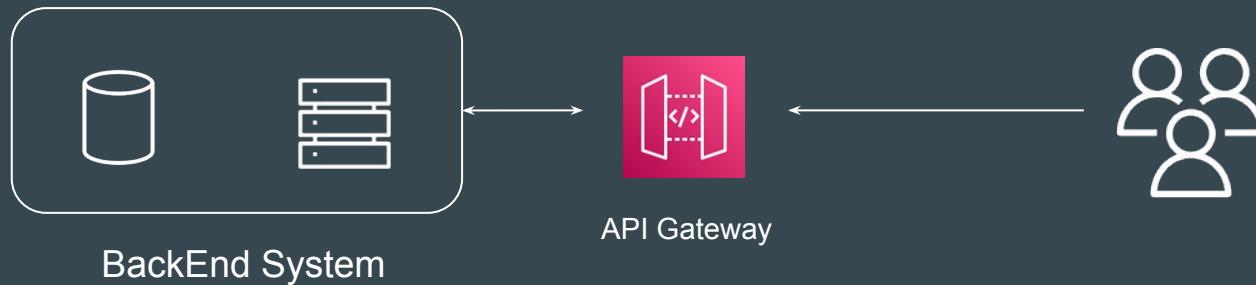
APIs act as the "**front door**" for applications to access data, business logic, or functionality from your backend services.

Hence API should be able to be highly available and handle thousands of requests.



Understanding the Basics

Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale.



REST APIs vs HTTP APIs



Understanding the Basics

REST APIs and HTTP APIs are both RESTful API products.

REST APIs support more features than HTTP APIs, while HTTP APIs are designed with minimal features so that they can be offered at a lower price.

Which to Choose?

Choose REST APIs if you need features such as API keys, per-client throttling, request validation, AWS WAF integration, or private API endpoints.

Choose HTTP APIs if you don't need the features included with REST APIs.

Core Differences - Security

API Gateway provides a number of ways to protect your API from certain threats, like malicious actors or spikes in traffic.

Security features	REST API	HTTP API
Mutual TLS authentication	✓	✓
Certificates for backend authentication	✓	
AWS WAF	✓	

Core Differences - API Management

Choose REST APIs if you need API management capabilities such as API keys and per-client rate limiting

Features	REST API	HTTP API
Custom domains	✓	✓
API keys	✓	
Per-client rate limiting	✓	
Per-client usage throttling	✓	

Core Differences - Monitoring

API Gateway supports several options to log API requests and monitor your APIs

Feature	REST API	HTTP API
Amazon CloudWatch metrics	✓	✓
Access logs to CloudWatch Logs	✓	✓
Access logs to Amazon Kinesis Data Firehose	✓	
Execution logs	✓	
AWS X-Ray tracing	✓	

Core Differences - Endpoint Type

The endpoint type refers to the endpoint that API Gateway creates for your API

Endpoint types	REST API	HTTP API
Edge-optimized	✓	
Regional	✓	✓
Private	✓	

Core Differences - Development

As you're developing your API Gateway API, you decide on a number of characteristics of your API.

These characteristics depend on the use case of your API.

Features	REST API	HTTP API
CORS configuration	✓	✓
Test invocations	✓	
Caching	✓	
User-controlled deployments	✓	✓
Automatic deployments		✓
Custom gateway responses	✓	
Canary release deployments	✓	
Request validation	✓	
Request parameter transformation	✓	✓
Request body transformation	✓	

**When someone deployed HTTP
API for prod environment**

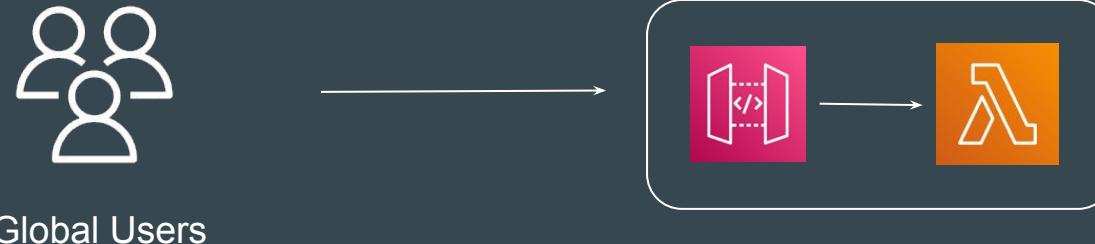


API Gateway Practical



Overall Implementation Architecture

1. Create HTTP API
2. API will invoke a backend Lambda function.

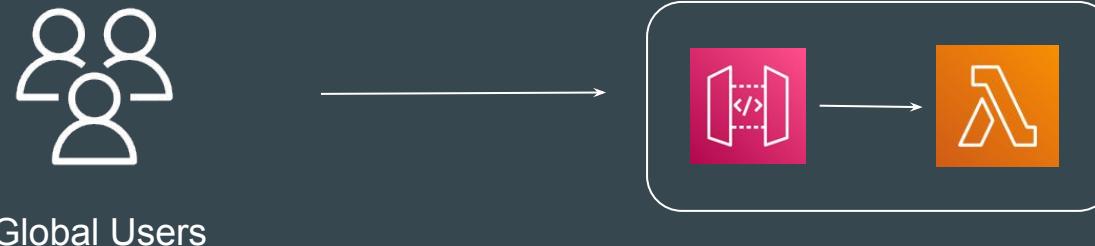


Creating REST API

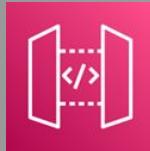


Overall Implementation Architecture

1. Create REST API
2. API will invoke a backend Lambda function.

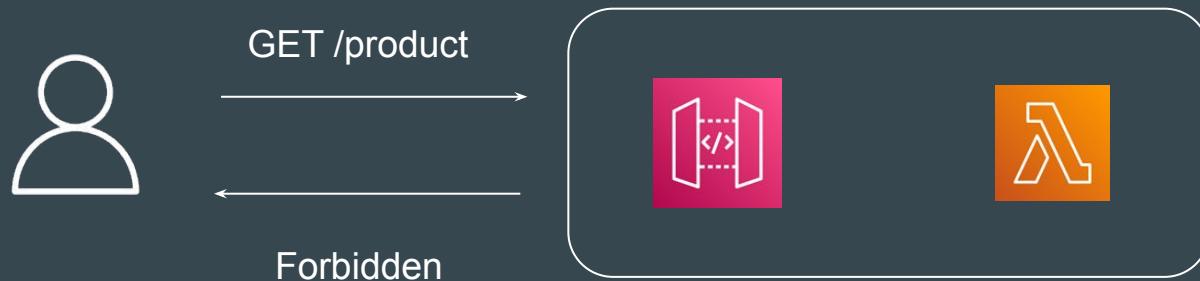


API Keys and Usage Plans



Basics of API Keys

API keys are alphanumeric string values that you distribute to application developer customers to grant access to your API.



Connecting Through API Key

You can use the **X-API-KEY** header while connecting to the API Endpoint.

```
C:\Users\zealv>curl --header "x-api-key: bDa2v0891F9TBgusPLptR253M4QzpVlrlTzKPPg3" https://9jxbr4wdac.execute-api.us-east-1.amazonaws.com/dev
{"statusCode":200,"body":"\"Hello from Lambda!\""}
```

Usage Plan

A **usage plan** specifies who can access one or more deployed API stages and methods—and optionally sets the target request rate to start throttling requests.

The plan uses API keys to identify API clients and who can access the associated API stages for each key.

The screenshot shows the 'demo-usage-plan' configuration page. The top navigation bar has tabs for 'Details', 'API Keys', and 'Marketplace'. The 'API Keys' tab is selected. Below the tabs, the usage plan details are listed:

- ID:** 8ad74n
- Name:** demo-usage-plan
- Description:** No description.
- Rate:** 10 requests per second
- Burst:** 20 requests
- Quota:** 1,000 requests per month starting on the 1st day

Below these details is a section titled 'Associated API Stages' with a 'Add API Stage' button. A table lists the associated API stage:

API	Stage	Method Throttling	Actions
demo-api	dev	No Methods Configured	Configure Method Throttling

Points to Note

After you create, test, and deploy your APIs, you can use API Gateway usage plans to make them available as product offerings for your customers.

You can configure usage plans and API keys to allow customers to access selected APIs, and **begin throttling requests** to those APIs based on defined limits and quotas.

These can be set at the API, or API method level.

Points to Note

API Gateway throttles requests to your API using the token bucket algorithm, where a token counts for a request

When request submissions exceed the steady-state request rate and burst limits, API Gateway begins to throttle requests. Clients may receive **429 Too Many Requests**

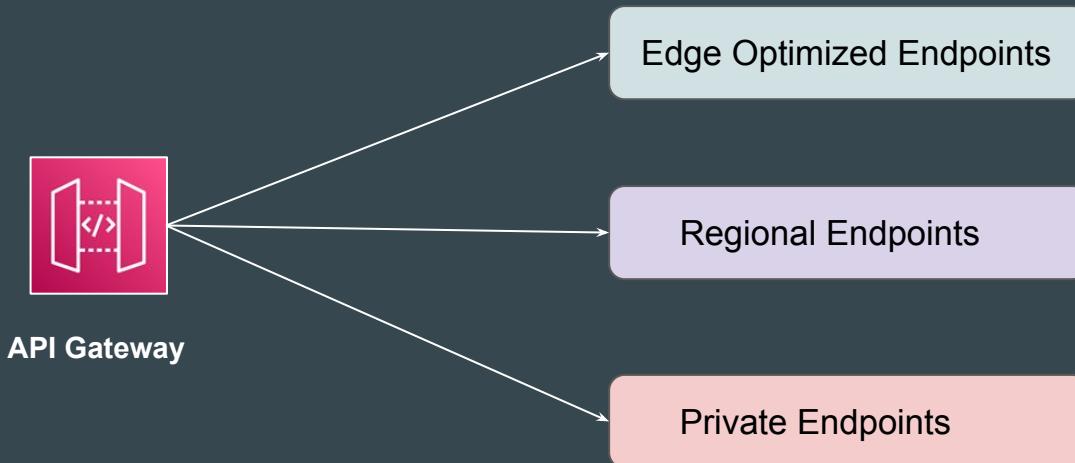
There is a default quota of 10,000 requests per second (RPS) applicable at per account per region.

API Gateway Endpoint Types



API Endpoints

Depending on where the majority of your API traffic originates from, you can create an appropriate API Gateway endpoint type.

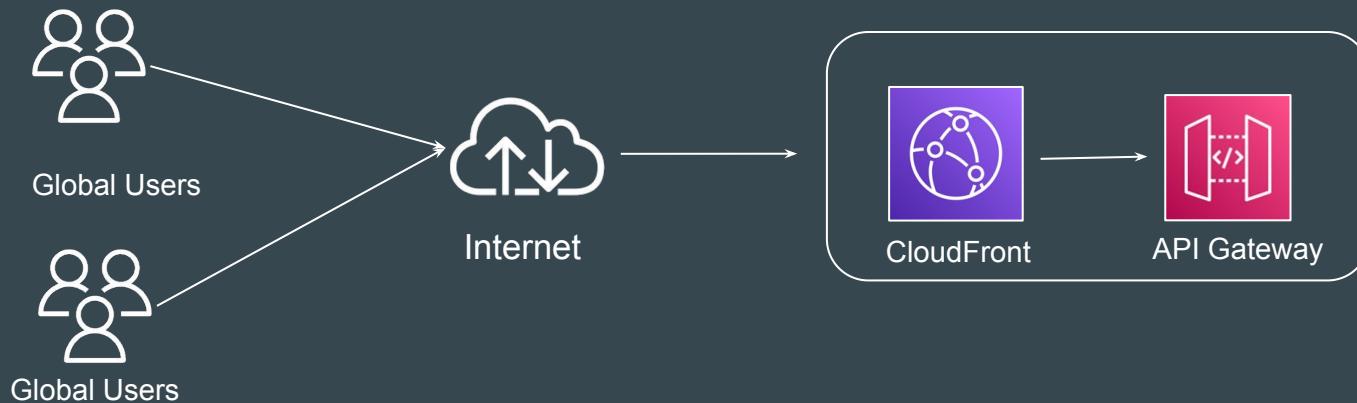


Edge-optimized API endpoints

An edge-optimized API endpoint is best for geographically distributed clients.

API requests are routed to the nearest CloudFront Point of Presence (POP).

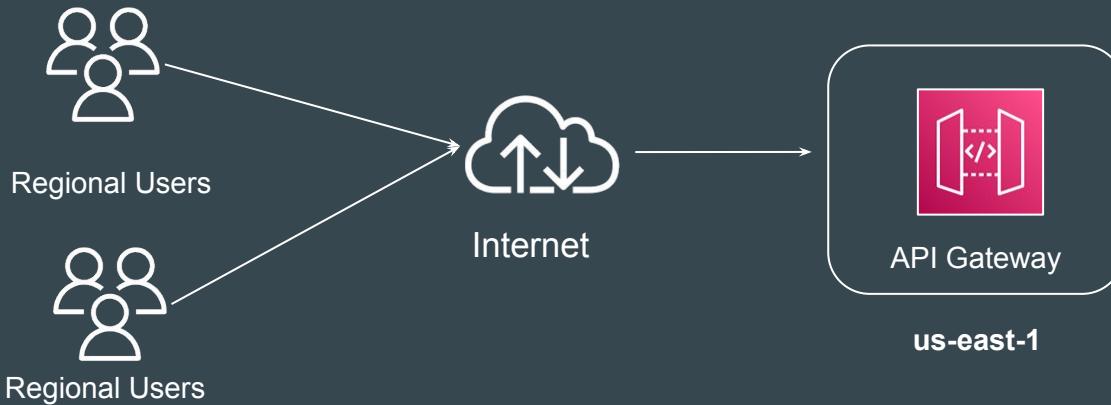
This is the default endpoint type for API Gateway REST APIs.



Regional API endpoints

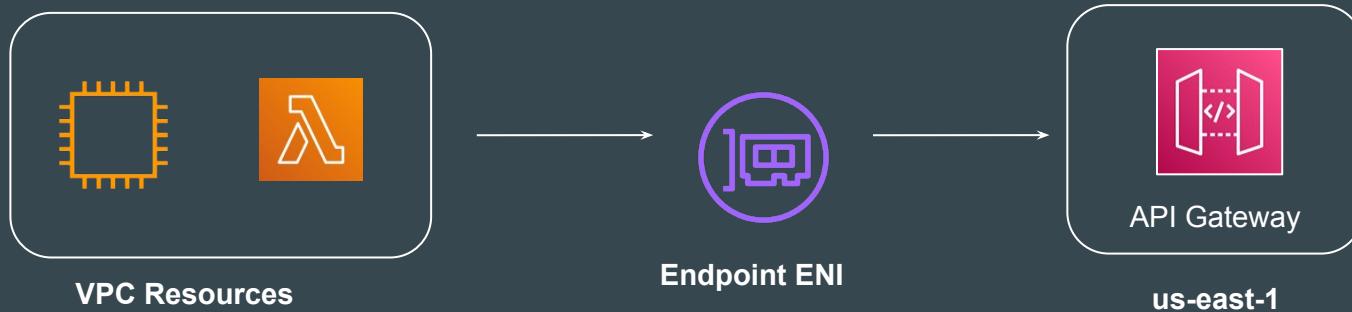
A regional API endpoint is intended for clients in the same region.

When a client running on an EC2 instance calls an API in the same region, or when an API is intended to serve a small number of clients with high demands, a regional API reduces connection overhead.



Private API endpoints

A private API endpoint is an API endpoint that can only be accessed from your Amazon Virtual Private Cloud (VPC) using an interface VPC endpoint

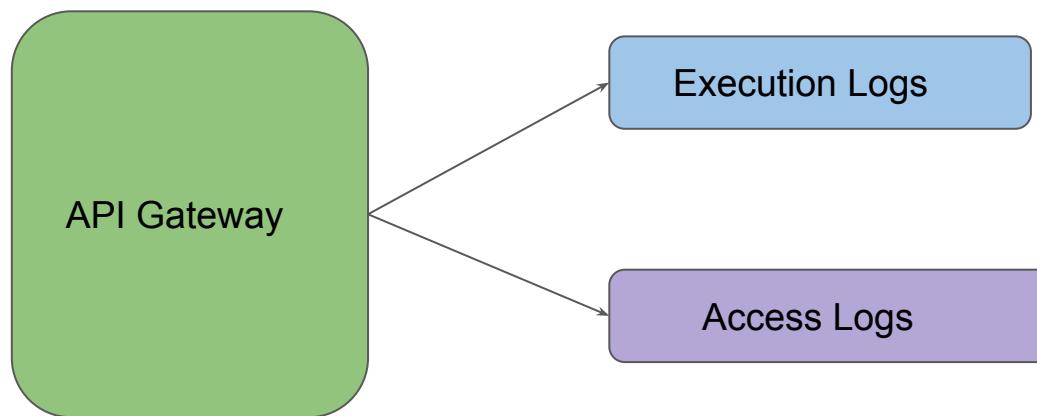


API Gateway Logging

Back to Logging!

Logging at API Gateway Level

Logging at API Gateway allows customers to log calls that are made to the API Gateway along with detailed information as API Gateway goes through each step of processing the request.



1. Execution Logs

Records the API Gateway internal information as the request is processed.
These are fully managed by the API Gateway.

Contains information like:

- The request URL
- The request data received by API Gateway
- The request data sent to the Lambda function
- The response received from the Lambda function
- The response data sent by API Gateway

Timestamp	Message
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Extended Request ID: E7F9GEUHMPH4m- (e242937f-ea62-41c7-8a34-97acc44755c8) Verifying usage plan for request: e242937f-ea62-41c7-8a34-97acc44755c8, API key: API Stage: new120
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) API key authorized because method "GET" does not require API key. Request will not contribute to usage limit.
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Usage plan check succeeded for API Key and API Stage new10000j/Dev
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Starting execution for request: e242937f-ea62-41c7-8a34-97acc44755c8
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) HTTP Method: GET, Resource Path: /
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Method request string: {}
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Method request query string: {}
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Method request headers: {User-Agent:curl/7.51.0, X-Forwarded-Proto:https, X-Forwarded-For:115.99.99.99}
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Method request body before transformations: {}
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint request URL: https://lambda.us-east-1.amazonaws.com:2015-03-31/functions/arn:aws:lambda:us-east-1::242937f-ea62-41c7-8a34-97acc44755c8
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint request headers: {x-amzn-lambda-integration-tag:e242937f-ea62-41c7-8a34-97acc44755c8, Authorization: }
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint request body after transformations: {}
2020-02-19T19:57:18.983+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Sending request to https://lambda.us-east-1.amazonaws.com:2015-03-31/functions/arn:aws:lambda:us-east-1::242937f-ea62-41c7-8a34-97acc44755c8
2020-02-19T19:57:19.049+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Received response: Status: 200, Integration latency: 66 ms
2020-02-19T19:57:19.049+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint response Headers: {Date:Wed, 19 Feb 2020 14:27:19 GMT, Content-Type:application/json, Content-Length:10}
2020-02-19T19:57:19.049+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint response body before transformations: {"body": "Hello from Lambda!"}
2020-02-19T19:57:19.049+05:30	(e242937f-ea62-41c7-8a34-97acc44755c8) Endpoint response body after transformations: {"body": "Hello from Lambda!"}

Useful when a specific request needs troubleshooting.

2. Access Logs

Logs related to who has accessed the API.
Very similar to the Apache / Nginx Logs.

Contains information like:

- The caller's IP address
- The request time
- The request HTTP method
- The request URL
- The response HTTP status code, etc.

Log events		Actions		Query log group					
Filter events		30s	1m	30m	1h	12h	custom	grid	list
▶	Timestamp								
▶	2020-02-19T19:57:18.981+05:30		Message						
			115.99.75.21,-,-,19/Feb/2020:14:27:18 +0000,GET,/,HTTP/1.1,200,33,e242937f-6a62-41c7-8a3						

CloudWatch Metrics for API Gateway

There are certain metrics that are made available in CloudWatch for the API Gateway resource.

Some of these metrics include:

- 5XX Error
- Latency
- Count
- 4XX Error

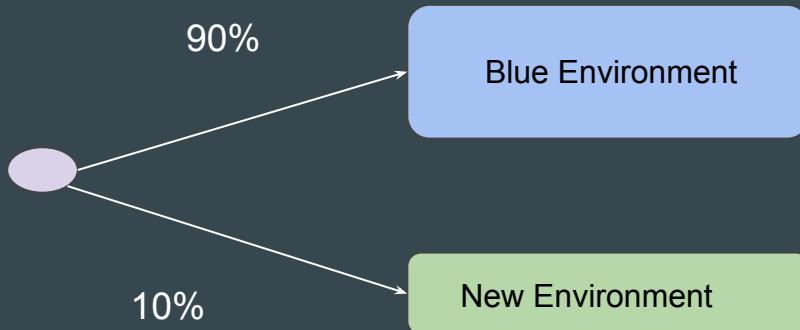


API Gateway Canary Deployment



Revising Canary Deployment Model

Canary Deployment is a process where we deploy a new feature and shift some % of traffic to the new feature to perform some analysis to see if feature is successful.



API Gateway Support

API Gateway **supports** Canary based deployment method.

base-stage Stage Editor

Delete Stage Configure Tags

Invoke URL: <https://y7ml5vfsw1.execute-api.ap-southeast-1.amazonaws.com/base-stage>

Settings Logs/Tracing Stage Variables SDK Generation Export Deployment History Documentation History **Canary**

Manage Canary settings here. A Canary is used to test new API deployments and/or changes to stage variables. A Canary can receive a percentage of requests going to your stage. In addition, API deployments will be made to the Canary first before being able to be promoted to the entire stage.

Promote Canary Delete Canary

Stage's Request Distribution

Percentage of requests directed to **Canary** 50% ↗
Percentage of requests directed to **base-stage** 50%

Canary Deployment

Deployment date Jun 8, 2023 3:26:51 PM

Description No description.

Canary Stage Variables

By default, your Canary inherits stage variables from the stage. You can override these stage variables or add new ones. When promoting a Canary's settings to the stage, the stage is able to update its stage variables to reflect any overridden values and include any new stage variables created by the Canary.

Name	Stage Value	Canary Override Value
version	dev	prod

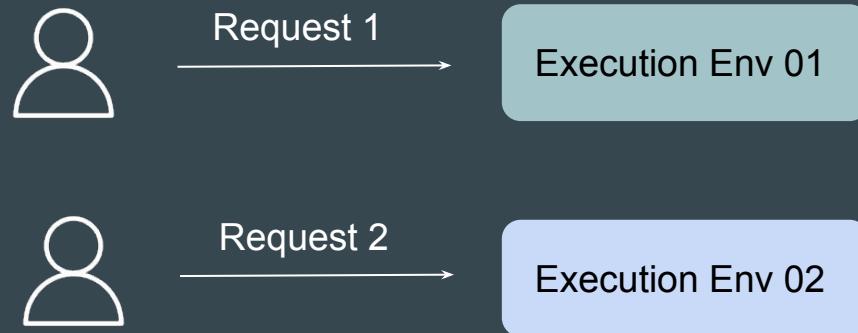
Basics of Lambda Concurrency



Understanding the Basics

Concurrency is the number of in-flight requests your AWS Lambda function is handling at the same time

For each concurrent request, Lambda provisions a separate instance of your execution environment.



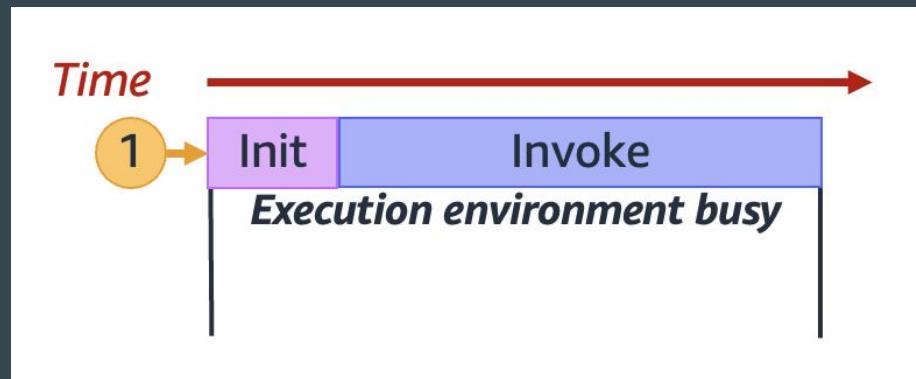
Point to Note

As your functions receive more requests, Lambda automatically handles scaling the number of execution environments until you reach your account's concurrency limit.

Understanding and visualizing concurrency

To handle a request, Lambda **must first initialize** an execution environment (the Init phase), before using it to invoke your function (the Invoke phase)

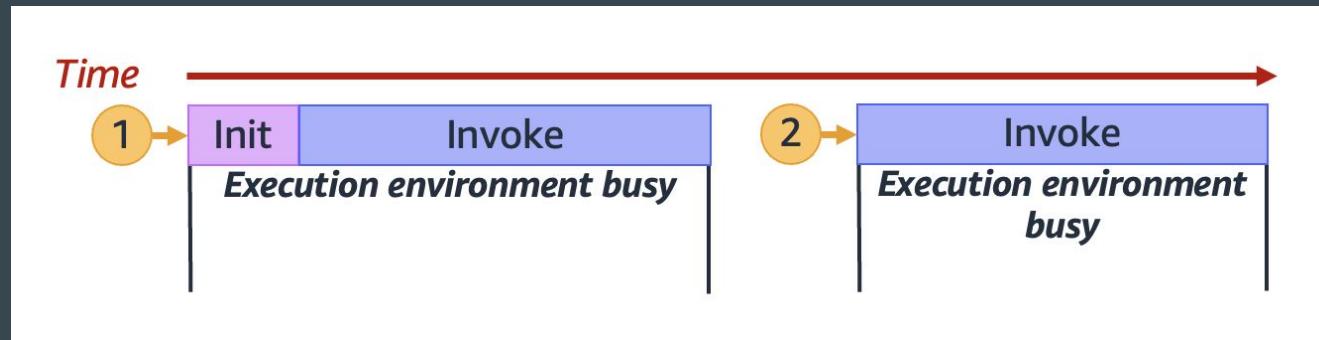
When Lambda finishes processing the first request, this execution environment can then process additional requests for the same function.



Point to Note

Lambda can reuse the same execution environment to handle the second request.

Single instance of your execution environment = Concurrency of 1



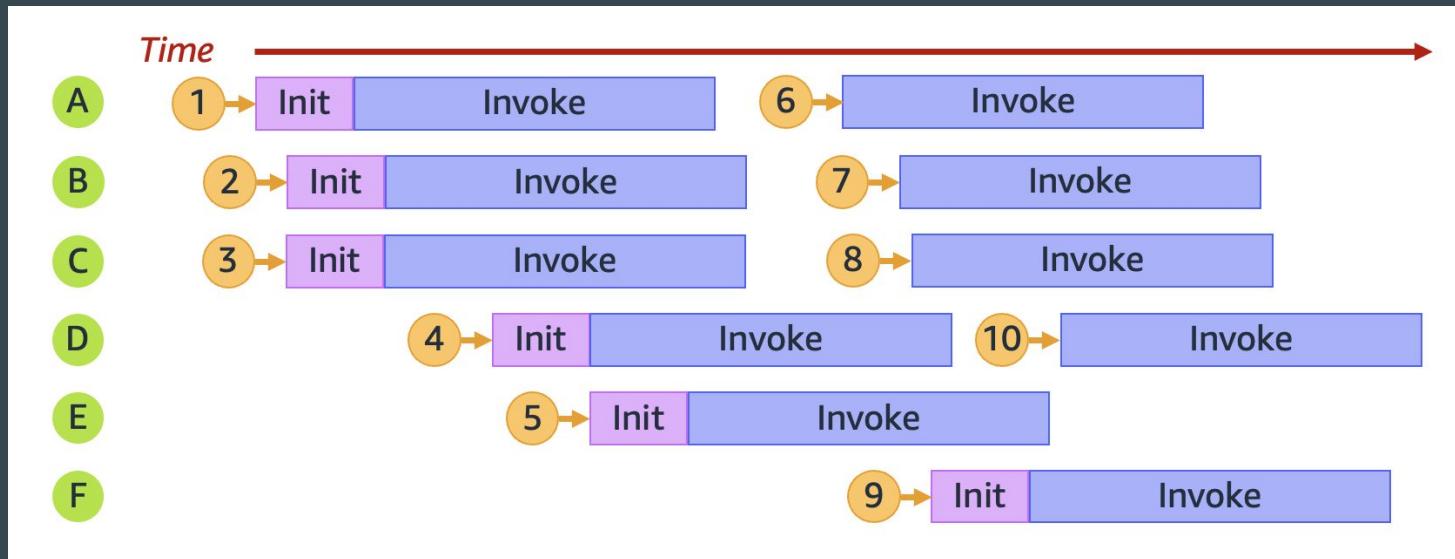
Creating Base Workflow

In real world scenario, Lambda may need to **provision multiple execution environment instances** in parallel to handle all incoming requests.

When your function receives a new request, one of two things can happen:

1. If a pre-initialized execution environment instance is available, Lambda uses it to process the request.
2. Otherwise, Lambda creates a new execution environment instance to process the request.

Sample Workflow - 10 Requests



Reserved and Provisioned Concurrency



Understanding the Basics

The default concurrency limit per AWS Region is 1,000 invocations at any given time

Your functions share this pool of 1,000 concurrency on an on-demand basis.

Your function **experiences throttling** (i.e. it starts to drop requests) if you run out of available concurrency.

Practical Point of View

Some of your functions might be more critical than others.

As a result, you might want to configure concurrency settings to ensure that critical functions get the concurrency they need.

Concurrency = 200

Concurrency = 600

Oops, only 200 left!



Dev Lambda Function

QA Lambda Function

Prod Lambda Function

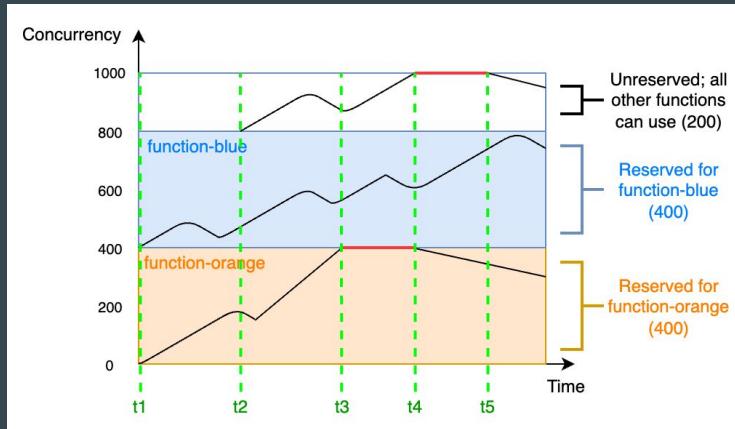
Concurrency Controls

Concurrency Control	Description
Reserved concurrency	<p>Reserve a portion of your account's concurrency for a function.</p> <p>Useful if you don't want other functions taking up all the available unreserved concurrency.</p>
Provisioned concurrency	<p>Pre-initialize a number of environment instances for a function.</p> <p>Useful for reducing cold start latencies.</p>

Reserved concurrency

If you want to guarantee that a certain amount of concurrency is available for your function at any time, use reserved concurrency.

When you dedicate reserved concurrency to a function, no other function can use that concurrency.



Challenge with Reserved Concurrency

You use reserved concurrency to define the maximum number of execution environments reserved for a Lambda function.

However, none of these environments come **pre-initialized**

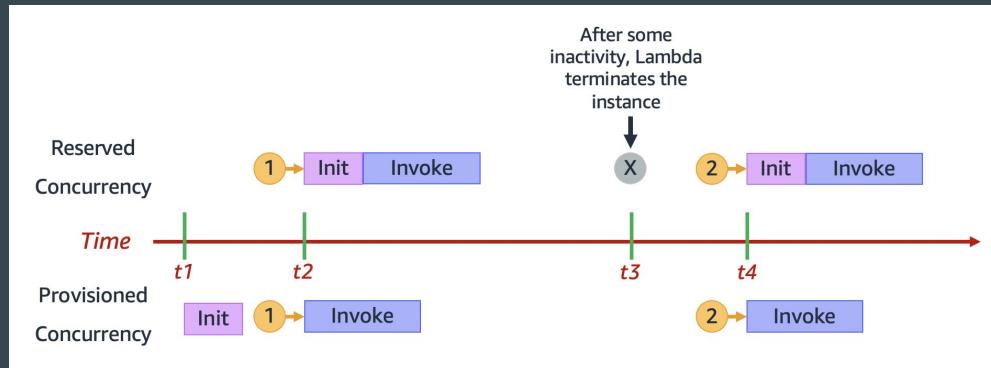
As a result, your function invocations may take longer because Lambda must first initialize the new environment before being able to use it to invoke your function

When initialization takes longer than expected, this is known as a cold start. To mitigate cold starts, you can use provisioned concurrency.

Provisioned concurrency

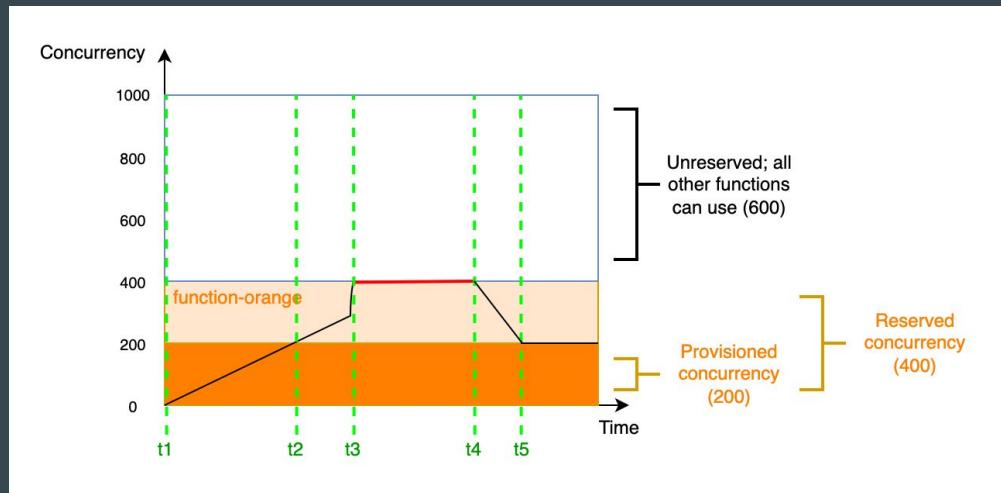
Provisioned concurrency is the number of pre-initialized execution environments you want to allocate to your function

If you set provisioned concurrency on a function, Lambda initializes that number of execution environments so that they are prepared to respond immediately to function requests.



Practical Point of View

You could use provisioned concurrency to set a baseline amount of environments to handle request during weekdays, and use reserved concurrency to handle the weekend spikes.



Comparing Reserved and Provisioned concurrency

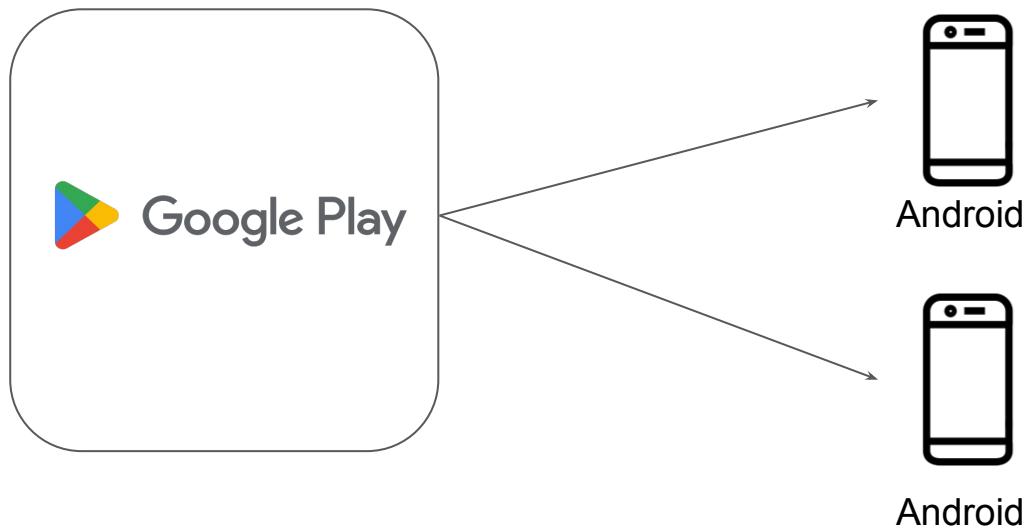
Topic	Reserved concurrency	Provisioned concurrency
Definition	Maximum number of execution environment instances for your function.	Set number of pre-provisioned execution environment instances for your function.
Provisioning behavior	Lambda provisions new instances on an on-demand basis.	Lambda pre-provisions instances (i.e. before your function starts receiving requests).
Cold start behavior	Cold start latency possible, since Lambda must create new instances on-demand.	Cold start latency eliminated, since Lambda doesn't have to create instances on-demand.
Throttling behavior	Function throttled when reserved concurrency limit reached.	If reserved concurrency not set: function uses unreserved concurrency when provisioned concurrency limit reached. If reserved concurrency set: function throttled when reserved concurrency limit reached.
Default behavior if not set	Function uses unreserved concurrency available in your account.	Lambda doesn't pre-provision any instances. Instead, if reserved concurrency not set: function uses unreserved concurrency available in your account. If reserved concurrency set: function uses reserved concurrency.
Pricing	No additional charge.	Incurs additional charges.

Elastic Container Registry (ECR)

Storing Container Images

Understanding with Analogy

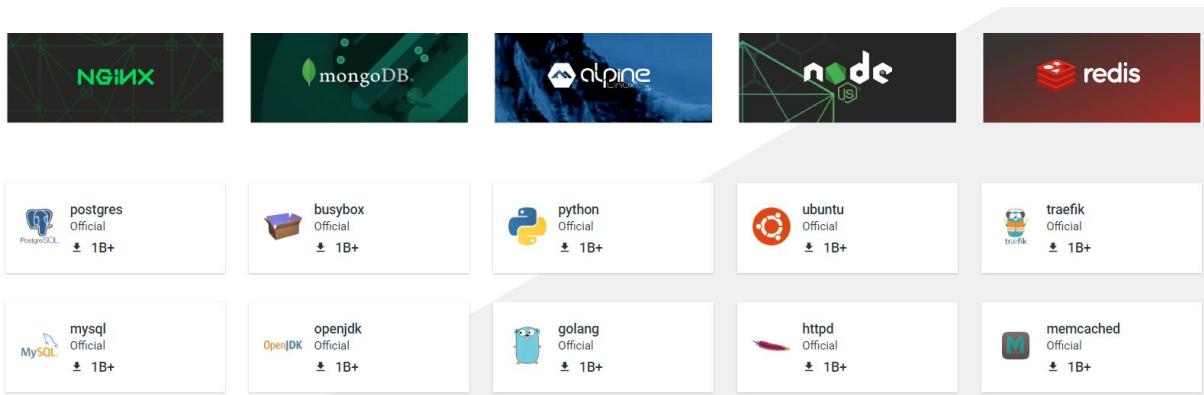
Google Play is an online store where people go to find their favorite apps, games, movies, TV shows, books, and more.



Importance of Container Registry

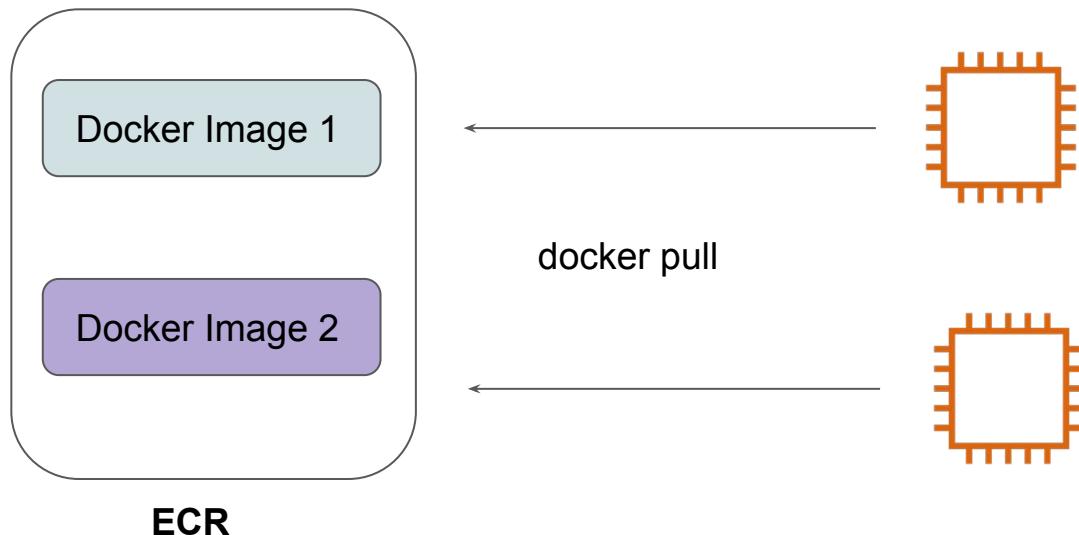
Container Registry is a single place for your team to manage Docker images.

Whenever you launch a Docker Container, the associated image is pulled from Registry.



Basics of ECR

Amazon ECR is a fully managed container registry for storing Docker Images.

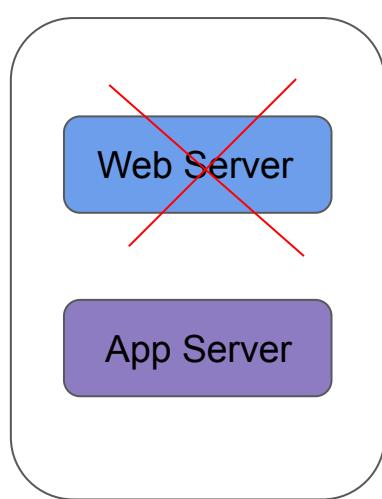


Container Orchestration

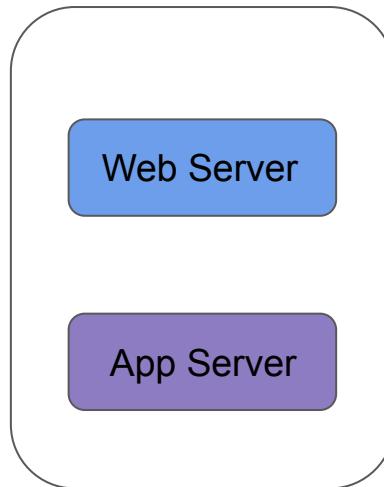
Build once, use anywhere

Getting Started

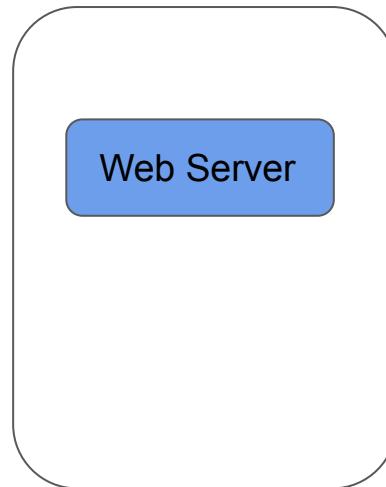
Container orchestration is all about managing the life cycles of containers, especially in large, dynamic environments.



VM 1

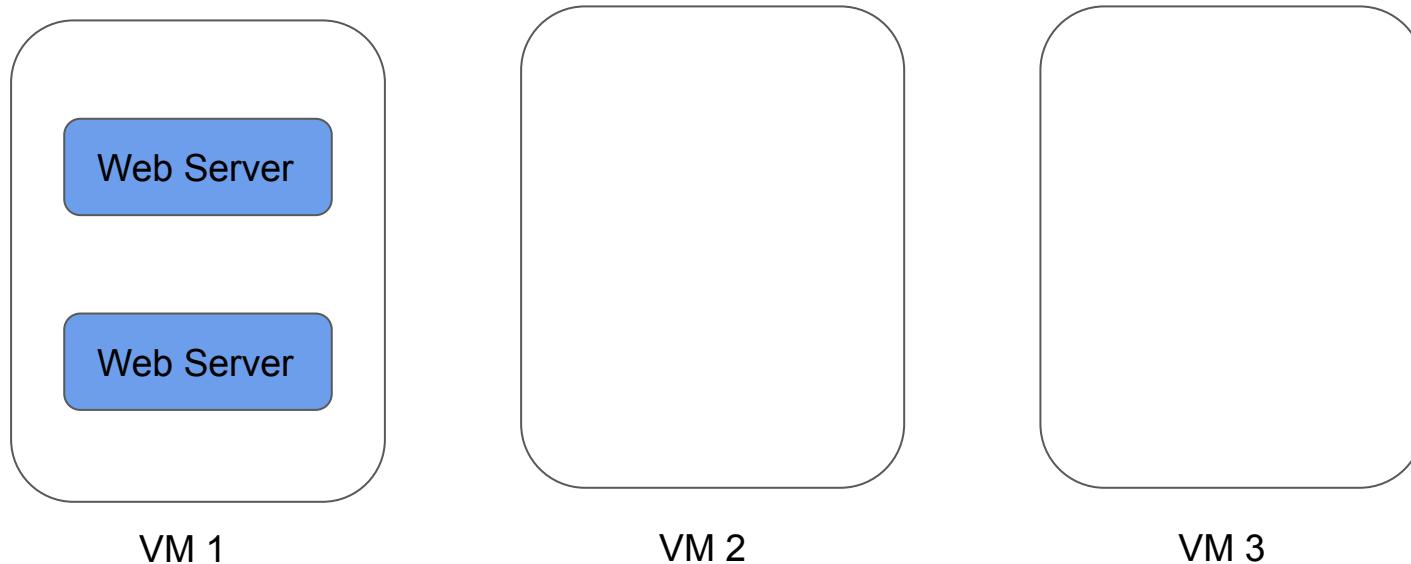


VM 2



VM 3

Requirement: Minimum of 2 web-server should be running all the time.



Importance of Container Orchestration

Container Orchestration can be used to perform lot of tasks, some of them includes:

- Provisioning and deployment of containers
- Scaling up or removing containers to spread application load evenly
- Movement of containers from one host to another if there is a shortage of resources
- Load balancing of service discovery between containers
- Health monitoring of containers and hosts

Container Orchestration Solutions

There are many container orchestration solutions which are available, some of the popular ones include:

- Docker Swarm
- Kubernetes
- Apache Mesos
- Elastic Container Service (AWS ECS)

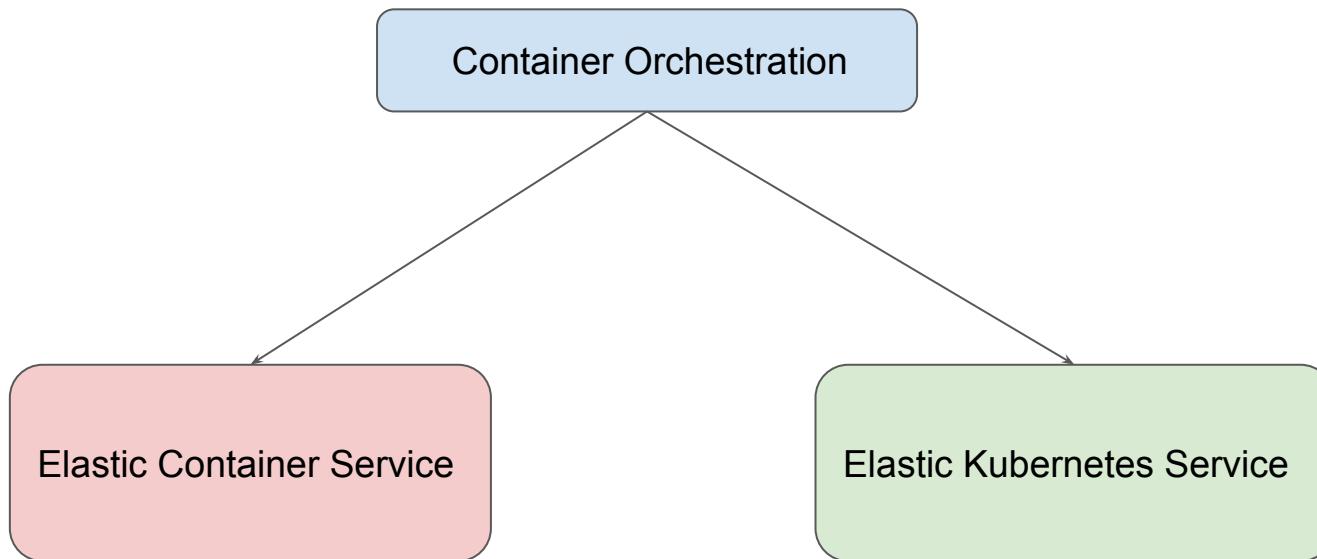
There are also various container orchestration platforms available like EKS.

Container Orchestration in AWS

Choosing Right Orchestrator

Container Orchestration in AWS

There are two primary services that are extensively used for container orchestration use-cases.



Important Difference

Pointers	AWS EKS	AWS ECS
Open-Source	Yes	No
Complexity	More Complex	Less Complex
Community Support	More	Less

Choosing Right Orchestrator

If you plan to work exclusively on AWS, you should choose ECS as it offers more in-depth AWS integration than Amazon EKS.

Organizations with limited expertise and insufficient resources to invest in learning Kubernetes can go with ECS.

If you plan to deploy containers across multiple platforms, you can choose EKS.

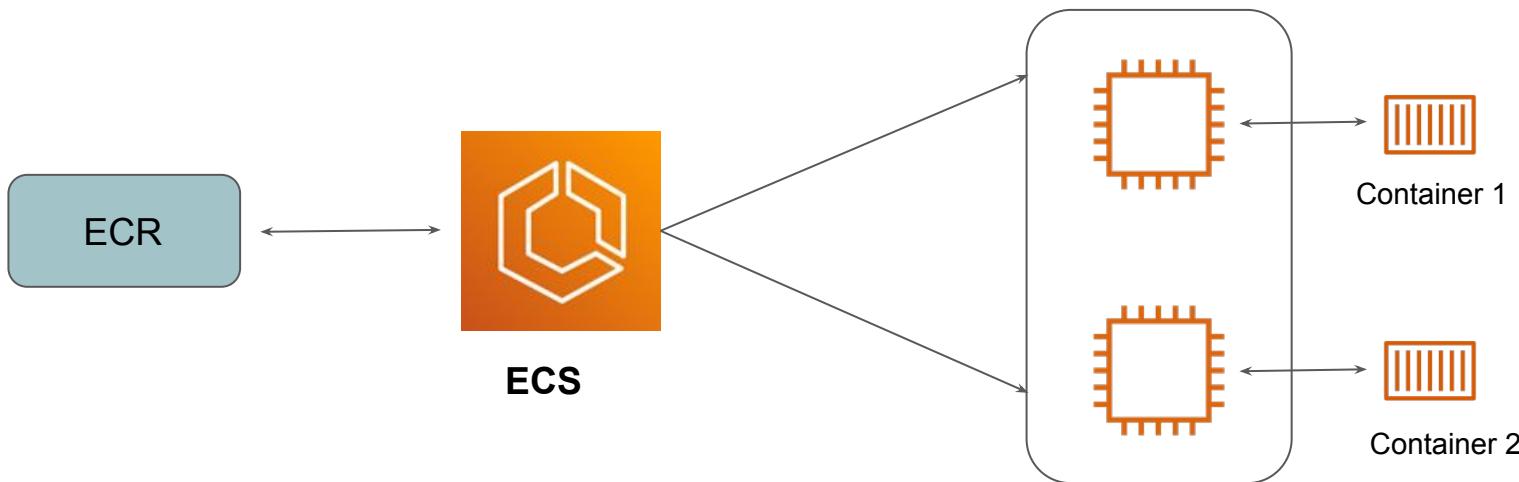
Elastic Container Service (ECS)

Container Management

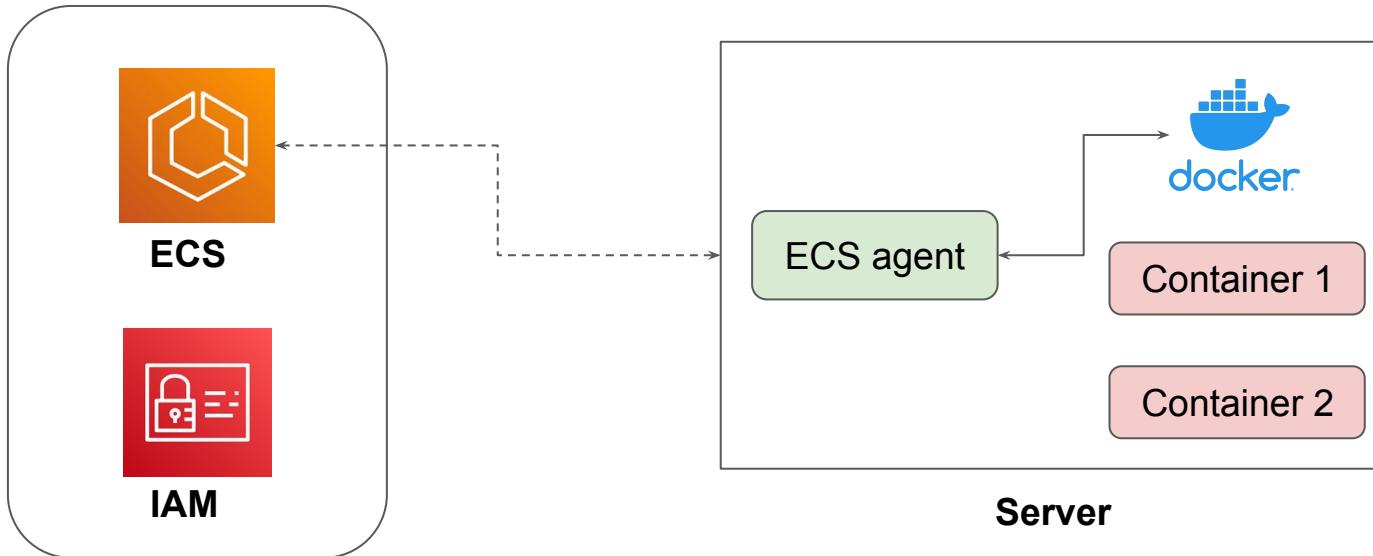
Basics of Service

Amazon Elastic Container Service (Amazon ECS) is a highly scalable and fast container management service.

You can use it to run, stop, and manage containers on a cluster.



High-Level Workflow



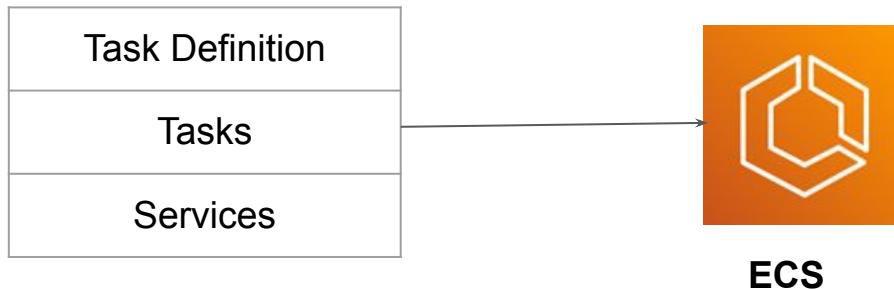
Components of ECS

Container Management

Basic Components

There are three primary components of ECS Cluster:

Task Definition, Tasks and Service



Component - Task Definition

A task definition is a text file that describes one or more containers that form your application.

It contains information like operating system, containers to use, ports to open, storage

Container - 1 [Info](#)

[Essential container](#) [Remove](#)

Container details
Specify a name, container image, and whether the container should be marked as essential. Each task definition must have at least one essential container.

Name	Image URI	Essential container
nginx	nginx:latest	Yes ▾

Port mappings [Info](#)
Add port mappings to allow the container to access ports on the host to send or receive traffic. Any changes to port mappings configuration impacts the associated service connect settings.

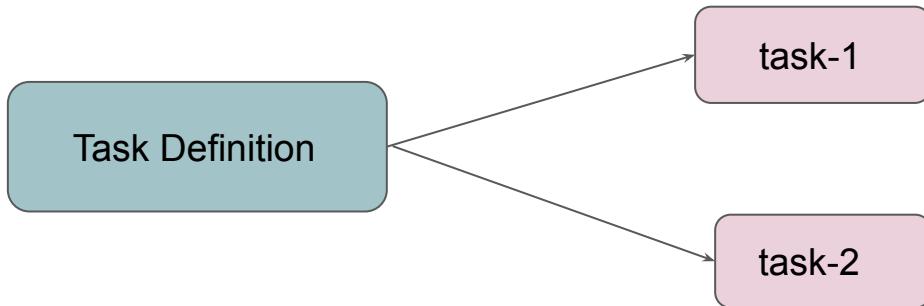
Host port	Container port	Protocol	
80	80	TCP ▾	Remove

[Add more port mappings](#)

Component - Task

A task is the instantiation of a task definition within a cluster.

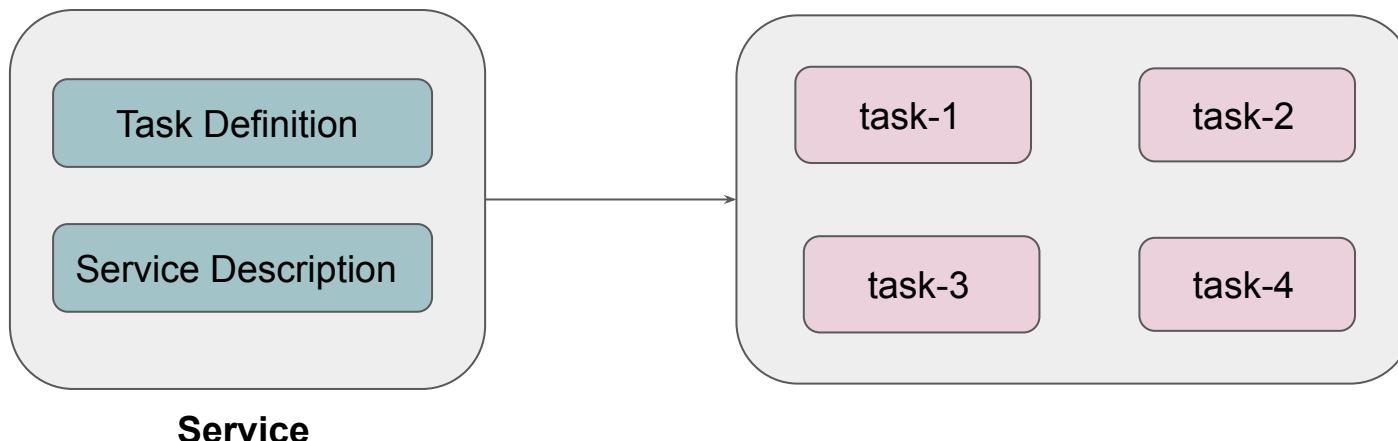
After you create a task definition for your application within Amazon ECS, you can specify the number of tasks to run on your cluster.



Component - Service

Service to run and maintain your desired number of tasks simultaneously in an Amazon ECS cluster.

If any of your tasks fail or stop for any reason, the Amazon ECS service scheduler launches another instance based on your task definition



Introduction to Kubernetes

Orchestrator Engine

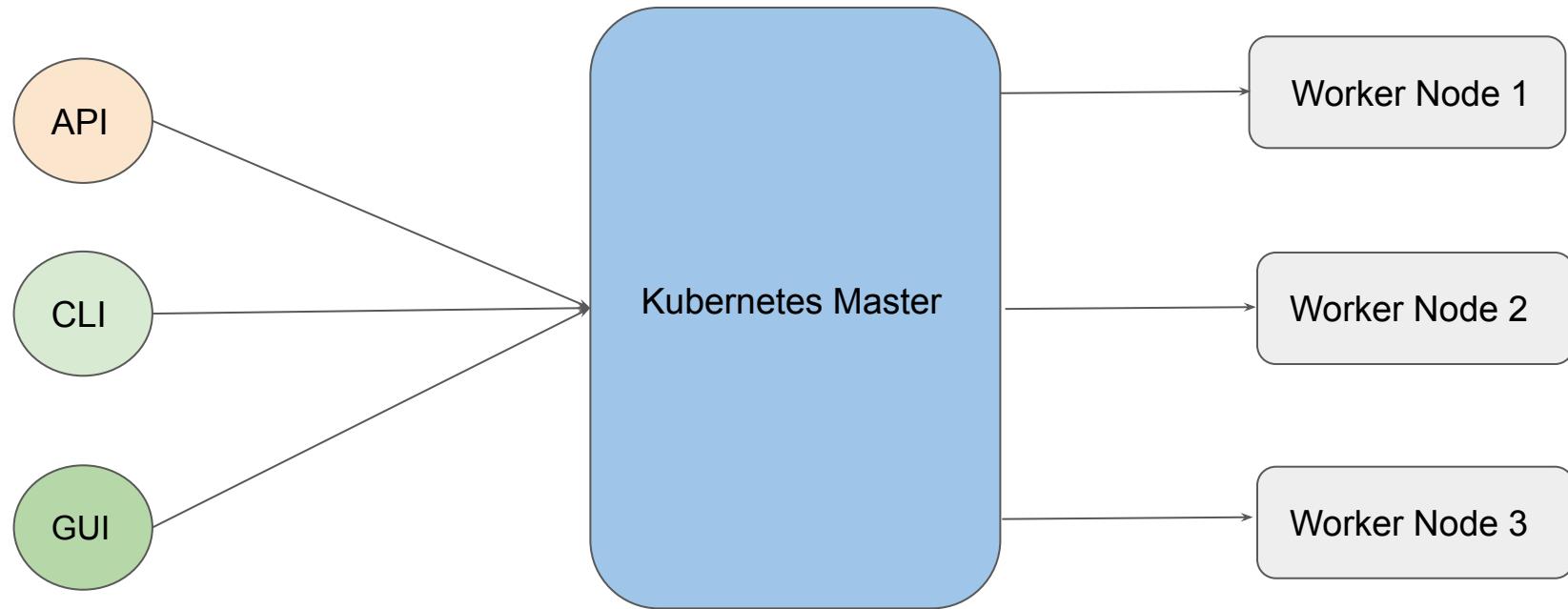
Introduction to Kubernetes

Kubernetes (K8s) is an open-source container orchestration engine developed by Google.

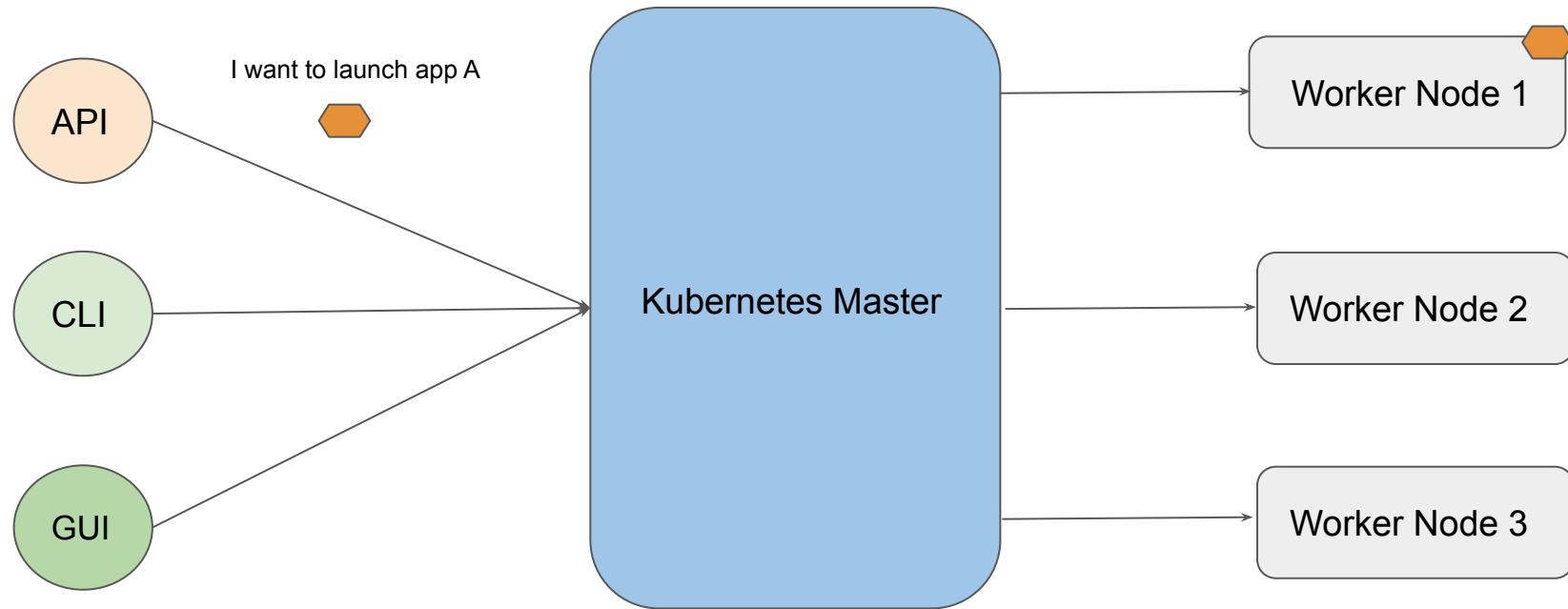
It was originally designed by Google, and is now maintained by the Cloud Native Computing Foundation.



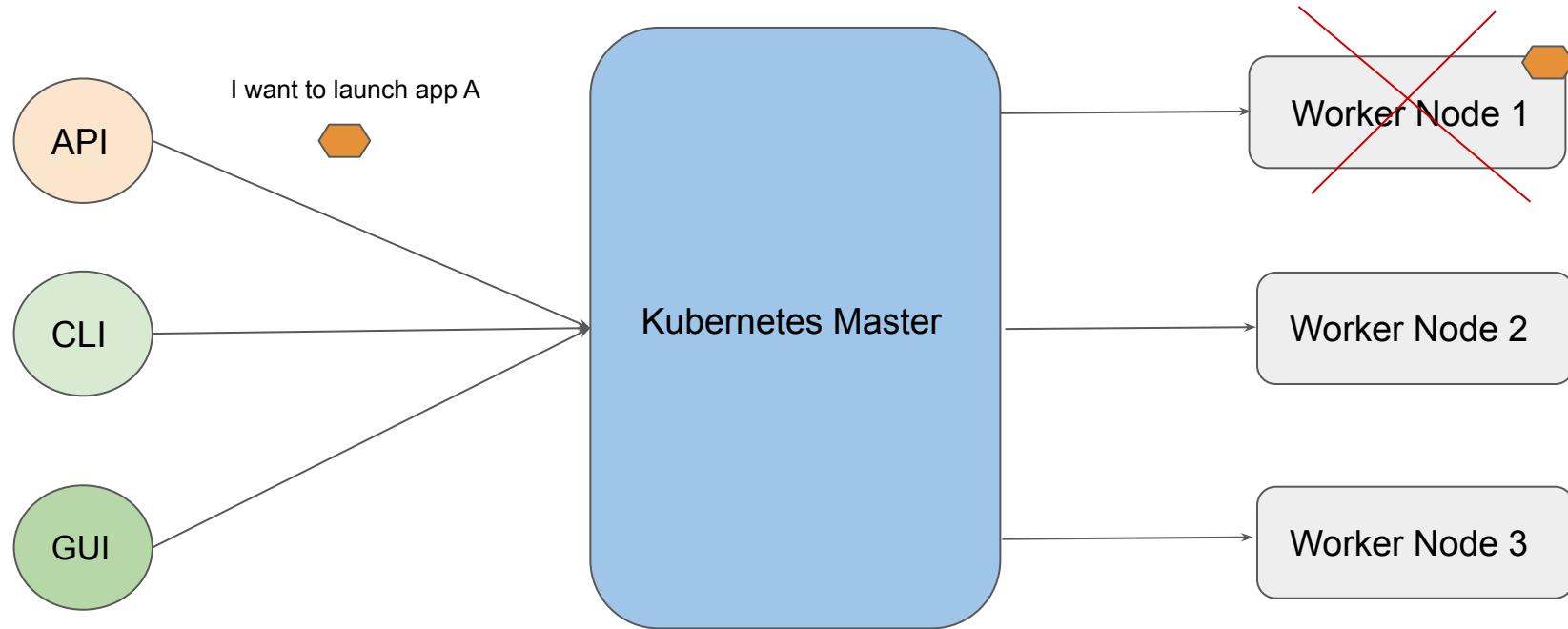
Architecture of Kubernetes



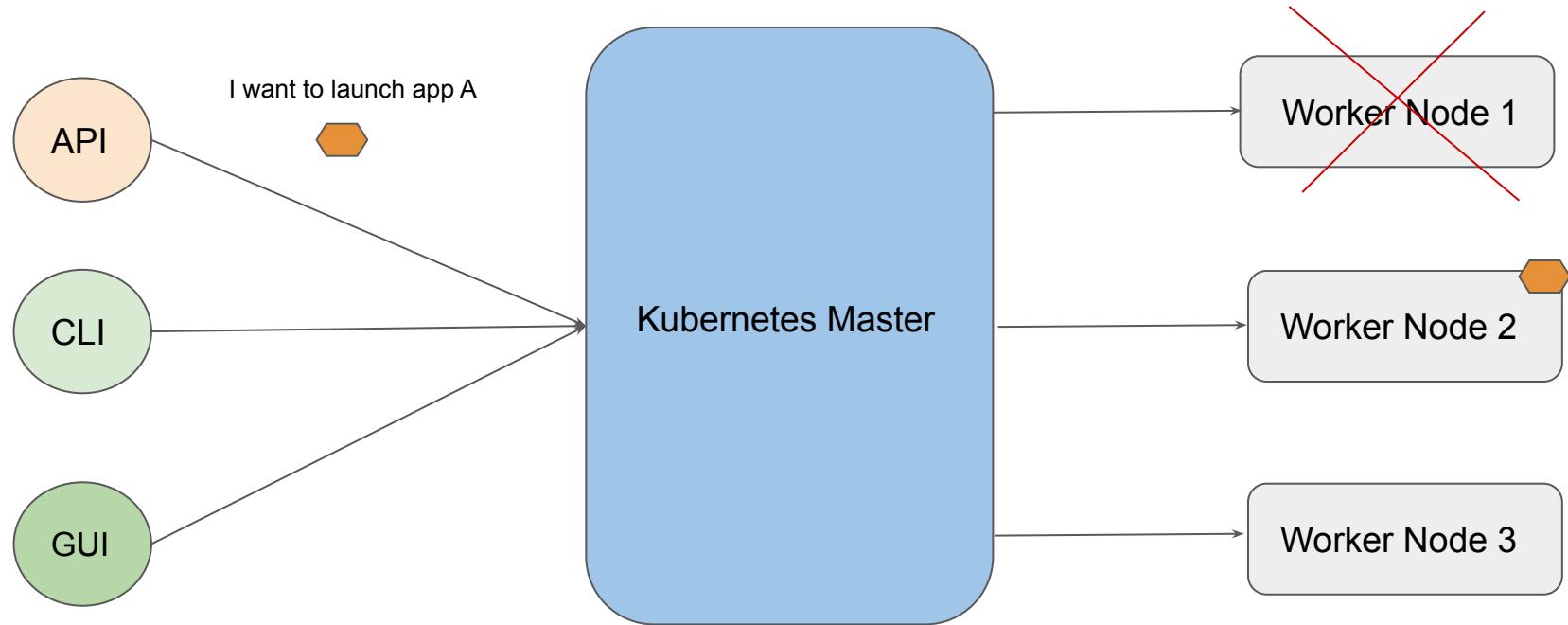
Architecture of Kubernetes



Architecture of Kubernetes



Architecture of Kubernetes

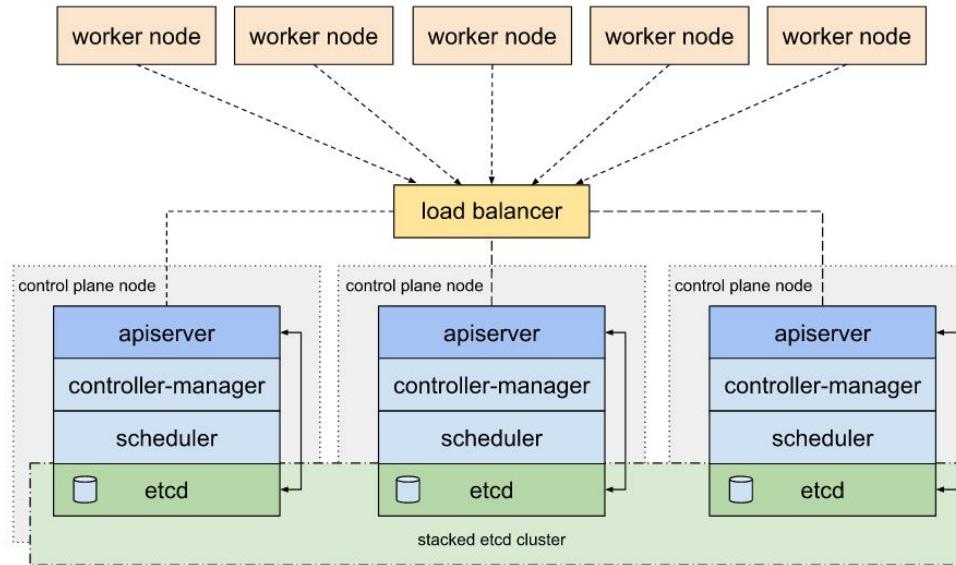


Elastic Kubernetes Service

Managed Kubernetes in AWS

Operating Kubernetes is Hard

Building and Maintaining entire Kubernetes cluster takes lot of time and resources.

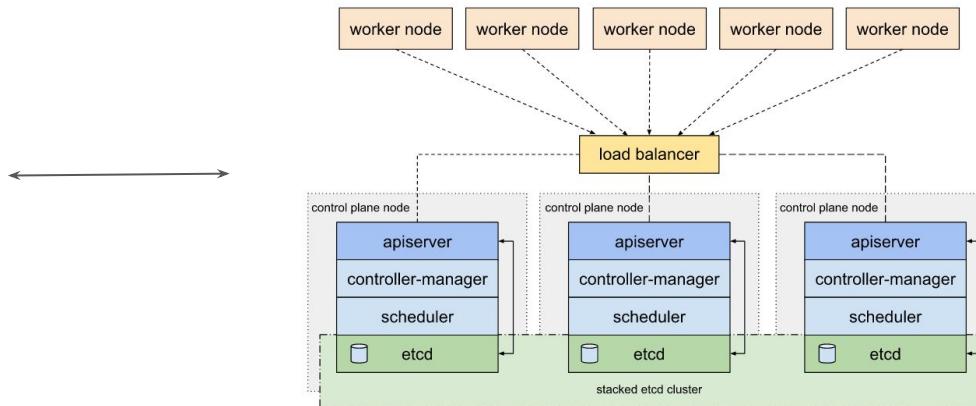


Understanding the Basics

Amazon Elastic Kubernetes Service (Amazon EKS) is a managed service that you can use to run Kubernetes on AWS.

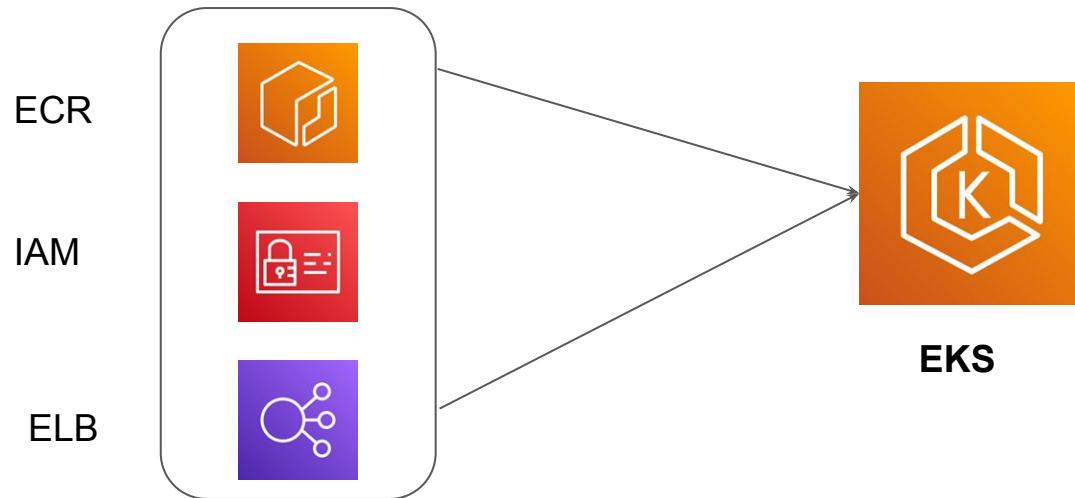


EKS



Benefits of EKS

EKS provides tight integration with various other AWS services like ECR, IAM, ELB to provide end to end features for application deployments.

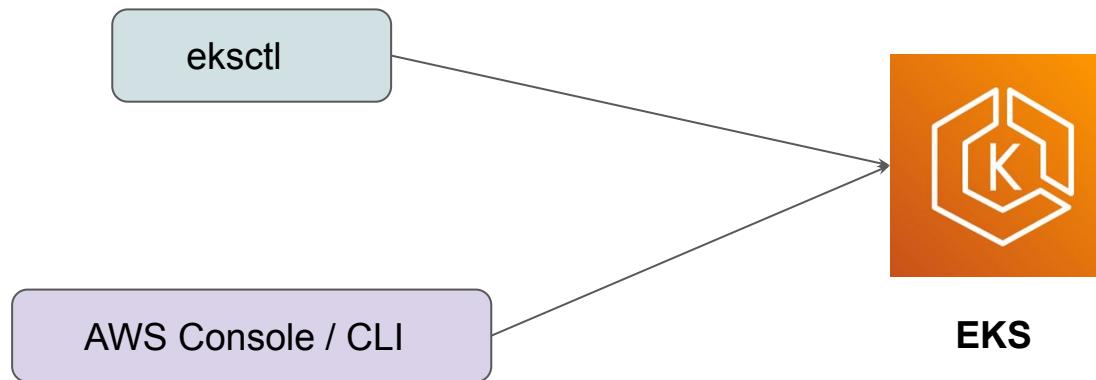


EKS Practical Steps

Let's Create EKS Cluster

Approaches to Create EKS Cluster

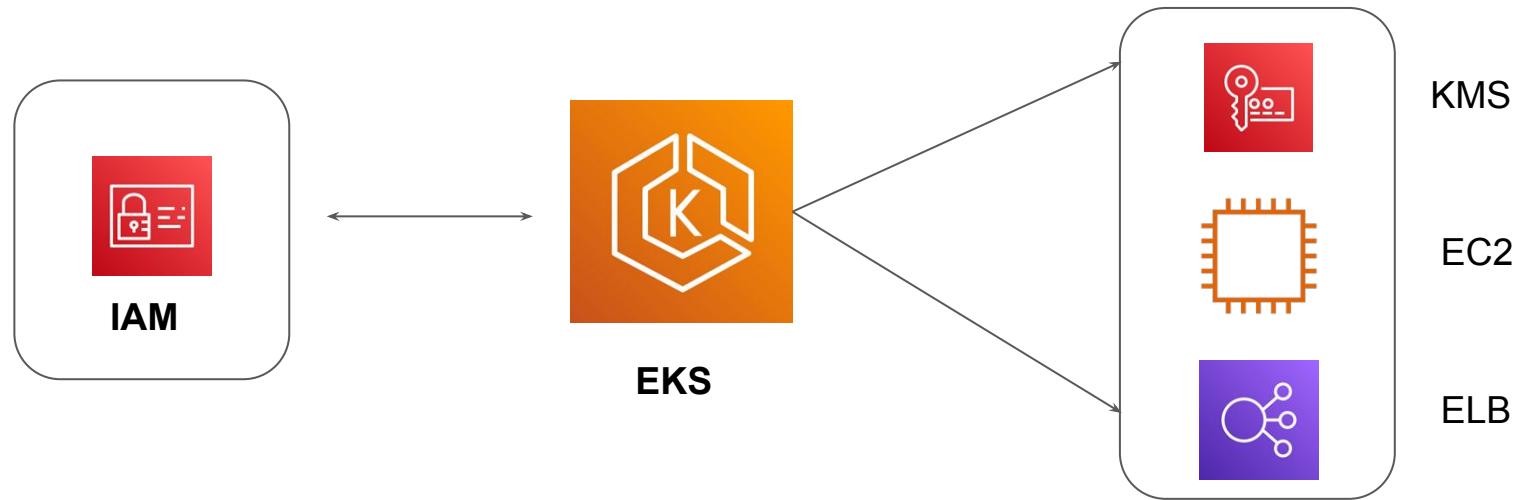
There are two primary ways to create EKS Cluster



Step 1 - Build EKS Cluster

In this step, we build the base EKS Cluster.

Appropriate IAM Role needs to be associated so EKS can manage resource on your behalf.



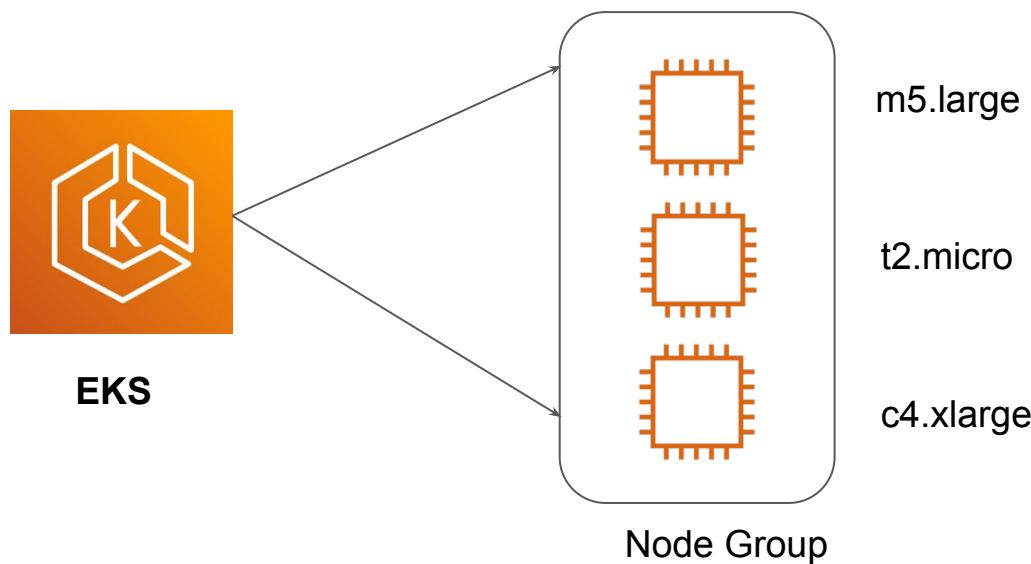
Important Configuration - Building Cluster

Configs	Description
Kubernetes Version	Sets the K8s Version for your cluster.
Cluster Service Role	Allows EKS to manage resources.
VPC	VPC for cluster resources.
Cluster Endpoint Access	Public / Private Access to EKS Cluster.
Networking Add-Ons	To Configure appropriate networking in cluster.
Logging	Enable Logging for K8s Components.

Step 2 - Create Node Group

A node group is a group of EC2 instances that supply compute capacity to your Amazon EKS cluster.

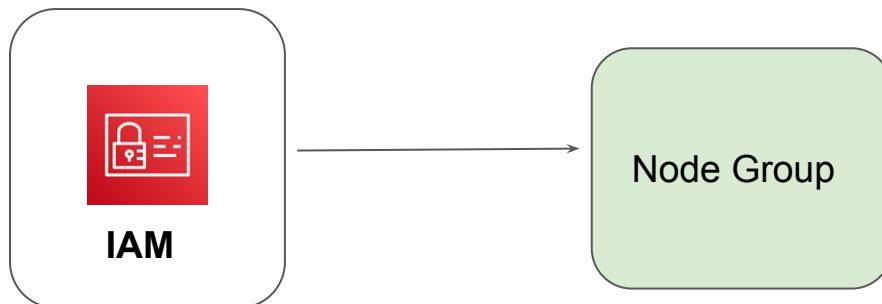
Configuration: AMI ID, Instance Type, Auto-Scaling Configuration, Disk Size.



IAM Role for NodeGroup

An IAM Role needs to be associated with NodeGroup to ensure EC2 instance can perform following operations:

Fetch Images from ECR, Manage Network Interfaces, and others.



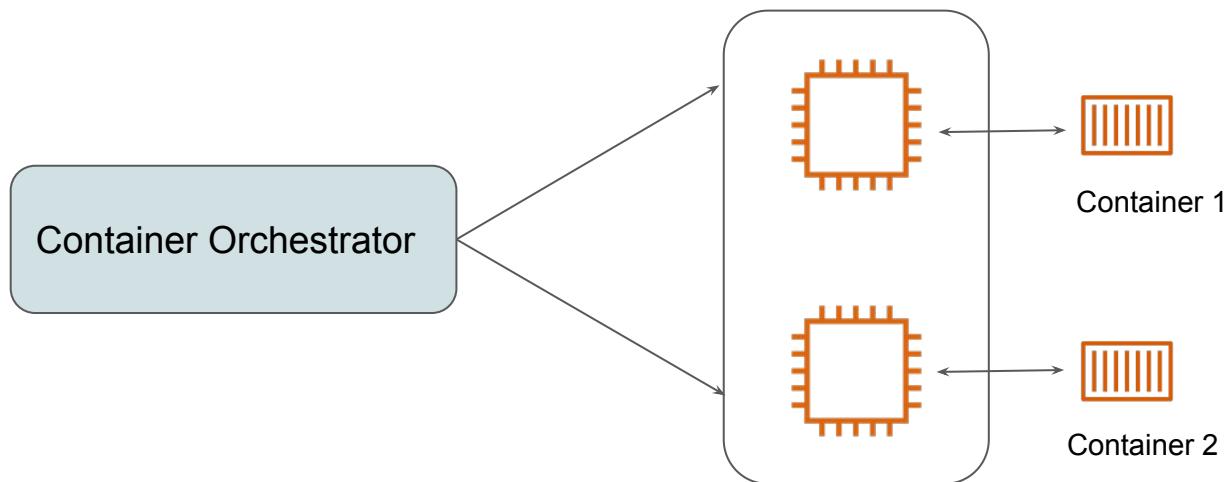
AWS Fargate

Serverless for Containers

Container Management in EC2

In generic scenario, Container Orchestrator will deploy Containers in EC2 instances that are running in your AWS environments.

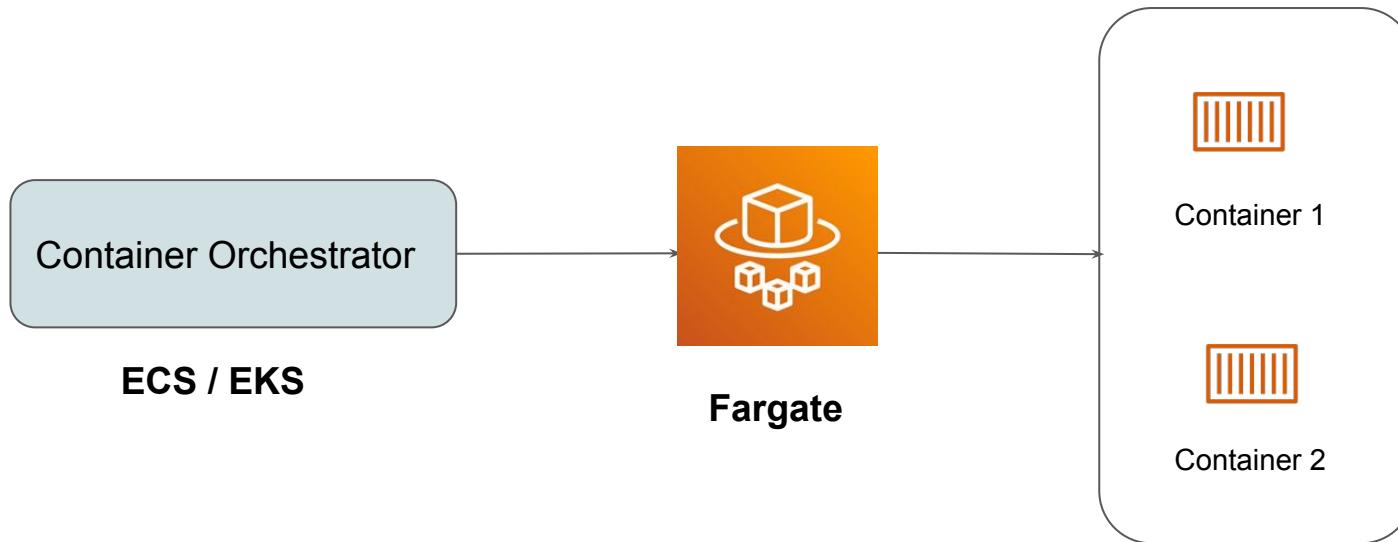
Challenge: Vulnerability Assessment, Patch Management, Security of EC2 and others.



Overview of Fargate

AWS Fargate is a **serverless, pay-as-you-go** compute engine that lets you focus on building applications without managing servers.

No need to maintain servers. AWS takes care of it from their side.



Amazon FSx



Basics of Filesystems

There are **multiple popular set of file systems / storage platforms** available in the industry that are used extensively based on specific use-cases

File Systems	Description
Lustre	Parallel distributed file system, generally used for large-scale cluster computing (HPC)
Open ZFS	Encompasses the functionality of traditional file systems and logical volume manager. Benefits: protection against data corruption, efficient data compression etc

Understanding the Challenges

Many organizations have use-case to leverage the rich feature sets and fast performance of widely-used open source and commercially-licensed file systems.

This would lead to lot of time-consuming administrative tasks like hardware provisioning, software configuration, patching, and backups.

Introduction to FSx

Amazon FSx makes it easy and cost effective to launch and run popular file systems.

It provides cost-efficient capacity and high levels of reliability, and integrates with other AWS services so that you can manage and use the file systems in cloud-native ways.

FSx_N
Amazon FSx
for NetApp ONTAP

FSx_Z
Amazon FSx
for OpenZFS

FSx_W
Amazon FSx
for Windows File Server

FSx_L
Amazon FSx
for Lustre

Benefits of FSx

Benefits	Description
Simple and fully managed	<p>In minutes and with a few clicks, you can launch a fully managed file system.</p> <p>No need to worry about configuring, patching, backups etc.</p>
Secure and compliant	<p>Amazon FSx automatically encrypts your data at-rest and in-transit.</p> <p>Complies with PCI-DSS, ISO, SOC certifications</p>
Integration with AWS services	<p>Integrate with AWS services, including Amazon S3, AWS KMS, Amazon SageMaker, Amazon WorkSpaces, AWS ParallelCluster.</p>

Amazon FSx for Lustre

Provides cost-effective, high-performance, scalable file storage for compute workloads such as machine learning, high performance computing (HPC), video processing, and financial modeling.

Integrates seamlessly with Amazon S3, SageMaker, EKS etc

Amazon FSx for Windows File Server

Provides simple, fully managed, highly reliable file storage that's accessible over the industry-standard Server Message Block (**SMB**) protocol.

Built on Windows Server, providing full SMB support and a **wide range of administrative features** like user quotas, data deduplication, and end-user file restore. Accessible from Windows, Linux, and macOS.

Integrates with Microsoft Active Directory (AD) to support Windows-based environments and enterprises.

FSx for OpenZFS

Provides simple, cost-effective, high-performance file storage built on the OpenZFS file system accessible over the industry-standard **NFS** protocol.

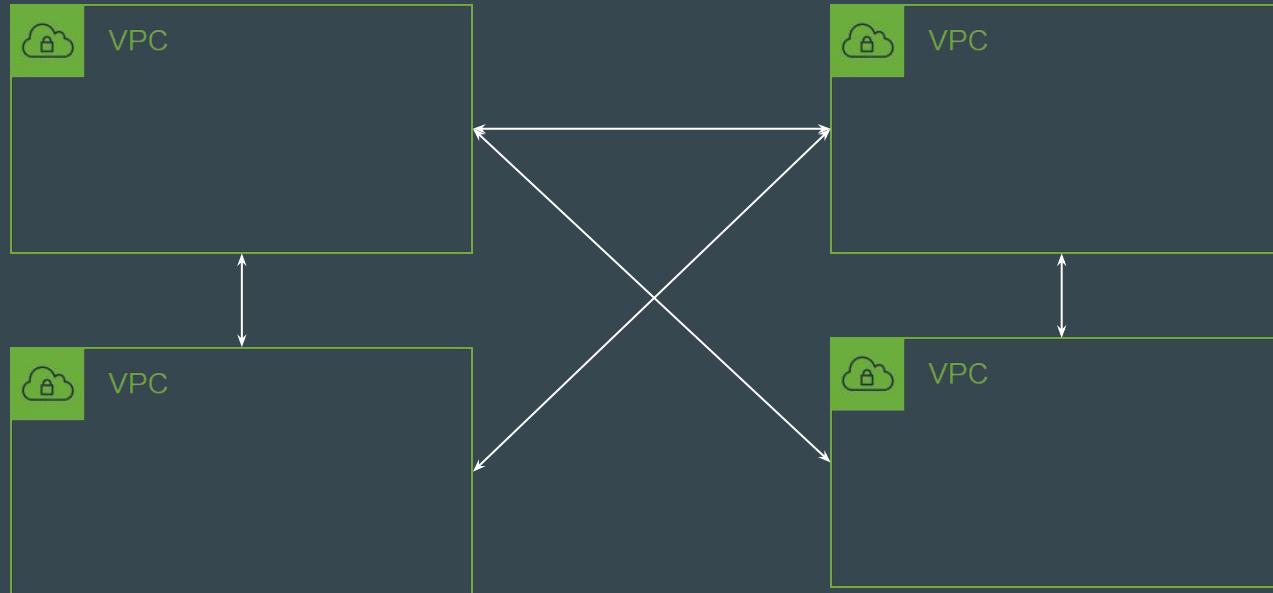
Provides powerful OpenZFS data management capabilities including Z-Standard/LZ4 compression, instant point-in-time snapshots, and data cloning, thin provisioning, and user/group quotas.

Transit Gateways



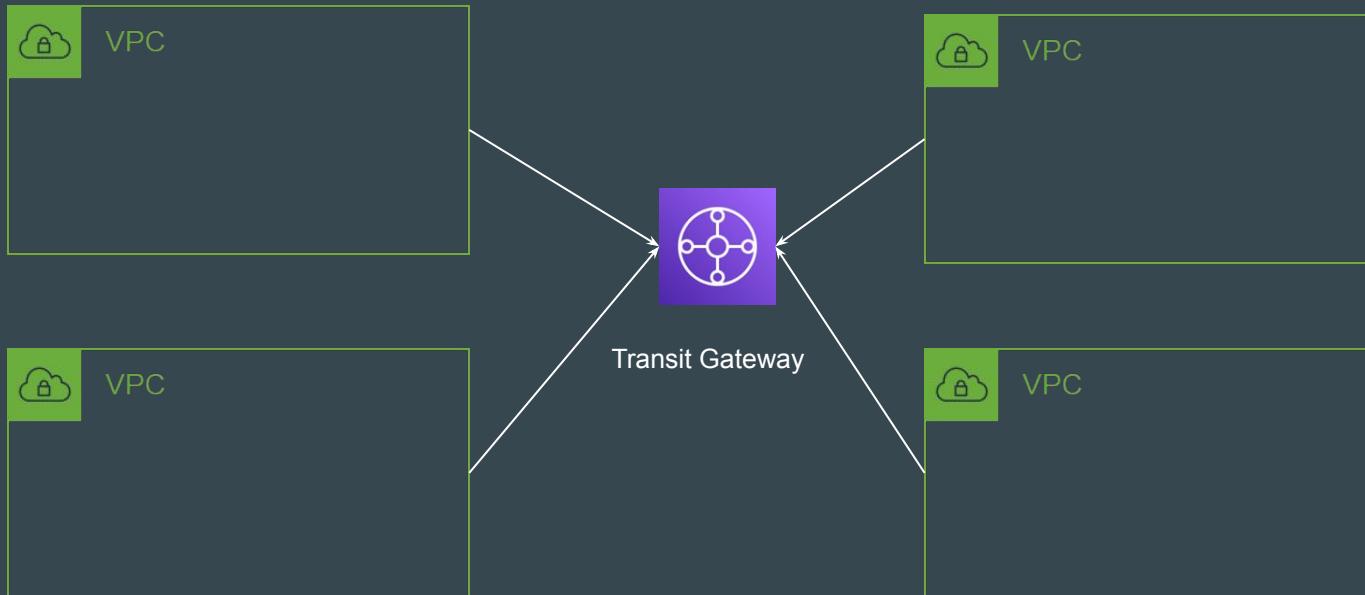
Use-Case: Connecting 4 VPCs

More the Number of VPCs, more the number of peering connection you have to establish for inter-connectivity related use-case.

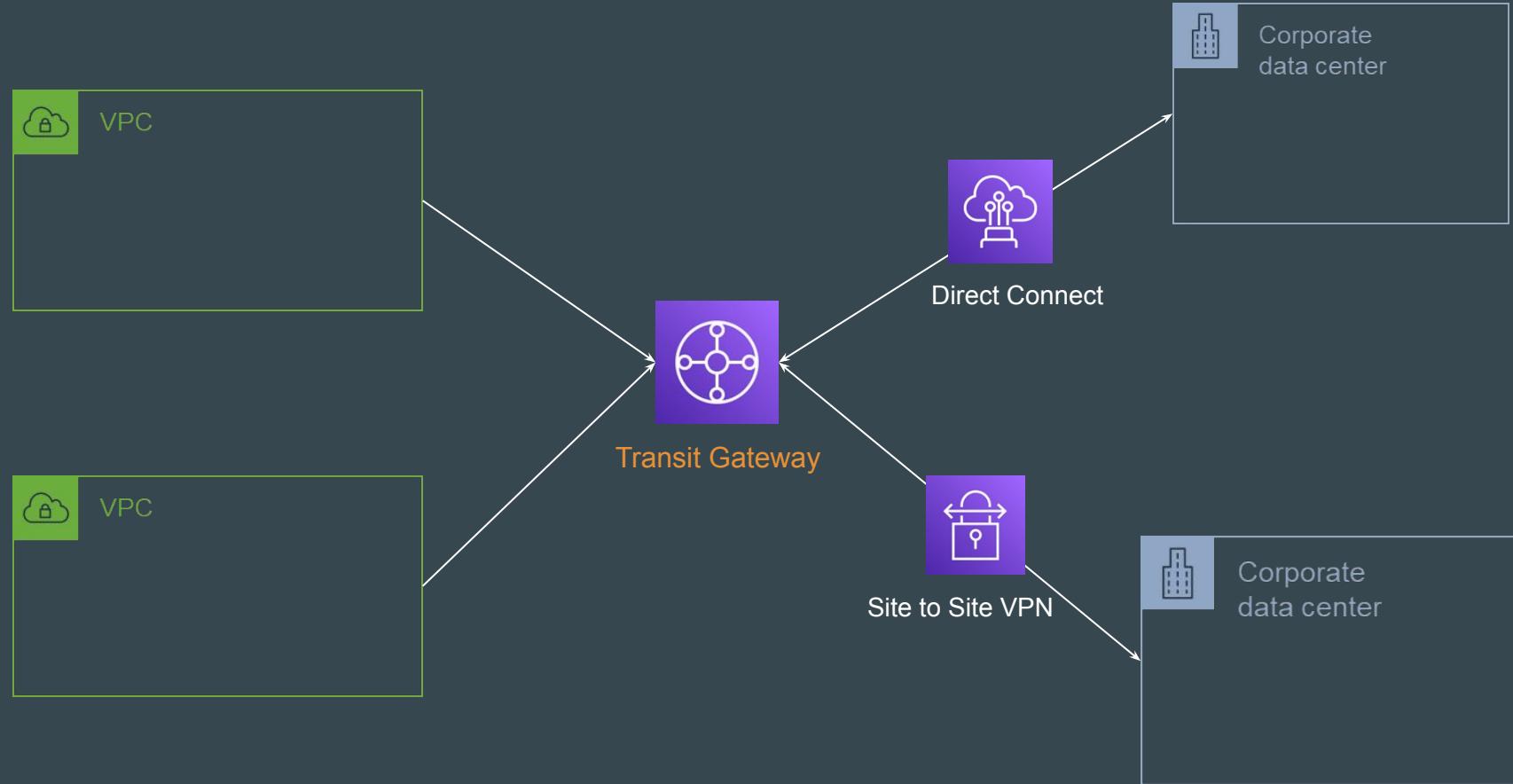


Introducing Transit Gateway

AWS Transit Gateway **connects** your Amazon Virtual Private Clouds (VPCs) and on-premises networks through a central hub



Larger Setup



Amazon FSx for NetApp ONTAP

Provides feature-rich, high-performance, and highly-reliable storage built on NetApp's popular ONTAP file system and fully managed by AWS.

Accessible via industry-standard NFS, SMB, and iSCSI protocols.

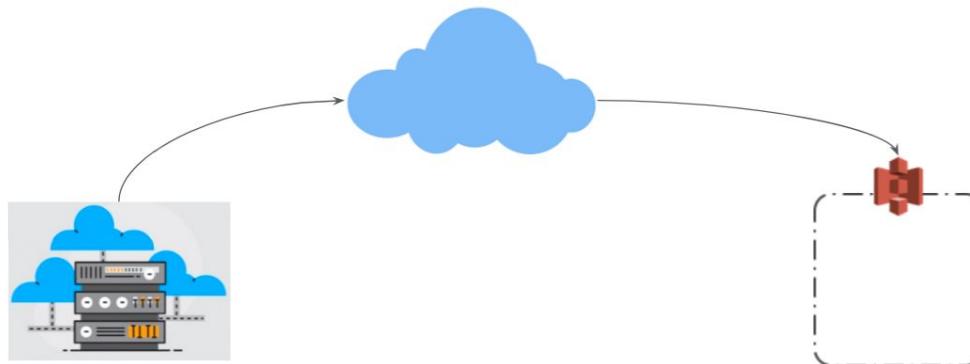
Integrates with Microsoft Active Directory (AD) to support Windows-based environments and enterprises.

VPC Endpoints

Private Communication is Better

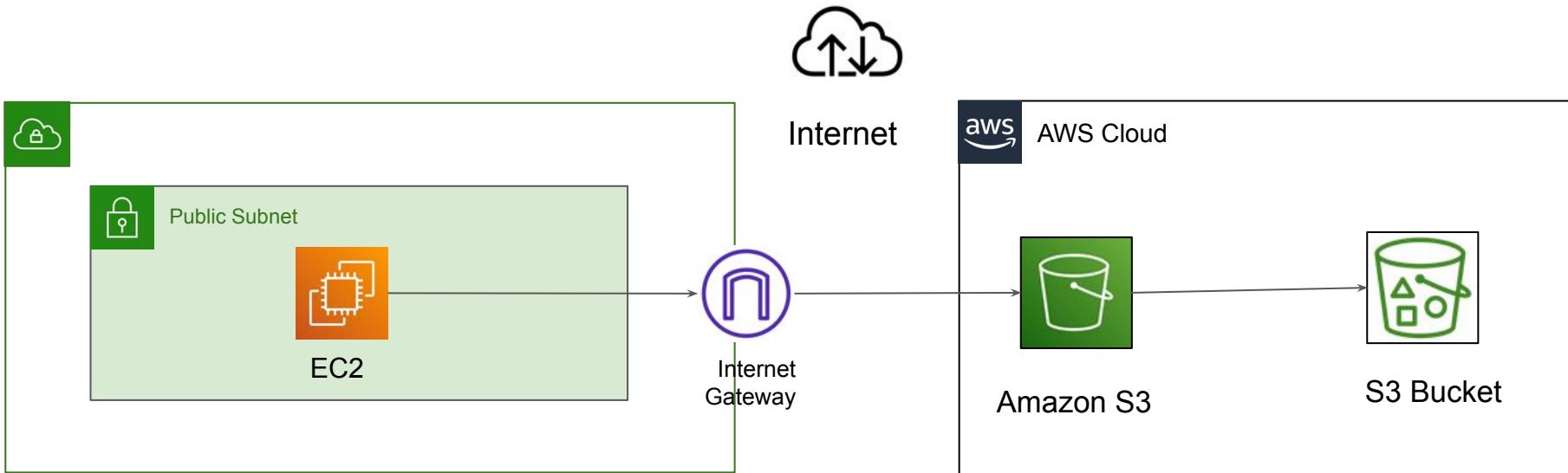
Use-Case: EC2 and S3 Communication

For EC2 instances to be able to access public resources like S3, DynamoDB and others, the traffic needed to be passed via Internet Gateway.

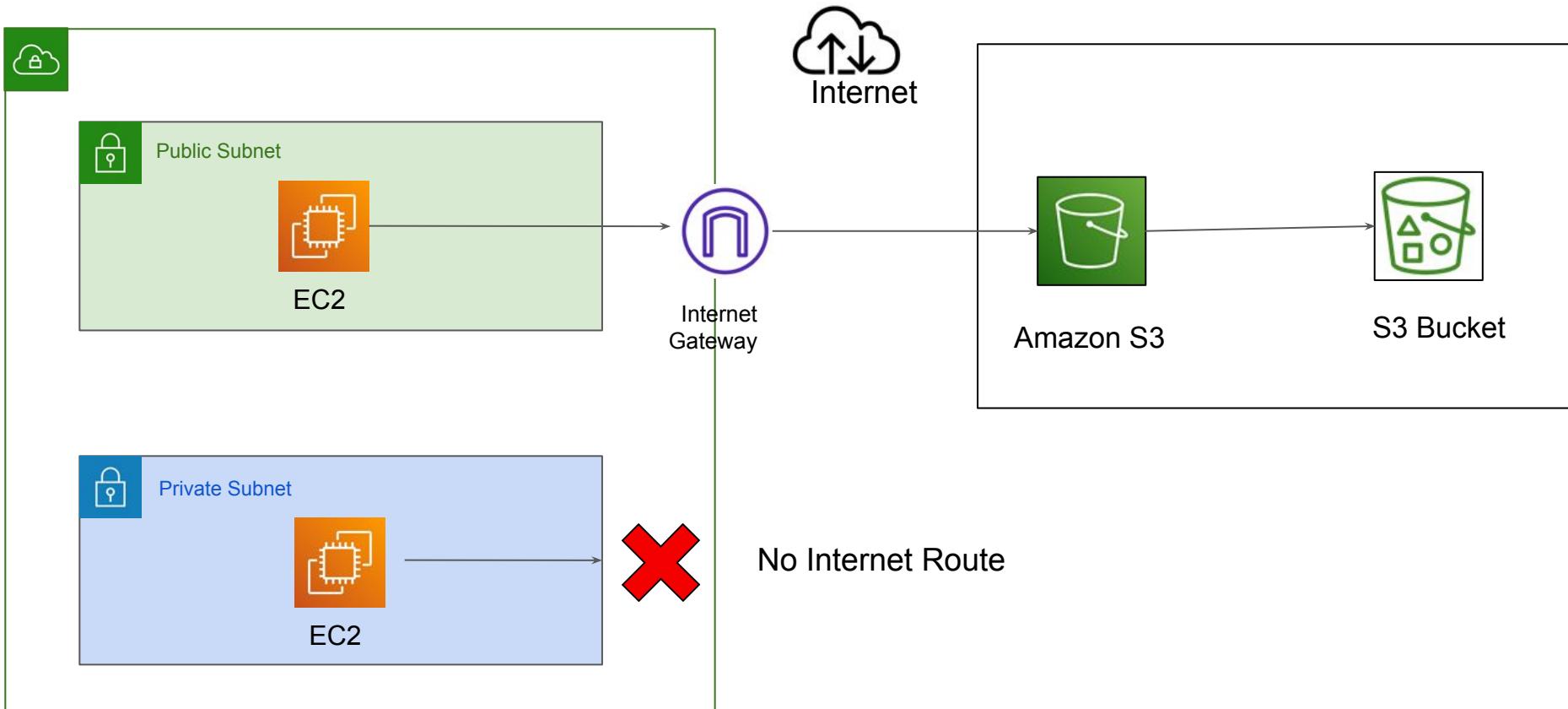


Architectural Perspective

EC2 traffic towards S3 is routed to Internet Gateway



Challenge with Private Workloads

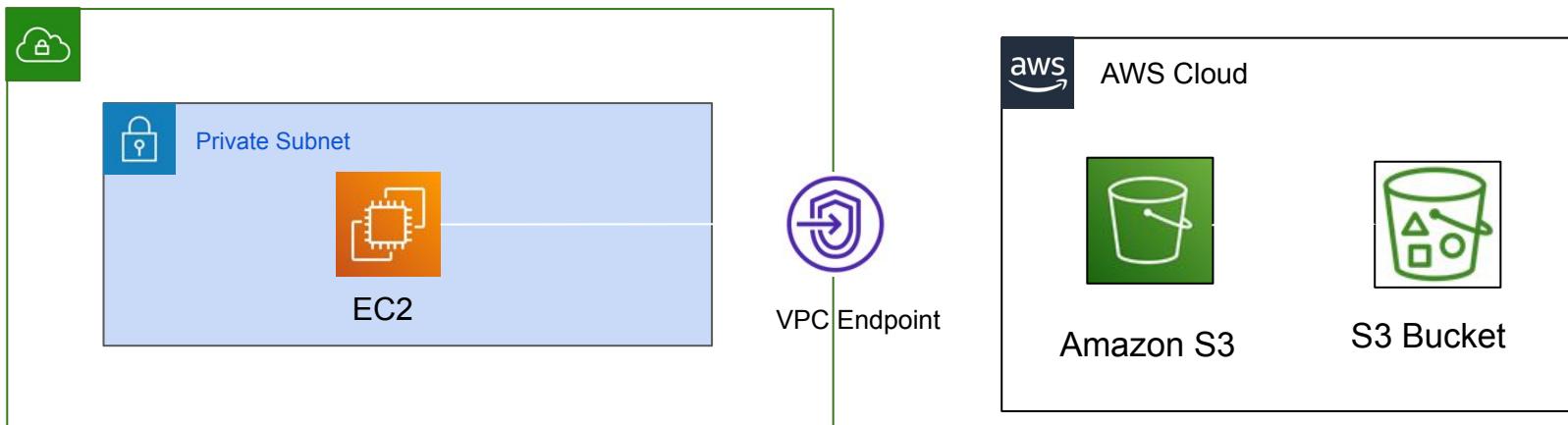


Downsides of Using Public Internet

1. Data transfer cost of AWS
2. Higher Latency
3. Can bottleneck your internet gateway.
4. Security

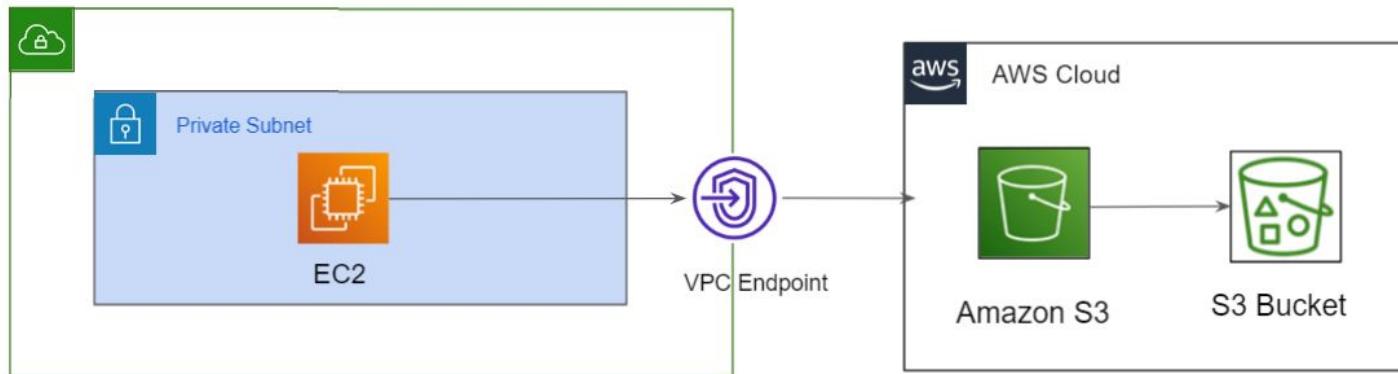
Overview of VPC Endpoints

VPC Endpoints allows us to connect VPC to another AWS services OR other supported services over AWS private network.



Overview of VPC Endpoints

VPC Endpoints allows us to connect VPC to another AWS services OR other supported services over AWS network.



Revising Important Pointers

AWS PrivateLink is a technology that enables you to privately access services by using private IP addresses.

To use AWS PrivateLink, you can create a VPC endpoint for a service in your VPC.

VPC Endpoint allows us to connect VPC to another AWS services over AWS network.

Traffic between your VPC and the other service does not leave the Amazon network.

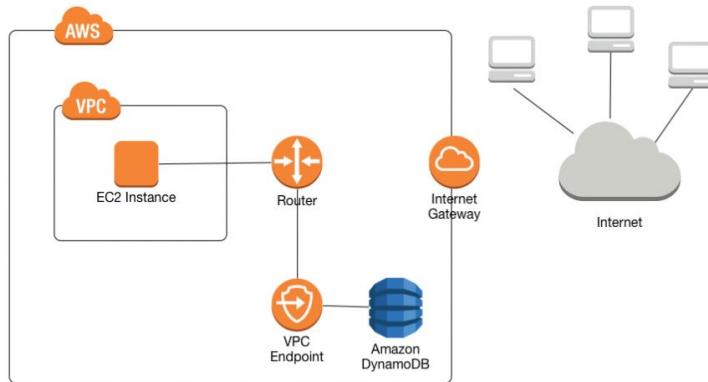
Interface Endpoints

New Generation Endpoint

Downsides of Gateway Endpoints - 1

In Gateway endpoints approach, the VPC endpoint was created outside your VPC and traffic was routed via route table.

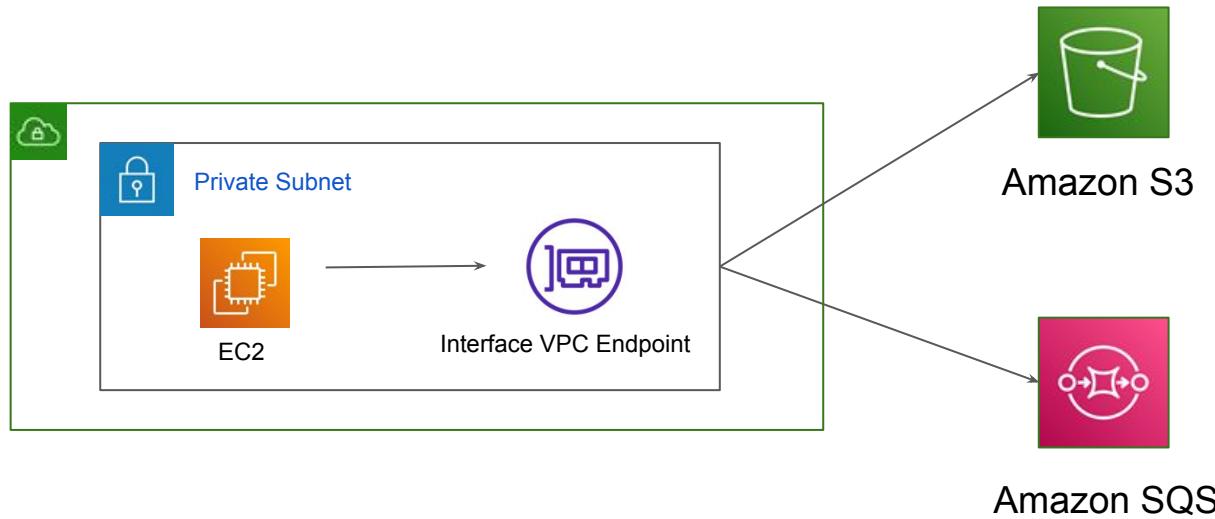
Thus, it is not possible to use it directly from VPN's or Direct connects and various others.



Interface Endpoints

An interface endpoint is an elastic network interface with a private IP address from the IP address range of your subnet.

It serves as an entry point for traffic destined to a supported AWS service or a VPC endpoint service.



Benefits of Interface Endpoint

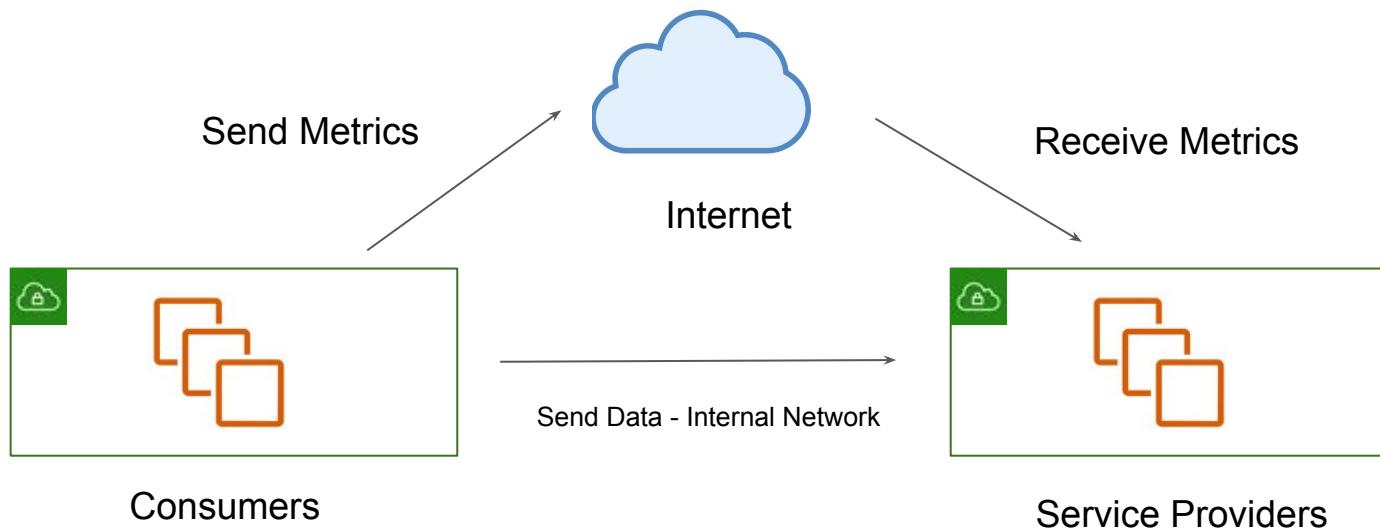
1. Interface endpoints enable the use of security groups to restrict access to the endpoint.
2. VPN's and Direct Connect based connections are supported.
3. Interface endpoints supports lot of services unlike Gateway endpoints.

VPC Endpoint Services

More Use-Cases Supported

Sample Use-Case

There are many service providers like DataDog, New Relic for which we need to upload our server/application metrics through Internet.



Dashboards using Metrics Collected

New Relic University > **SELECT** | Run New Relic University

Calagator Demo App Created by evose@newrelic.com Last edited 4/3/18

Search 2 attributes Add Dashboard Note Edit

Default 30m 60m 6h 1d 7d Custom

Calagator Demo App
AN OPEN SOURCE CALENDAR AGGREGATOR

This demo app is built using the open source calagator project.

[Calagator on Github](#)

Transaction Duration & Queing Since 60 minutes ago

NAME	TRANS...	Avg D...	Avg Q...
Controller/Middleware/Rack/ActionDispatch::Static/call	971	0.01	0.07
Controller/calagator/site/index	319	0.56	0.1
Controller/calagator/events/search	313	5.78	0.1
Controller/Middleware/Rack/ActionDispatch::Routing::RouteSet/call	187	0.21	0.12
Controller/calagator/sources/new	20	0.17	0.04

Frequent Transactions in the past 4 weeks Since 4 weeks ago

Transactions

631 K Controller/Middleware/Rack/
210 K Controller/calagator/site/inde
197 K Controller/calagator/events/s
120 K Controller/Middleware/Rack/
29.4 K Controller/Middleware/Rack/
13 K Controller/calagator/sources/
10.3 K Controller/calagator/sources/

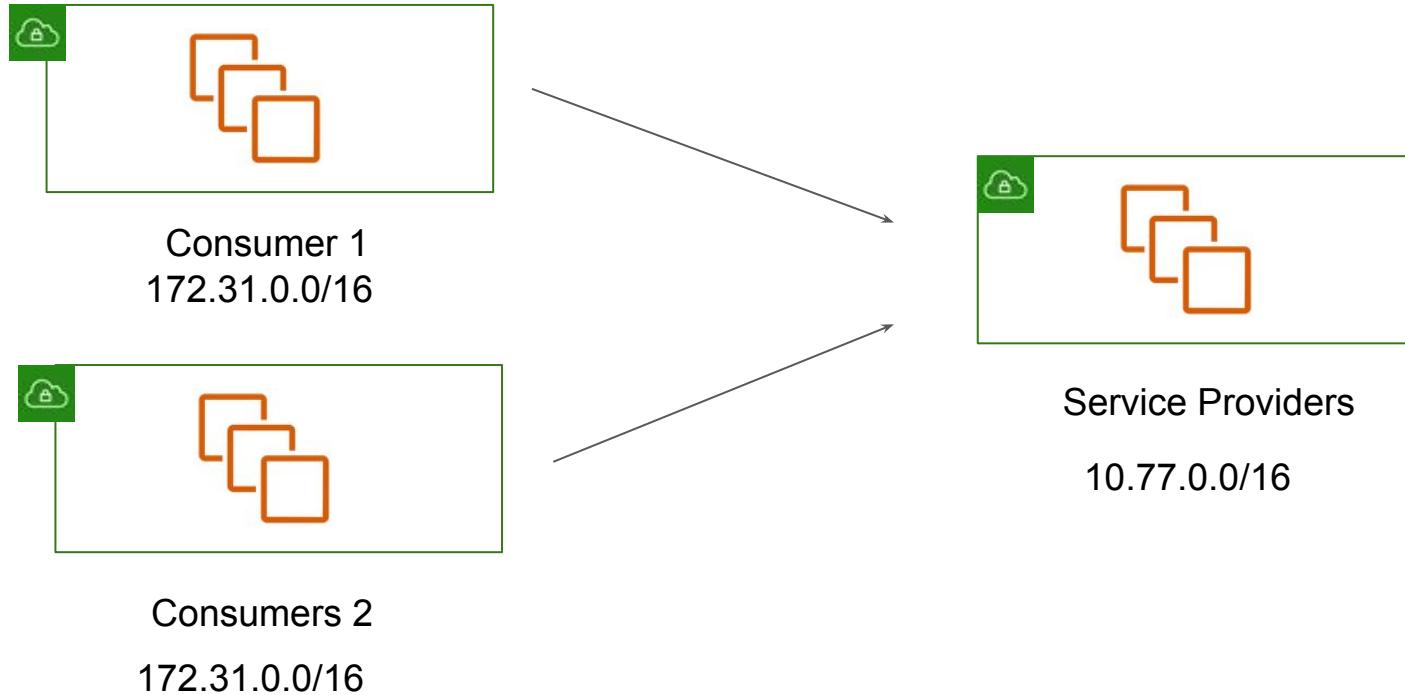
NRU Calagator Demo - Apdex Since 12 hours ago

Apdex Since 60 minutes ago

Transaction Duration Since 1 month ago

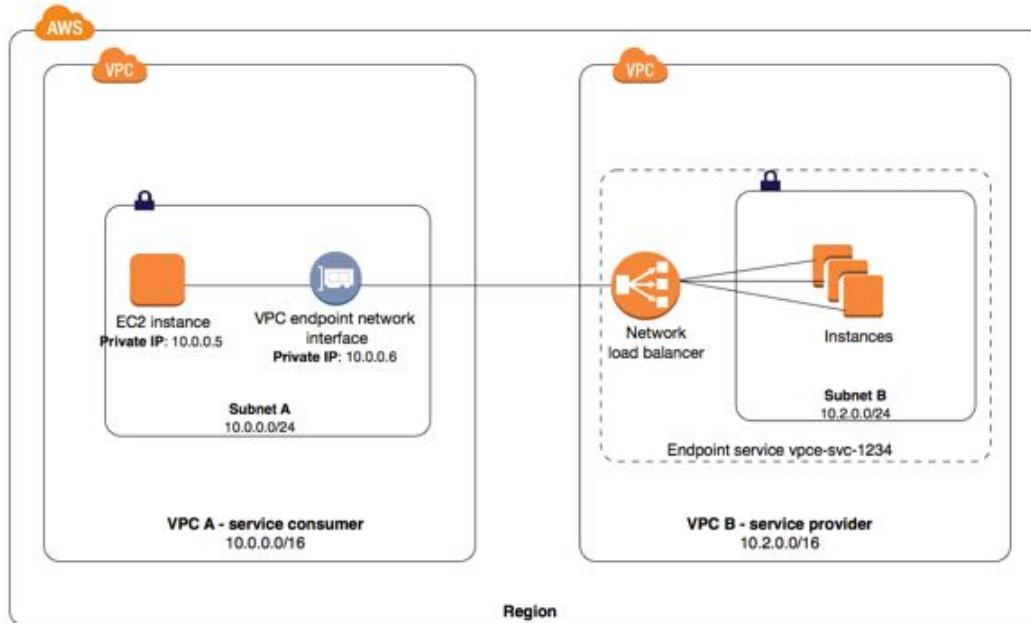
Possible Approach - VPC Peering

VPC Peering Approach will have multiple challenges related to CIDR overlap between clients.



Service VPC Endpoints

You can create your own application in your VPC and configure it as an AWS PrivateLink-powered service (referred to as an endpoint service)



EC2 Pricing

Cost Optimization

Paying for EC2 Instances

There are five primary ways in which we can pay for EC2 instance usage.

1. On-Demand
2. Savings Plan
3. Reserved Instances
4. Spot Instances
5. Dedicated Hosts

On-Demand Pricing

With On-demand instances, we pay for compute capacity per hour or per second depending on the instances which is being run.

No upfronts payments are needed and we can increase or decrease the capacity whenever it is needed.



On-Demand Can Lead to Unexpected Issues

Monday: 500 customers using 16GB RAM on-demand servers individually.

Wednesday: 30 customers using 16GB RAM on-demand servers individually.

A “Cloud Service Provider” will not have a clear picture on how many servers should the provision. Too high → resources might unused and too low → money loss



Reserved Instance

Reserved Instance provides us with significant discount (upto 75%) compared to on-demand instance pricing.

Reserved instance are assigned to a specific availability zone and provides capacity reservation for AWS EC2 instances.

Example :

You know you will always be running 20 servers of m4.2xlarge type of 1 year, then buy reserved instances for them.

Reserved Instance - Part 2

Example: g4dn.8xlarge instance type

Pricing Option	Hourly Cost	Total 3 year cost
On-Demand Instance	\$2.176	\$57,276
3 year all up-front - Reserved	-	\$21502
Savings		~62%

Spot Instance

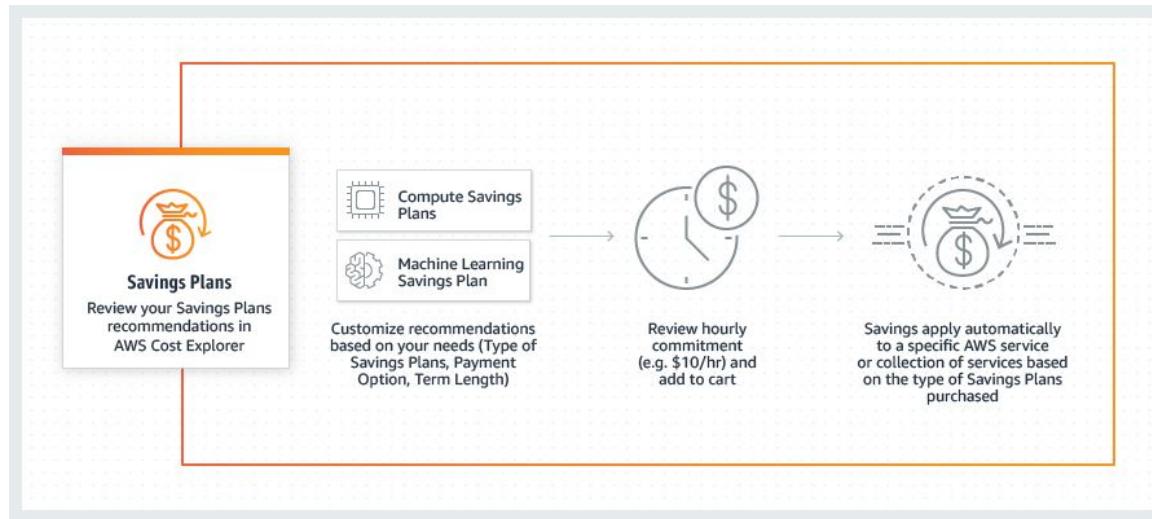
Spot instances allows us to bid on spare Amazon EC2 computing capacity for up to 90% of the on-demand cost.

Such instances are recommended for applications that can have flexible start and end times



Savings Plans

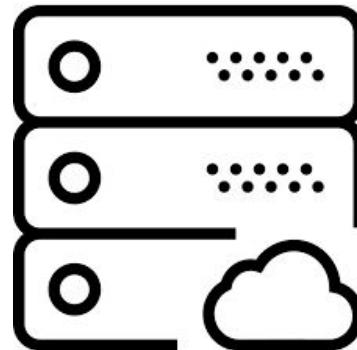
Savings Plans are a flexible pricing model that offer low prices on EC2 and Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in \$/hour) for a 1 or 3 year term.



Dedicated Host

A dedicated host is a physical EC2 server dedicated for your use.

It can be purchased on-demand as well as reserved instance.



Reserved Instances

Money Optimization

Types of Reserved Instances

Type of RIs	Description
Standard RIs	These provide the most significant discount (up to 72% off On-Demand) and are best suited for steady-state usage.
Convertible RIs	These provide a discount (up to 54% off On-Demand) and the capability to change the attributes of the RI
Scheduled RIs	These are available to launch within the time windows you reserve.

RI Types

With convertible RI, we can :

- Convert to new instance family eg R3 to M4 to C4 to T2
- Convert to new operating system eg Windows to Linux
- Convert to new instance price [eg if AWS reduces the public rate for our instance]
- Convert to new instance size [eg : from m4.xlarge to m4.2xlarge]
- Convert tenancy [eg dedicated instance to default]
- Convert to different payment option [no upfront to partial upfront]

Reservation Term

Reservation Term	Description
No Upfront	No upfront required. Lower discount rate compared to others.
Partial Upfront	You make a low upfront payment and are then charged a discounted hourly rate for the instance for the duration of the Reserved Instance term.
All Upfront	You pay for the entire Reserved Instance term with one upfront payment. Provides the largest discount.

Regional vs Zonal RIs

	Regional RI	Zonal RI
Ability to Reserve Capacity	No Reservation in Capacity.	Capacity Reserved in the specific Availability Zone.
Availability Zone Flexibility	The Reserved Instance discount applies to instance usage in any Availability Zone in the specified Region.	Reserved Instance discount applies to instance usage in the specified Availability Zone only.
Instance size flexibility	The Reserved Instance discount applies to instance usage within the instance family, regardless of size.	No instance size flexibility—the Reserved Instance discount applies to instance usage for the specified instance type and size only.

Scenario

Scenario 1 :

Customer has following instances running:

- 2 x m4.large instance running in us-east-1a and us-east-1b region.
- 2 x t2.large instance running across us-east-1b and us-east-1c region

Customer has following RI:

- 2 x m4.large, default tenancy, us-east-1b region (zonal RI)
- 2 x t2.large, default tenancy, us-east-1 regional RI

Additional pay : 1 x m4.large instance will be charged at the on-demand rate.

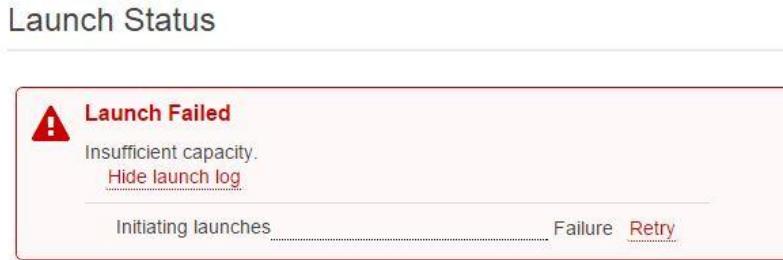
On-Demand Capacity Reservations

Reserving EC2 Capacity

Understanding the Challenge

With On-Demand Instance, there are chances that new instance might not launch due to insufficient capacity error.

Typical Solution → Go with Reserved Instances (1 year, 3 year term)



Benefits of On-Demand Capacity Reservation

On-Demand Capacity Reservations enable you to reserve compute capacity for your Amazon EC2 instances in a specific Availability Zone for any duration

You can create Capacity Reservations at any time, without entering into a one-year or three-year term commitment, and the capacity is available immediately.

Reservation details

Reservation ends
Ending your reservation releases held capacity and thus prevents additional instances from being launched against it. Any launched instances continue to run and accrue applicable instance usage charges. You can view and manage those instances, if any, from the Launch Reservations view.

Manually
I will cancel my reservation when I am finished.

Specific time
Prevent launching instances against this reservation.

2021/06/12  20:28

Instance eligibility
Indicate the criteria for instances that can fulfill this reservation.

Any instance with matching details
Instance type, platform, and Availability Zone must match what is specified in this reservation.

Only instances that specify this reservation
When you launch instances, you must specify the reservation ID or the reservation resource group ARN associated with this reservation.

On-Demand Capacity Reservation with RI

You can combine Capacity Reservations with Regional Reserved Instances to receive a discount.

	On-Demand Capacity Reservation	Regional RI
Term	No commitment required. Can be created and canceled as needed.	Requires a fixed one-year or three-year commitment
Capacity benefit	Capacity reserved in a specific Availability Zone.	No capacity reserved.
Billing discount	No billing discount.	Provides a billing discount.

Pricing

When the Capacity Reservation enters the active state, you are charged the equivalent On-Demand rate whether you run instances in the reserved capacity or not.

If you do not use the reservation, this shows up as unused reservation on your EC2 bill.

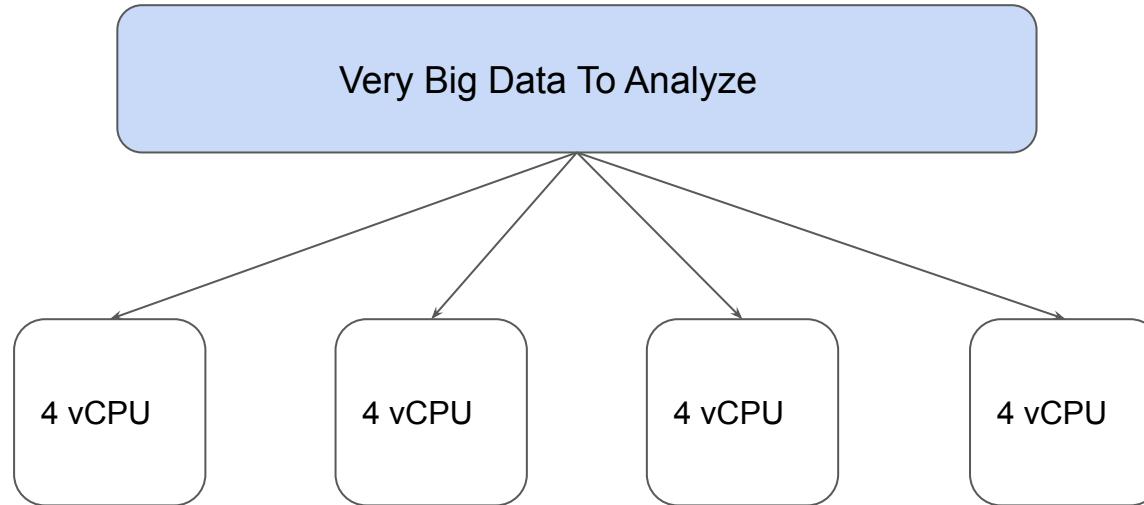
For example, if you create a Capacity Reservation for 20 m4.large Linux instances and run 15 m4.large Linux instances in the same Availability Zone, you will be charged for 15 active instances and for 5 unused instances in the reservation.

EC2 Fleet

Launching EC2 based on Requirements

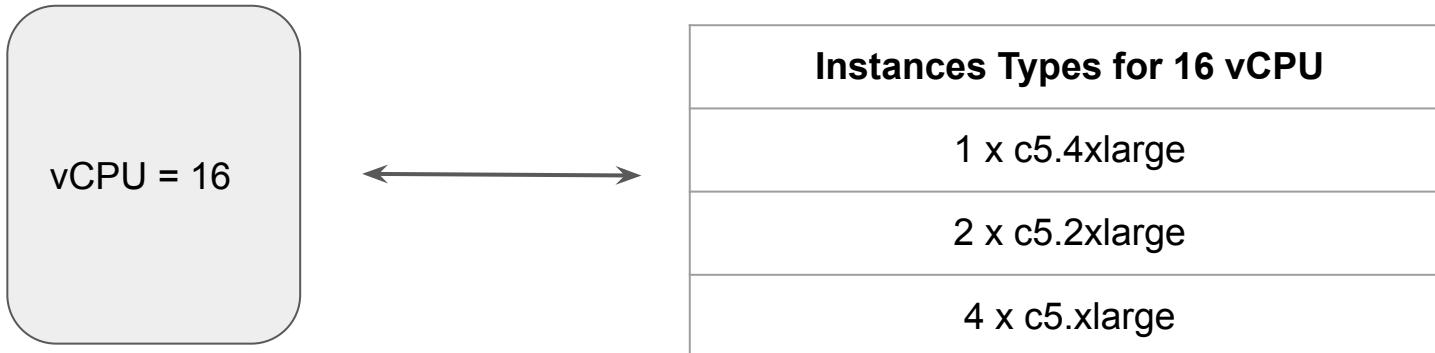
Original Feature

EC2 Fleet allows users to launch a fleet of Spot Instances that spans EC2 instance types and Availability Zones without having to write custom code to discover capacity or monitor prices.



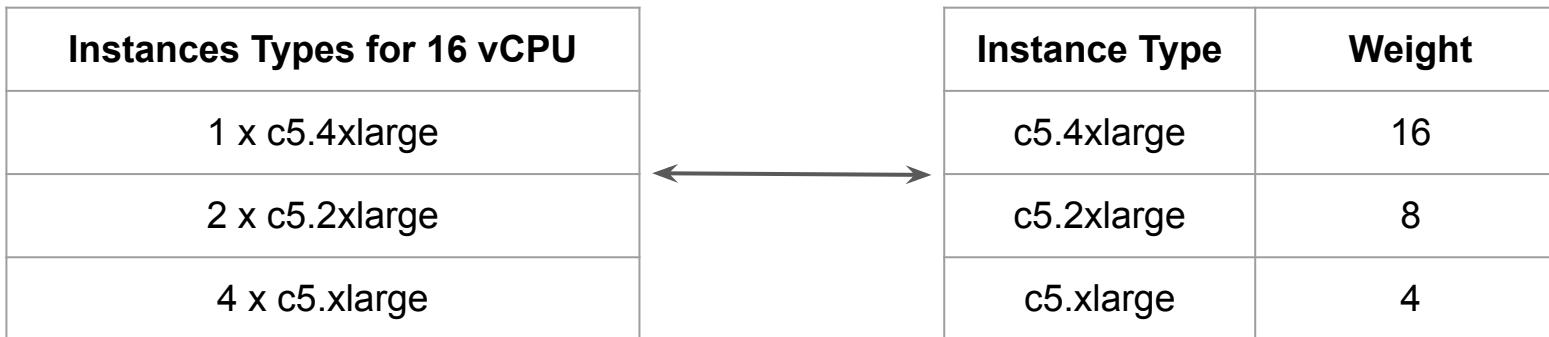
Understanding With A Use-Case

Let us assume, to perform the analysis, you require in total of 16 vCPU.



Setting Weight

You can assign a set of weighted capacity to a set of EC2 instance Types.



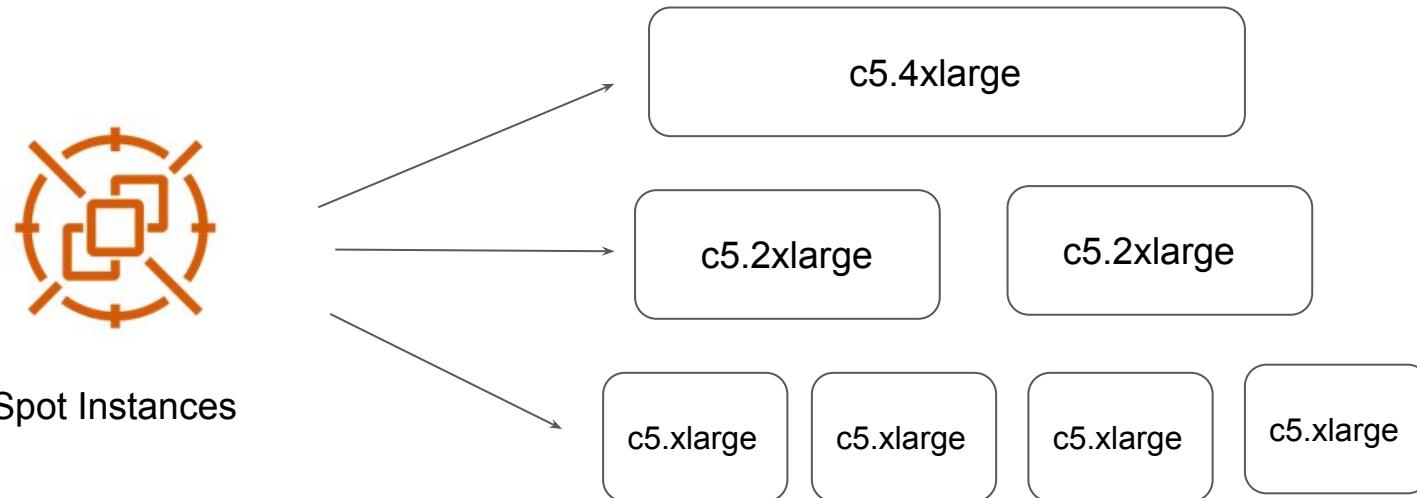
Setting Weight

```
"Overrides": [  
  {  
    "InstanceType": "c5.4xlarge",  
    "WeightedCapacity": 16,  
  },  
  {  
    "InstanceType": "c5.2xlarge",  
    "WeightedCapacity": 8,  
  },  
  {  
    "InstanceType": "c5.xlarge",  
    "WeightedCapacity": 4,  
  },  
]
```

Target Capacity = 16

Setting Target Capacity

EC2 Fleet will select the most cost effective combination of instance types and Availability Zones (both specified in the template) using the current prices for the Spot Instances and public prices for the On-Demand Instances



Overview of EC2 Fleet - New

The EC2 Fleet attempts to launch the number of instances that are required to meet the target capacity that you specify in the fleet request.

The fleet can comprise only On-Demand Instances, only Spot Instances, or a combination of both On-Demand Instances and Spot Instances.

```
"TargetCapacitySpecification": {  
    "TotalTargetCapacity": 16,  
    "OnDemandTargetCapacity": 4,  
    "SpotTargetCapacity": 8,  
    "DefaultTargetCapacityType": "Spot"  
}
```

Interpret the Requirements

I want total of 16 vCPUs.

8 vCPU should be fulfilled using On-Demand instance type.

8 vCPU should be fulfilled using Spot instance type.

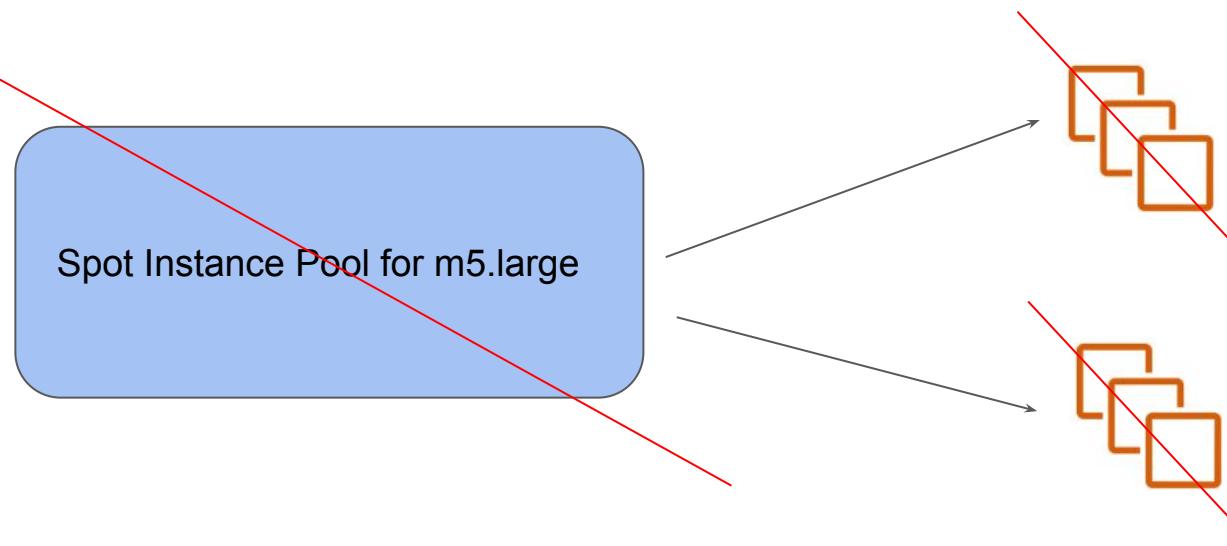
```
"TargetCapacitySpecification": {  
    "TotalTargetCapacity": 16,  
    "OnDemandTargetCapacity": 8,  
    "SpotTargetCapacity": 8,  
    "DefaultTargetCapacityType": "Spot"  
}
```

Allocation Strategy for Spot Instances

Optimizing Costs

Understanding Spot Instance Pool

A Spot instance pool is a set of unused EC2 instances with the same instance type and size (for example, m5.large), availability zone (AZ), in the same region



Allocation Strategy

Depending on your requirements and use-case, there are three primary allocation strategy for Spot instances.

Allocation Strategy	Description
lowest-price	The Spot Instances come from the Spot capacity pool with the lowest price. This is the default strategy.
diversified	The Spot Instances are distributed across all Spot capacity pools.
capacity-optimized	Provisions Spot Instances from the most-available Spot Instance pools by analyzing capacity metrics.

Preference - lowest-price

Choose the lowest-price allocation strategy if:

If your fleet is small or runs for a short time, the probability that your Spot Instances will be interrupted is low, even with all of the instances in a single Spot capacity pool.

Therefore, the lowest-price strategy is likely to meet your needs while providing the lowest cost.

Since the price constantly changes, the existing instances in ASG can be terminated and be replaced by new, cheaper ones thus potentially disrupting your service at a higher rate.

Preference - diversified

If your fleet is large or runs for a long time, you can improve the availability of your fleet by distributing the Spot Instances across multiple pools using the diversified strategy.

For example, if your EC2 Fleet specifies 10 pools and a target capacity of 100 instances, the fleet launches 10 Spot Instances in each pool.

If the Spot price for one pool exceeds your maximum price for this pool, only 10% of your fleet is affected.

Preference - capacity-optimized

If your fleet runs workloads that may have a higher cost of interruption associated with restarting work and checkpointing, then use the capacity-optimized strategy

This strategy does not look at the prices of the instance types in each pool configure but instead looks for the optimal capacity volume and chooses those instances to run your service on.

While the overall hourly cost of capacity-optimized allocation strategy might be slightly higher, the possibility of having fewer interruptions can lower the overall cost of your workload.

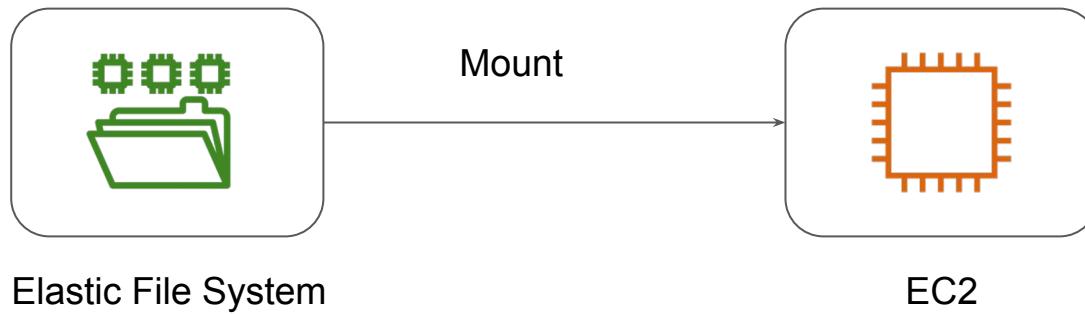
Amazon Elastic File System (EFS)

Network Attached Storage

Overview of Elastic File System

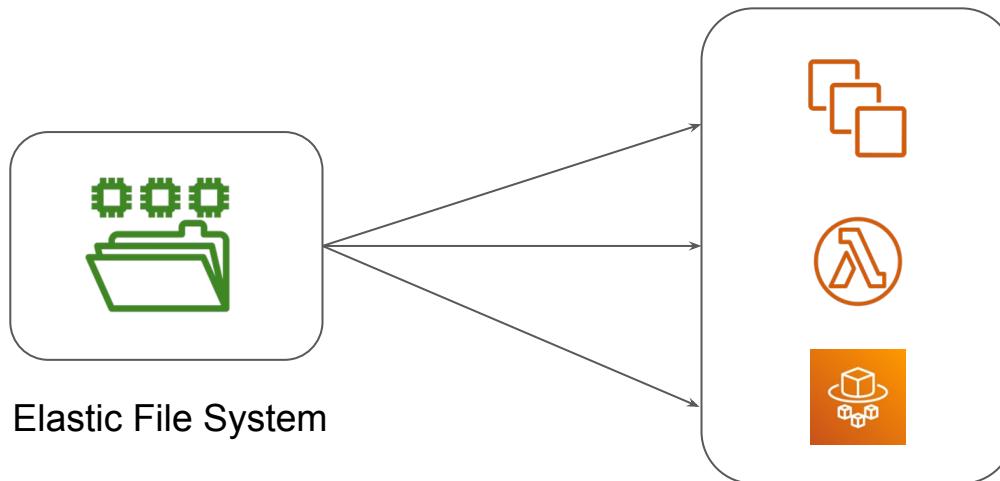
Amazon Elastic File System (Amazon EFS) provides a simple, serverless, set-and-forget elastic file system for use with AWS Cloud services and on-premises resources.

It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files



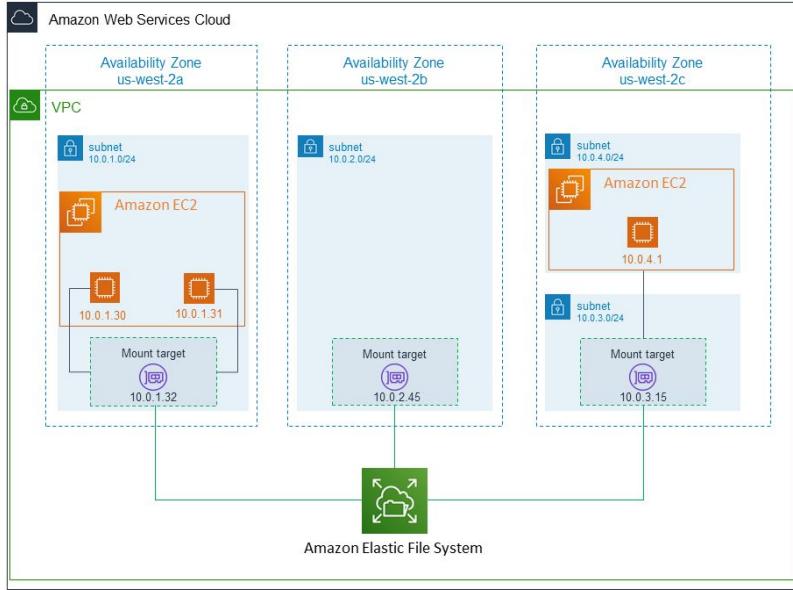
Attachment to Multiple Targets

Multiple compute instances, including Amazon EC2, Amazon ECS, and AWS Lambda, can access an Amazon EFS file system at the same time, providing a common data source for workloads.



Understanding EFS Architecture

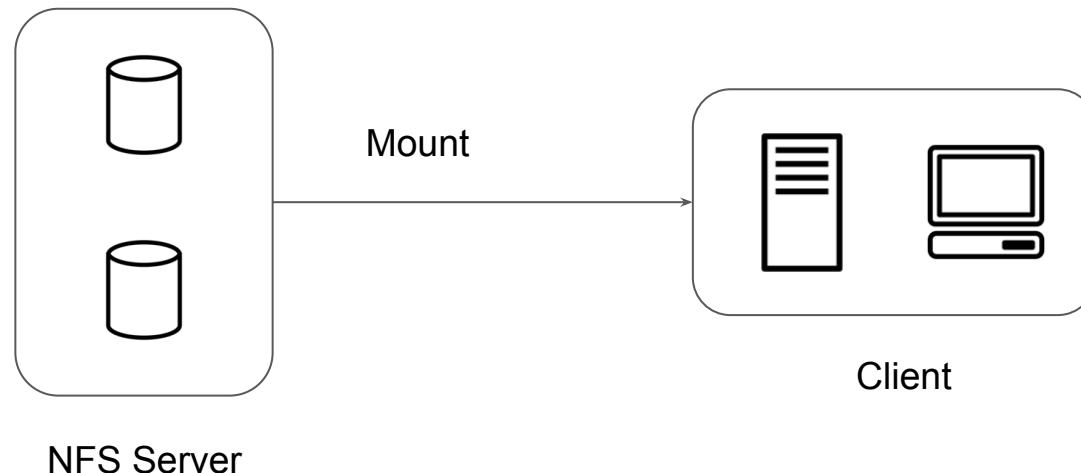
To access file system from instance inside the VPC, we need to create mount target in the VPC.



Network File System

Network File System (NFS) is a networking protocol for distributed file sharing.

EFS uses the Network File System version 4 (NFS v4) protocol



Pricing Considerations

AWS EFS is expensive when compared to other storage options like EBS, S3.

Consideration	Pricing
1 TB EFS with 80% frequently accessed data	\$250
1TB EBS Storage	\$102
1 TB of S3 Storage	\$24

Important Pointers

If performance is your concern, always prefer EBS.

EFS can even be accessed from on-premise datacenter using an AWS Direct Connect or AWS VPN connection.

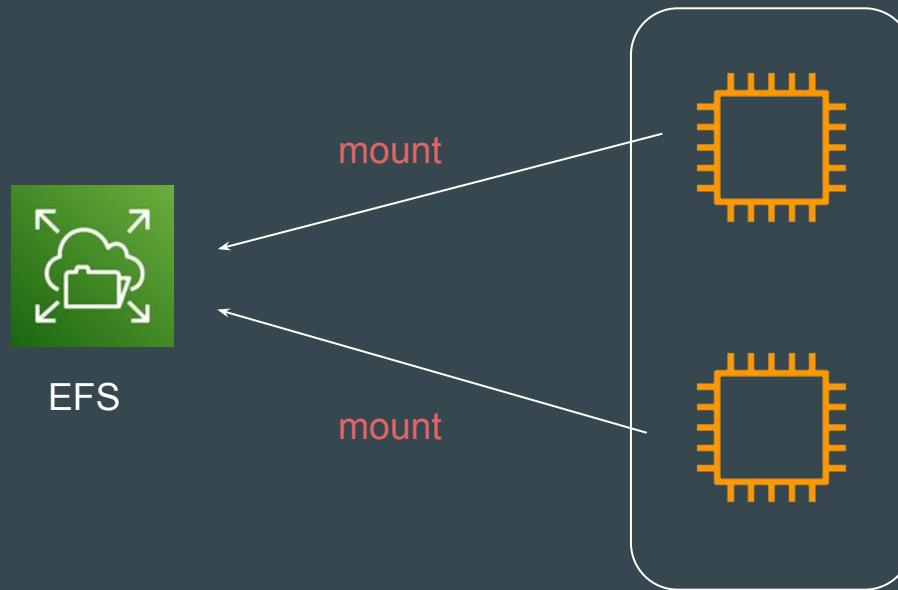
With Amazon EFS, you pay only for what you use per month.

EFS - File System Policies



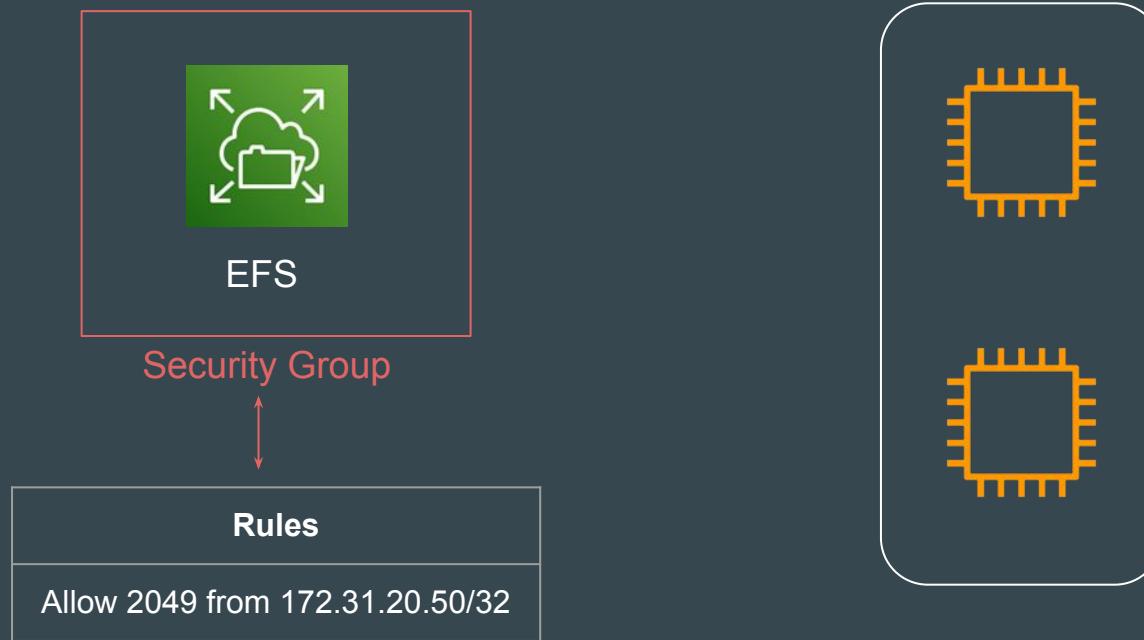
Understanding the Challenge

When you create EFS volume, by default, any EC2 instance will be able to mount it provided sufficient network connectivity is present (no authentication)



Implementing Restriction - Traditional Way

Traditionally when EFS volume was launched, the primary way to restrict access to EFS volume was through security groups.



EFS File System Policy

EFS File System policy is a **resource based policy** that allows granular control on the capabilities and accessibility at a EFS level.



File System Policy
Enforce Read-Only Access By Default
Prevent Anonymous Access
Enforce In-Transit Encryption

EFS File System Policy

File system policy

Policy options

Select one or more of these common policy options, or create a custom policy using the editor. | [Learn more](#)

- Prevent root access by default*
- Enforce read-only access by default*
- Prevent anonymous access
- Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

► [Grant additional permissions](#)

Policy editor [JSON]

Clear

```
1 ~ [  "Version": "2012-10-17",  
2     "Id": "efs-policy-wizard-2dd2103c-2c06-4fb9-886d-704522197902",  
3     "Statement": [  
4         {  
5             "Sid": "efs-statement-f3d5c694-e145-4096-8a0d-c6070f5d5f86",  
6             "Effect": "Allow",  
7             "Principal": {  
8                 "AWS": "*"  
9             },  
10            "Action": [  
11                "elasticfilesystem:ClientWrite",  
12                "elasticfilesystem:ClientMount"  
13            ],  
14            "Condition": {  
15                "Bool": {  
16                    "elasticfilesystem:AccessedViaMountTarget": "true"  
17                }  
18            }  
19        }  
20    ]  
21 ]  
22 ]
```

Manual changes will prevent the use of the policy options on the left until the editor is cleared.

Cancel

Save

Grant read and write access to a specific AWS role

```
{  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "arn:aws:iam::111122223333:role/Testing_Role"  
            },  
            "Action": [  
                "elasticfilesystem:ClientWrite",  
                "elasticfilesystem:ClientMount"  
            ],  
            "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-system/fs-1234abcd",  
            "Condition": {  
                "Bool": {  
                    "elasticfilesystem:AccessedViaMountTarget": "true"  
                }  
            }  
        }  
    ]  
}
```

Policy Example - Grant read-only access to IAM Role

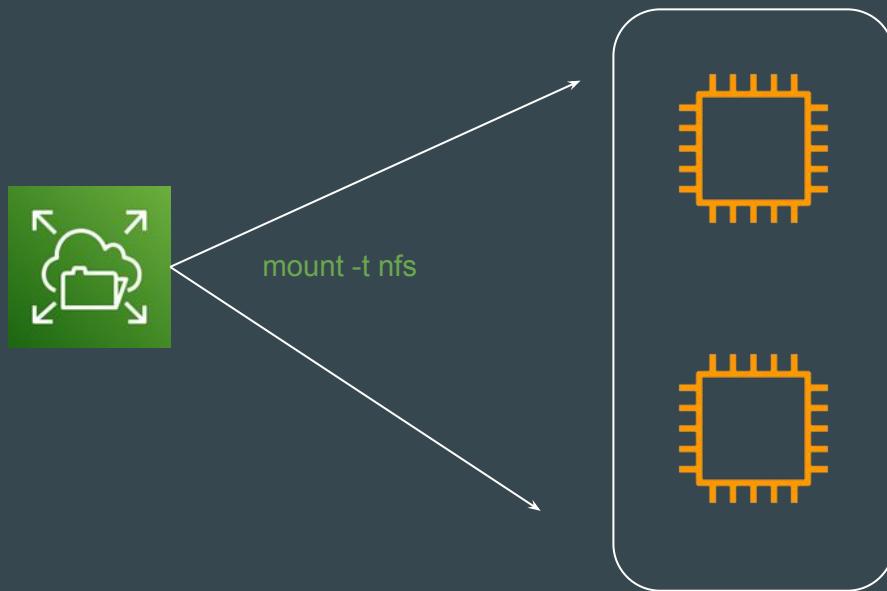
```
{  
    "Id": "read-only-example-policy02",  
    "Statement": [  
        {  
            "Sid": "efs-statement-example02",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "arn:aws:iam::111122223333:role/EfsReadOnly"  
            },  
            "Action": [  
                "elasticfilesystem:ClientMount"  
            ],  
            "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-system/fs-12345678"  
        }  
    ]  
}
```

EFS - Access Points



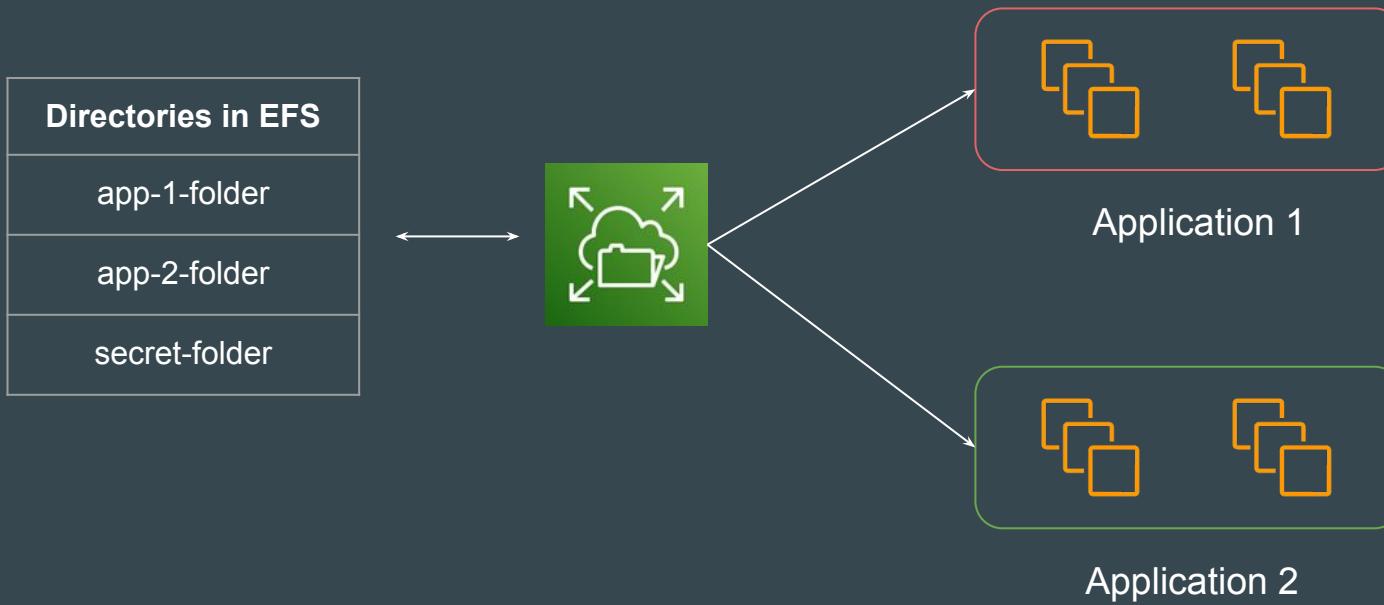
Setting the Base

When a EFS is mounted on EC2 instance, by default the root of the file system is made available to the EC2.



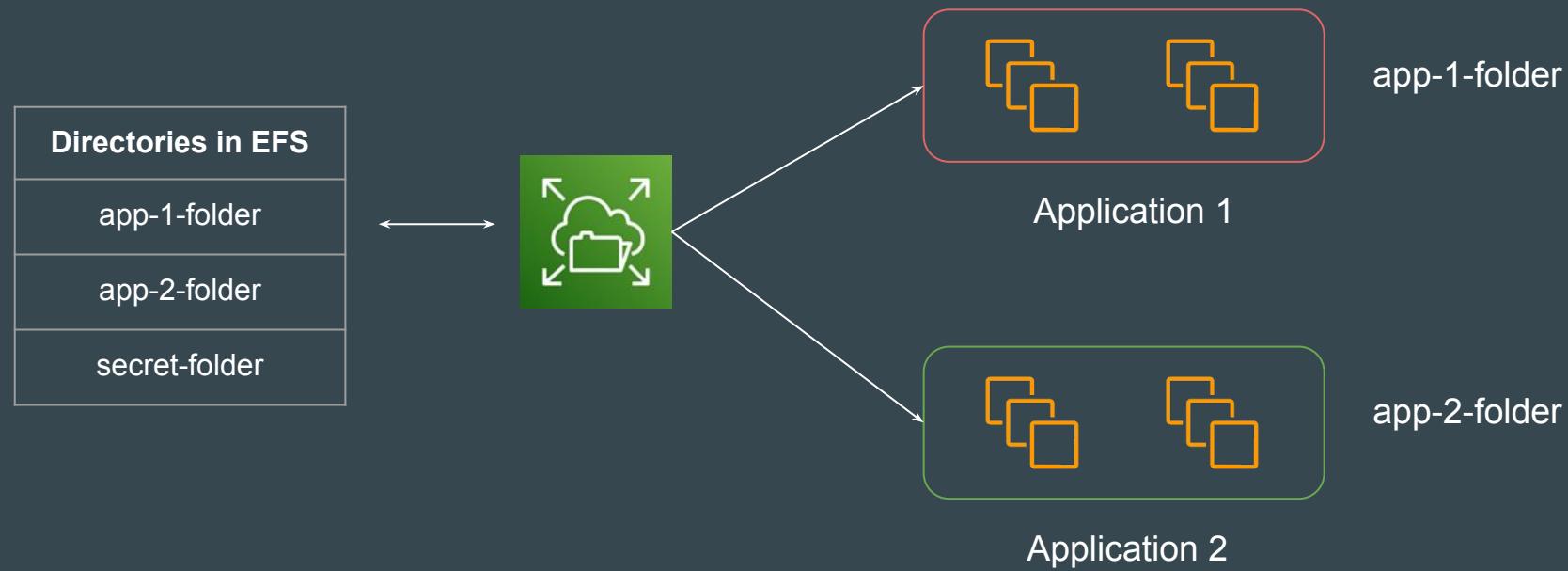
Understanding the Challenge

If EFS has multiple set of directories for different application, you do not want the ROOT of the file system available to all the clients.



EFS Access Points

Amazon EFS access points are **application-specific entry points** into an EFS file system that make it easier to manage application access to shared datasets



Cross Account EFS Access



Setting the Base

There can be a requirement where resources in Account B wants to access the EFS running in Account A.



EFS

Account A



Account B

Important Requirement - 1

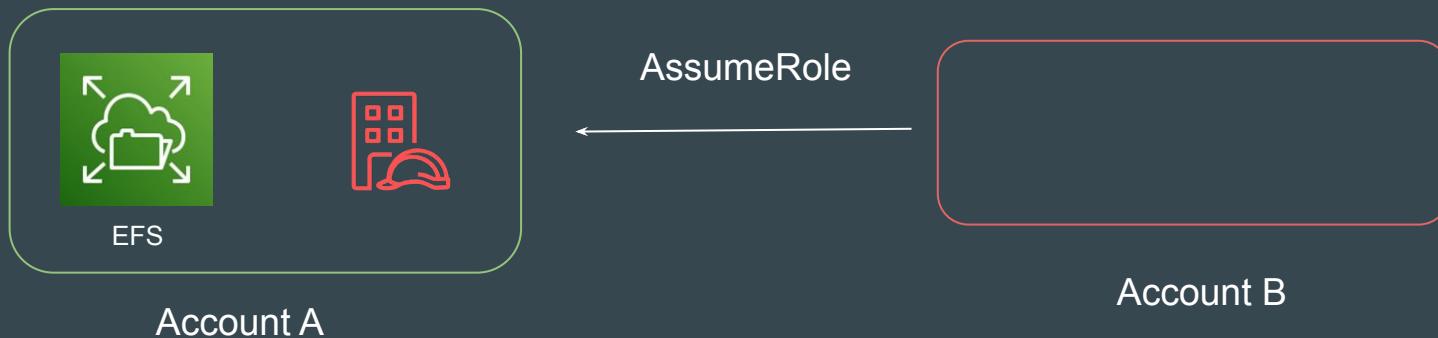
The VPCs of your NFS client and your EFS file system must be connected using either a VPC peering connection or a VPC Transit Gateway.

This allows Amazon Elastic Compute Cloud (Amazon EC2) instances from the same or different accounts, to access EFS file systems in a different VPC.



Important Requirement - 2

Create a cross account IAM Role in Account A that allows sts:AssumeRole action from Account B resources.



Important Requirement - 3

Create IAM Role in Account-B to allow the resources to be able to assume the Account-A IAM Role and gain necessary permissions.



Important Requirement - 4

Modify the EFS File system policy to allow mounts from Account-B hosting the resources.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Principal": {
                "AWS": "arn:aws:iam::<aws-account-id-B>:root"
            }
        }
    ]
}
```

Points to Note

You can mount your Amazon EFS file system by using IAM authorization for NFS clients and access points with the Amazon EFS mount helper.

By default, the mount helper uses DNS to resolve the IP address of your mount target.

So if you're mounting from another account or Amazon Virtual Private Cloud (Amazon VPC), you must resolve the Amazon EFS mount target IP manually.

AWS Health



Setting the Base

AWS Health events are notifications that AWS Health sends on behalf of other AWS services.

You can use these events to learn about upcoming or scheduled changes that might affect your account.

For example, AWS Health can send an event if AWS Identity and Access Management (IAM) plans to deprecate a managed policy or AWS Config plans to deprecate a managed rule.

Reference Screenshot

Event log								
Event		Status	Event category	Region / Zone	Start time	Last update time	Affected resources	Actions
Operational issue - WAF (Global)	Closed	Issue	-		May 27, 2023 at 2:57:00 AM UTC+5:30	May 27, 2023 at 4:14:04 AM UTC+5:30	-	< 1 >
Operational issue - Billing (Global)	Closed	Issue	-		April 16, 2023 at 8:11:48 PM UTC+5:30	April 16, 2023 at 11:19:30 PM UTC+5:30	-	
Operational issue - SNS (Ireland)	Closed	Issue	eu-west-1		April 6, 2023 at 9:50:00 PM UTC+5:30	April 6, 2023 at 10:11:56 PM UTC+5:30	-	
Operational issue - Health (Global)	Closed	Issue	-		April 6, 2023 at 1:09:00 AM UTC+5:30	April 6, 2023 at 2:36:38 AM UTC+5:30	-	
Operational issue - Quicksight (Mumbai)	Closed	Issue	ap-south-1		April 5, 2023 at 4:46:29 PM UTC+5:30	April 5, 2023 at 9:12:28 PM UTC+5:30	-	
Operational issue - CloudTrail (Mumbai)	Closed	Issue	ap-south-1		April 1, 2023 at 8:13:35 AM UTC+5:30	April 1, 2023 at 9:19:10 AM UTC+5:30	-	
Operational issue - CloudFormation (Oregon)	Closed	Issue	us-west-2		March 29, 2023 at 3:19:06 AM UTC+5:30	March 29, 2023 at 5:27:52 AM UTC+5:30	-	
Operational issue - InternetConnectivity (Oregon)	Closed	Issue	us-west-2		March 13, 2023 at 10:27:31 PM UTC+5:30	March 13, 2023 at 10:52:27 PM UTC+5:30	-	

AWS Health Events

Health Event Types	Description
Account-specific event	<p>Account-specific events are local to either your AWS account or an account in your AWS organization.</p> <p>For example, if there's an issue with an Amazon EC2 instance type in a Region that you use, AWS Health provides information about the event and the name of the affected resources.</p>
Public event	<p>Public events are reported service events that aren't specific to an account</p> <p>For example, if there's a service issue for Amazon S3 in the US East (Ohio) Region, AWS Health provides information about the event, even if you don't use that service or have S3 buckets in that Region.</p>

Monitoring Health Events

You can use Amazon EventBridge to detect and react to AWS Health events.

Then, based on rules that you create, EventBridge invokes one or more target actions when an event matches the values that you specify in a rule.

For example, you can use a Lambda function to pass a notification to a Slack channel when an AWS Health event occurs.



Reference Screenshot

Event pattern [Info](#)

Event source
AWS service or EventBridge partner as source
▼

AWS service
The name of the AWS service as the event source
▼

Event type
The type of events as the source of the matching pattern
▼

ⓘ This builder helps to build an event pattern to get events from AWS Health regarding health status of other AWS services.

Any service
 Specific service(s)
▼

Any event type category
 Specific event type category(s)
▼

Any event type code
 Specific event type code(s)
 X

Any resource
 Specific resource(s)

Event pattern
Event pattern, or filter to match the events

```
1 {
2   "source": ["aws.health"],
3   "detail-type": ["AWS Health Event"],
4   "detail": {
5     "service": ["EC2"],
6     "eventTypeCategory": ["issue"],
7     "eventTypeCode": ["AWS_EC2_OPERATIONAL_ISSUE"]
8   }
9 }
```

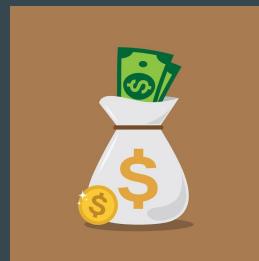
AWS Compute Optimizer



Understanding the Challenge

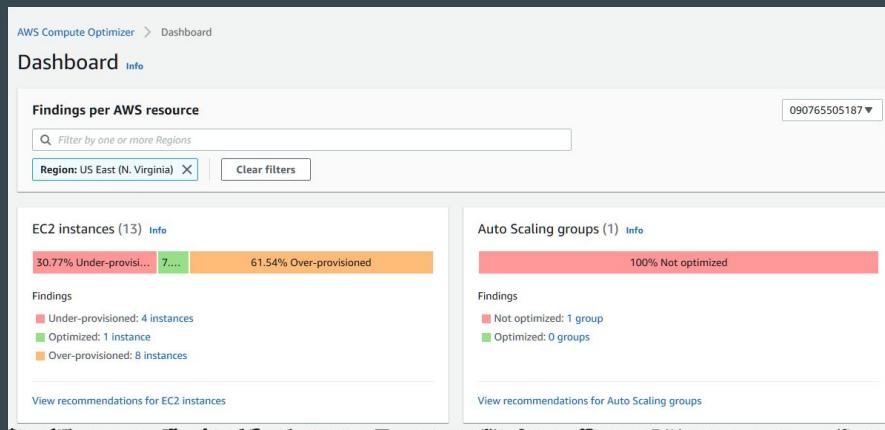
In most of the organizations, over provisioning is a big challenge.

This leads to large cost at the end of the month.



AWS Compute Optimizer

AWS Compute Optimizer **recommends optimal AWS resources** for your workloads to reduce costs and improve performance by using machine learning to analyze historical utilization metrics



Recommendations

AWS Computer Optimizer also provides the recommendations related to the optimal instance types.

AWS Compute Optimizer > Dashboard > Recommendations for EC2 instances

Recommendations for EC2 instances (8) Info
Recommendations for modifying current resources for better cost and performance.

Action ▾ View detail

Filter by one or more Regions: 090765505187 | Over-provisioned | < 1 > | ⌂

Region: US East (N. Virginia) | Clear filters

Instance ID	Instance name	Finding	Current instance type	Current On-Demand price	Recommended instance type	Recommended On-Demand price
i-0fb9323080785de1e	-	Over-provisioned	c5.xlarge	\$0.17 per hour	t3.large	\$0.0832 per hour
i-0f4f4c06ad8afe81a	-	Over-provisioned	m5.2xlarge	\$0.384 per hour	r5.xlarge	\$0.252 per hour
i-0f277818dffef522e9	-	Over-provisioned	c5.xlarge	\$0.17 per hour	t3.large	\$0.0832 per hour
i-0ceb95ed248026d24	-	Over-provisioned	m5.xlarge	\$0.192 per hour	r5.large	\$0.126 per hour
i-0af9322ff627d7e8f	-	Over-provisioned	m5.xlarge	\$0.192 per hour	r5.large	\$0.126 per hour
i-07084b94d1bcf391b	-	Over-provisioned	c5.xlarge	\$0.17 per hour	t3.large	\$0.0832 per hour
i-069f6e837890db127	-	Over-provisioned	c5.xlarge	\$0.17 per hour	t3.large	\$0.0832 per hour
i-0218a45abd8b53658	-	Over-provisioned	m5.xlarge	\$0.192 per hour	r5.large	\$0.126 per hour

Supported Resource Types

AWS Compute Optimizer delivers recommendations for selected types of:

- EC2 instances,
- EC2 Auto Scaling groups
- EBS volumes
- Amazon ECS services on AWS Fargate
- Lambda functions.

Points to Note

AWS Compute Optimizer uses [Amazon CloudWatch metrics](#) as basis for the recommendations.

By default, CloudWatch metrics are the ones it can observe from an hypervisor point of view, such as CPU utilization, disk IO, and network IO.

If you want AWS Compute Optimizer to take into account operating system level metrics, such as memory usage, you need to install a CloudWatch agent on your EC2 instance

Enhanced Infrastructure Metrics

Enhanced infrastructure metrics is **a paid feature** of Compute Optimizer that applies to Amazon EC2 instances.

Extends the utilization metrics analysis look-back period to up to three months (93 days), compared to the 14-day (2-week) period. This gives Compute Optimizer a longer history of utilization metrics data to analyze.

Recommendation preferences

Recommendation preferences augment the capabilities of Compute Optimizer to generate enhanced recommendations.

Enhanced infrastructure metrics - *paid feature* | [Info](#)

By default, Compute Optimizer stores and uses up to 14 days of your CloudWatch metrics history to generate your recommendations. After you activate enhanced infrastructure metrics, history.

 Inactive

Export Recommendations

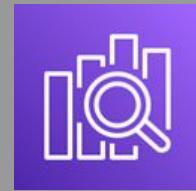
You can export your recommendations to record them over time, and share the data with others.

Recommendations are exported in a CSV file, and its metadata in a JSON file, to an existing Amazon Simple Storage Service (Amazon S3) bucket that you specify.

The screenshot shows the 'Exports' section of the AWS Lambda console. At the top, there is a search bar labeled 'Search by job ID or keyword' and dropdown menus for 'All resource types' and 'All statuses'. Below this is a table header with columns: 'Created', 'Resource type', 'Region info', 'Status info', 'Export destination Info', 'Object prefix Info', 'S3 bucket Info', 'Job ID', and 'Failure reason'. A single row of data is listed:

Created	Resource type	Region info	Status info	Export destination Info	Object prefix Info	S3 bucket Info	Job ID	Failure reason
02/15/2023, 11:42:45	EC2 instances	Asia Pacific (Singapore)	Completed	https://s3.console.aws.amazon.com/s3/object/kplab-s-sample-asset-bucket/compute-optimizer/04202557788/ap-southeast-1-2023-02-15T061245Z-563fe81a-b985-4a41-aeec-b9b526fc17b2.csv	-	kplabs-sample-asset-bucket	563fe81a-b985-4a41-aeec-b9b526fc17b2	-

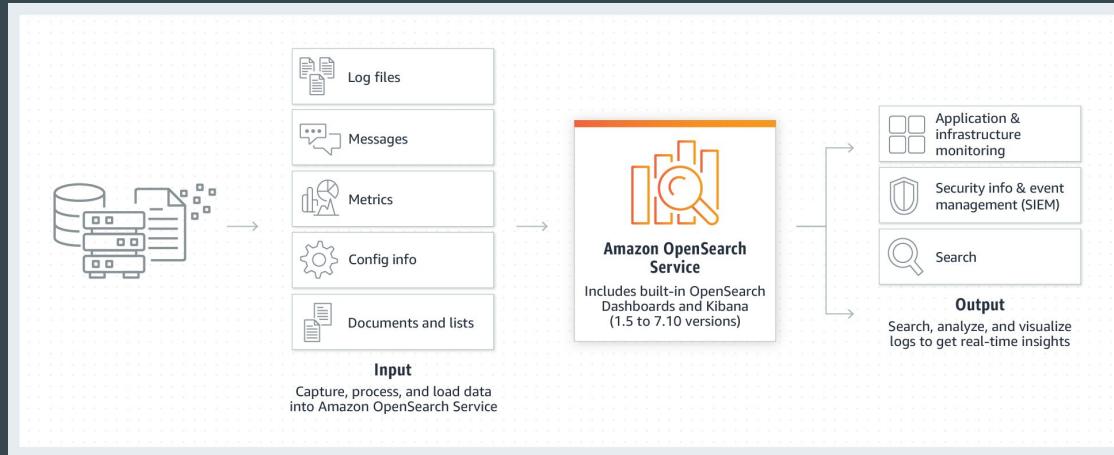
Amazon OpenSearch



Understanding the Basics

Amazon OpenSearch is initially based on the forked version of ElasticSearch

Allow ingesting, searching and visualization of data.



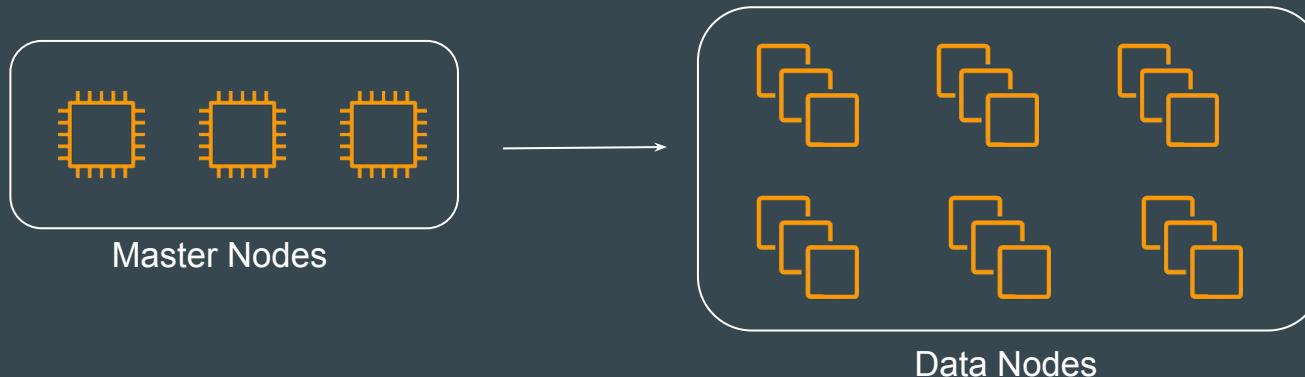
Storage Options - OpenSearch



Basics of Nodes

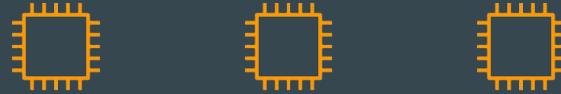
Amazon OpenSearch Service uses dedicated master nodes to increase cluster stability.

A **dedicated master node** performs cluster management tasks, but does not hold data or respond to data upload requests

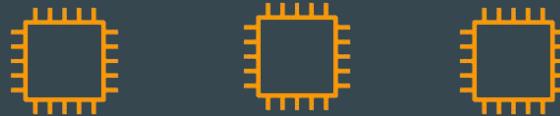


Storage Options

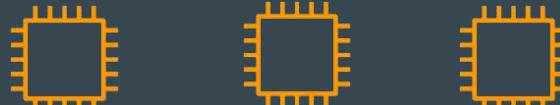
Master Nodes



“Hot” Data Nodes



UltraWarm Data Nodes



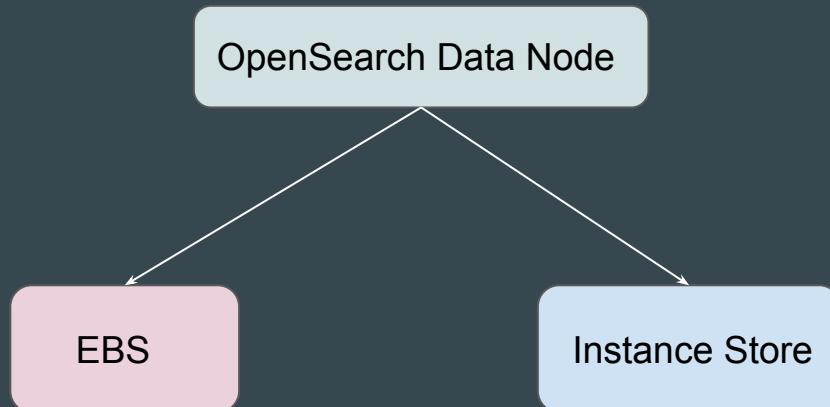
Ultra Warm Indices



Understanding the Basics

Standard data nodes use "**hot**" storage, which takes the form of instance stores or Amazon EBS volumes attached to each node.

Hot storage provides the fastest possible performance for indexing and searching new data.



Benefits of UltraWarm Mode

The UltraWarm tier acts like a caching layer on top of the data in Amazon S3.

UltraWarm moves data from Amazon S3 onto the UltraWarm nodes on demand, which speeds up access for subsequent queries on that data

You can add or remove UltraWarm nodes to increase or decrease the amount of cache against your data in Amazon S3 to optimize your cost per GB

Points to Note - UltraWarm

Data in UltraWarm is immutable (cannot modify the data)

If needed, you can bring back data to hot tier.

Cold Storage

Cold storage lets you store any amount of infrequently accessed or historical data on your Amazon OpenSearch Service domain and analyze it on demand, at a lower cost than other storage tiers

Similar to UltraWarm storage, cold storage is backed by Amazon S3. When you need to query cold data, you can selectively attach it to existing UltraWarm nodes.

Points to Note - Cold Storage

Data is not directly queryable and must be attached before being analyzed.

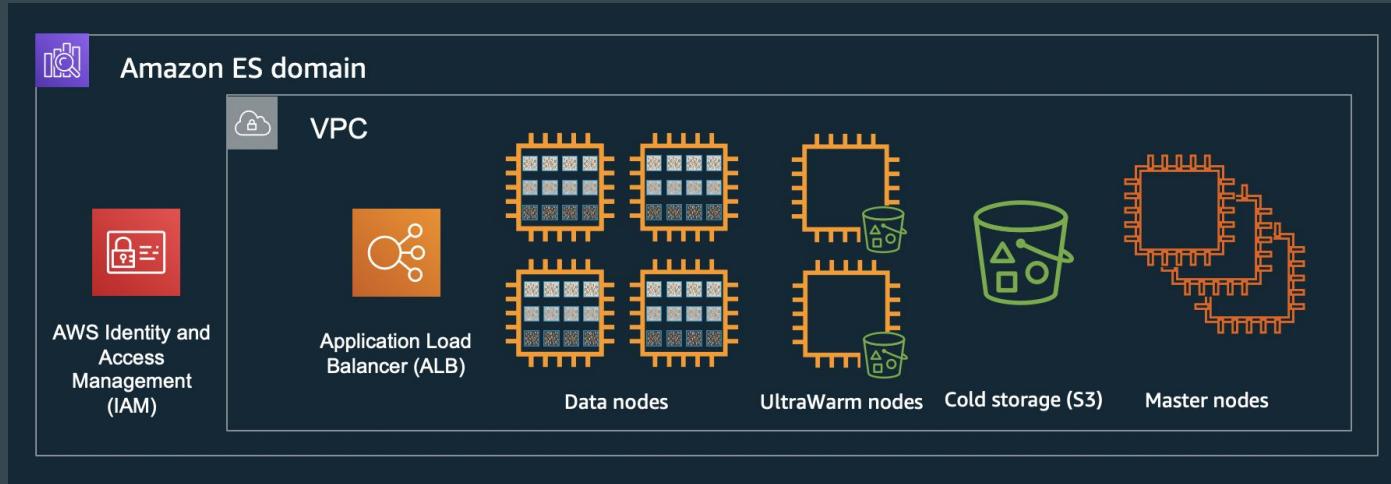
Billing Pointers

The **hot tier** requires you to pay for what is provisioned, which includes the hourly rate for the instance type. Storage is either Amazon EBS or a local SSD instance store.

UltraWarm nodes charge per hour just like other node types, but you only pay for the storage actually stored in Amazon S3.

Cold storage doesn't incur compute costs, and like UltraWarm, you're only billed for the amount of data stored in Amazon S3.

OpenSearch Architecture



Unified CloudWatch Agent

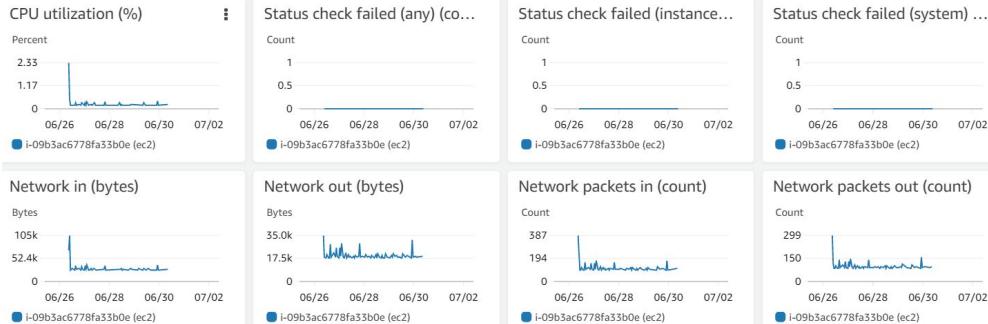
Metrics and Logs

Default CloudWatch Metrics

When we launch an EC2 instance in AWS, there are certain metrics that are captured by default.

Some of these include:

- CPU Utilization
- Network Related
- Disk Related

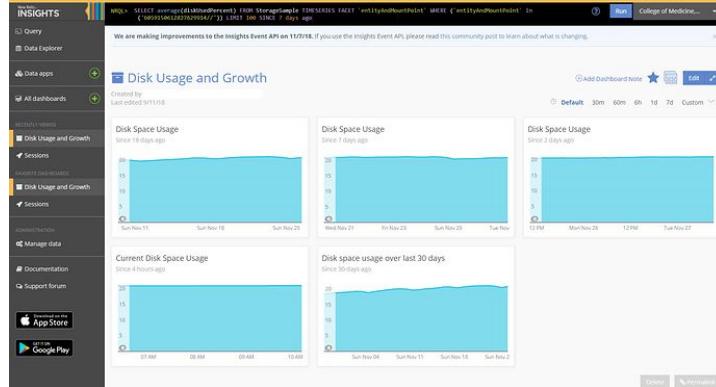


Challenge 1 -More Metrics Are Needed

There are various important metrics that needs to be collected in addition to the default ones.

Some of these include:

- Memory Metrics
- Disk Usage Metrics
- Netstat related.

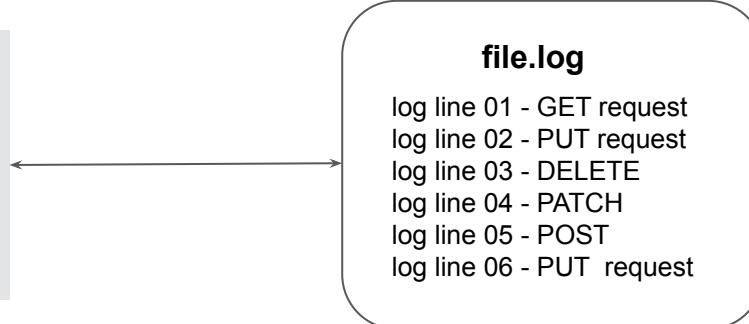


Challenge 2 - Log Monitoring

A server can contain a lot of log files, from system logs to the application logs.

During debugging, it is important to have log files at hand.

This means in default case; you need to give access to the server to an individual who wants to debug.

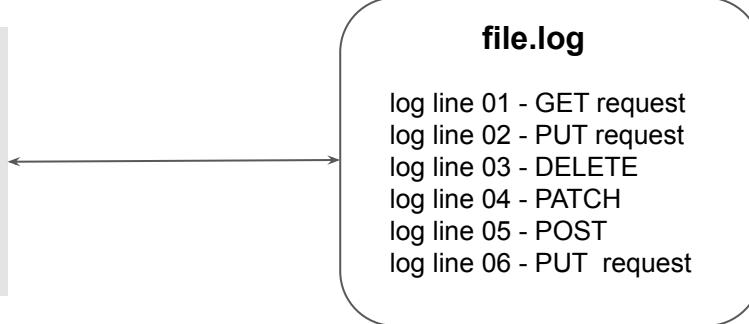
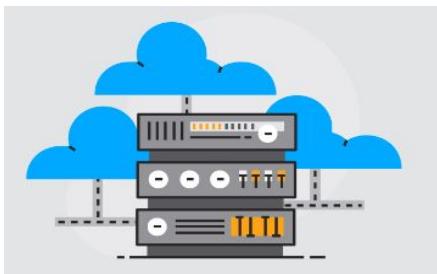


Disadvantage of the Approach

Access must be given to the server to the developers.

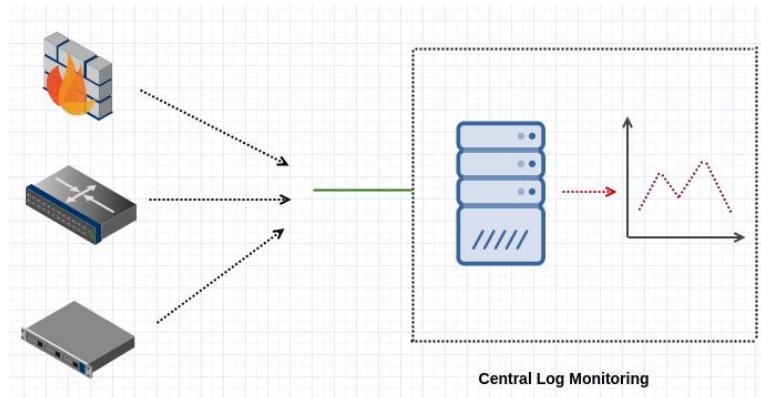
If the server gets terminated, the logs are lost.

No way to set up an alarm on certain conditions or create complex filters.



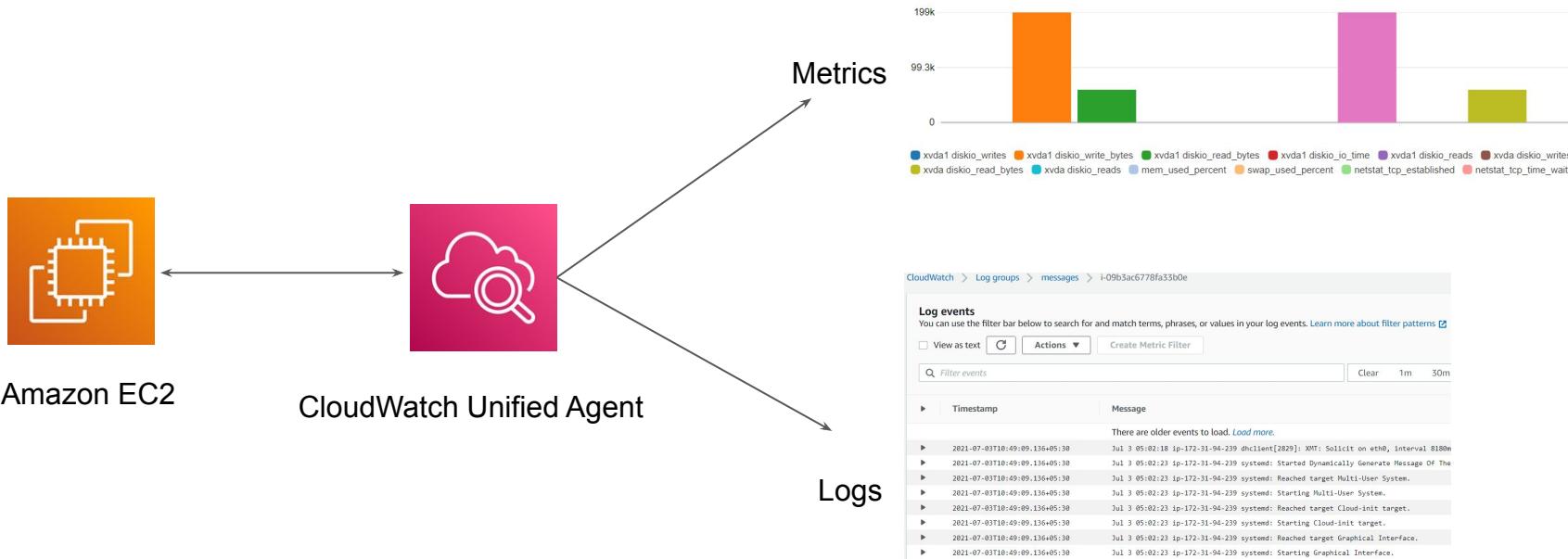
Better Way

- We create a Central Log Server.
- We push the log files from individual systems to Central Log Server.



Introducing Unified CloudWatch Agent

Unified CloudWatch Agent allows customers to capture both the internal system level metrics as well as logs collection.



How-To Steps

1. Create a IAM Role with CloudWatchAgentServer policy.
2. Create EC2 using IAM Role.
3. Install CloudWatch Agent.
4. Run CloudWatch Agent Configuration Wizard
5. Start Unified CloudWatch Agent.

Namespace, Metric, Dimension

CloudWatch Concepts

Important Concept 1 - Metrics

A metric is a variable that stores a time series data set.

Simple Analogy:

Metric = Variable to Monitor (CPU Utilization)

Data Points = Value Associated with the variable over a period of time (20%, 30%, 80%)



Important Concept 2 - Dimension

A dimension is a name/value pair that is part of the identity of a metric.

- CPU Utilization is 80%
- Dimension: instance-id is i-123456

Customers can also add their own dimensions, for example:

App=DB, Team=DB-IN

App=WebServer Team=SRE-US

Important Concept 3 - Namespace

Namespaces are basically container for metric.

Namespaces are useful if you want to avoid aggregating two different metrics with the same name.



Important Pointers

Metrics cannot be deleted, but they automatically expire after 15 months if no new data is published to them.

You can assign up to 10 dimensions to a metric.

CloudWatch treats each unique combination of dimensions as a separate metric, even if the metrics have the same metric name.

Metric Name: CPU Utilization

- Server=Prod, Team=SRE, Value 60
- Server=Dev, Team=APP, Value 90

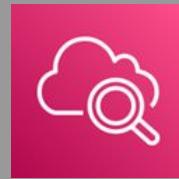
Important Pointers - Part 2

The size of a PutMetricData request is limited to 8KB for HTTP GET requests and 40KB for HTTP POST requests.

You can create up to 5000 alarms per AWS account. Metric data is kept for 2 weeks.

For free tier, you get 10 CloudWatch metrics, 10 alarms, 1,000,000 API requests, and 1,000 Amazon SNS email notifications per customer per month for free.

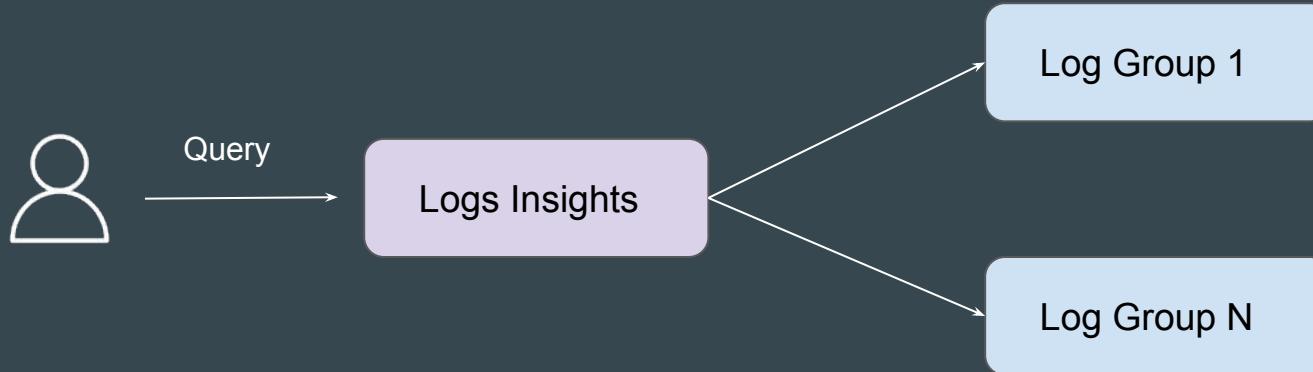
CloudWatch Logs Insights



Setting the Base

You can use CloudWatch Logs Insights to search log data that was sent to CloudWatch Logs using **purpose-built query language** with a few simple but powerful commands.

A single request can query up to 50 log groups.



Reference Screenshot

Logs Insights
Select log groups, and then run a query or [choose a sample query](#).

Start tailing 5m 30m 1h 3h 12h

Select log group(s)
`/aws/lambda/hello-world` X

Show more chosen log groups (+1) Clear all

```
1 stats count(*) by @LogStream
2 ... | limit 100
```

Run query Cancel Save History

Queries are allowed to run for up to 60 minutes.

Complete

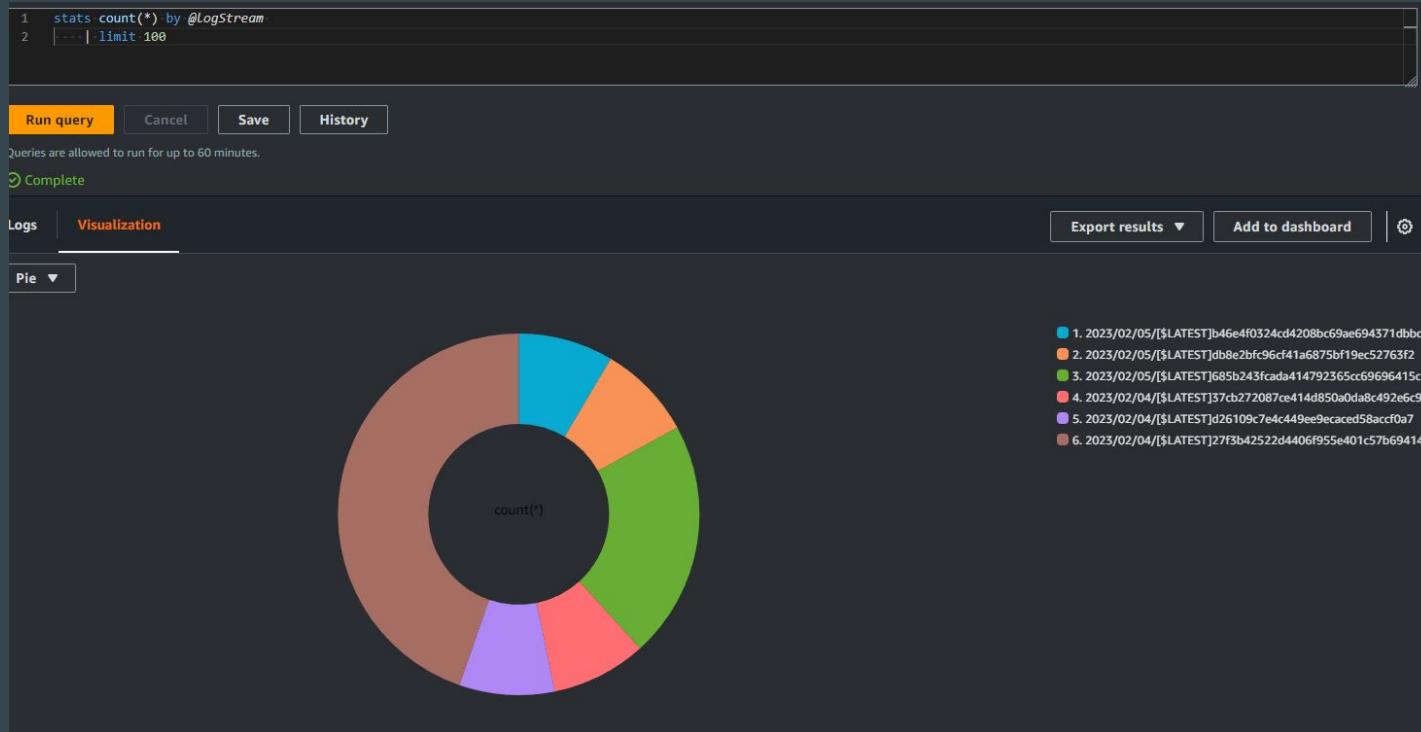
Logs Visualization Export results ▾ Add to

Showing 6 of 47 records matched ⓘ
47 records (5.9 kB) scanned in 4.3s @ 11 records/s (1.4 kB/s)



#	@LogStream	count(*)
► 1	2023/02/05/[...]b46e4f0324cd4208bc69ae694371dbbc	4
► 2	2023/02/05/[...]db8e2bfc96cf41a6875bf19ec52763f2	4
► 3	2023/02/05/[...]685b243fcada414792365cc69696415c	10
► 4	2023/02/04/[...]37cb272087ce414d850a0da8c492e6c9	4
► 5	2023/02/04/[...]d26109c7e4c449ee9ecaced58accf0a7	4
► 6	2023/02/04/[...]27f3b42522d4406f955e401c57b69414	21

Reference Screenshot - Visualization



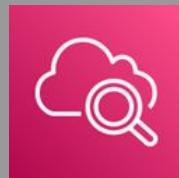
Points to Note

You can perform queries to help you more efficiently and effectively respond to operational issues.

If an issue occurs, you can use CloudWatch Logs Insights to identify potential causes and validate deployed fixes.

CloudWatch Logs Insights queries incur charges based on the amount of data that is queried.

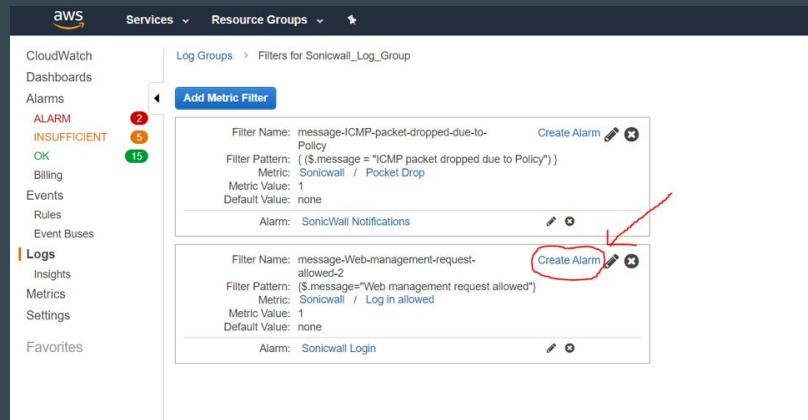
CloudWatch Metric Filters



Setting the Base

You can **search and filter** the log data coming into CloudWatch Logs by creating one or more metric filters

CloudWatch Logs uses these metric filters to turn log data into numerical CloudWatch metrics **that you can graph or set an alarm on**.



Reference Screenshot

Filter pattern
Specify the terms or pattern to match in your log events to create metrics.

 X

Test pattern

Select log data to test

Log event messages

Type log data to test with your Filter Pattern. Please use line breaks to separate log events.

```
{"eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}, "eventVersion":"1.08","userIdentity":{"type":"AssumedRole","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}}
```

Test pattern

Results

Found 4 matches out of 50 event(s) in the sample log.

▼ Show test results

Event number	Event message
15	{"eventVersion":"1.08","userIdentity":{"type":"IAMUser","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}}
16	{"eventVersion":"1.08","userIdentity":{"type":"IAMUser","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}}
17	{"eventVersion":"1.08","userIdentity":{"type":"IAMUser","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}}
20	{"eventVersion":"1.08","userIdentity":{"type":"Root","principalId":"AROAAQGgAAAAA...","sessionContext":{},"invokedBy":null}}

◀ ▶

Sample Queries

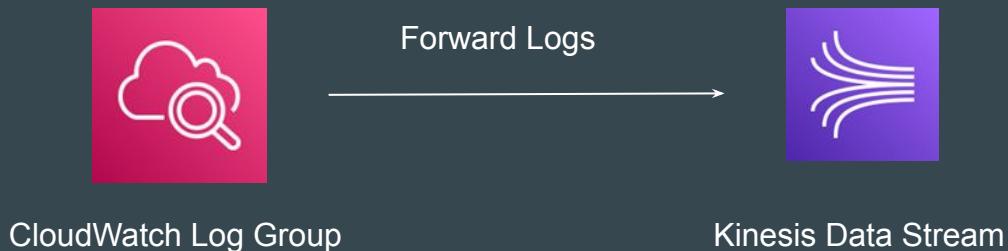
Queries	Description
ERROR	Single-term filter pattern that returns all log events where messages contain the word ERROR.
ERROR ARGUMENTS	multiple-term filter pattern that returns all log events where messages contain the words ERROR and ARGUMENTS.
?ERROR ?ARGUMENTS	filter pattern that returns all log events where messages contain the word ERROR or the word ARGUMENTS.
ERROR -ARGUMENTS	filter pattern that returns log events where messages include the term ERROR and exclude the term ARGUMENTS.
" "	filter pattern that returns all log events.

CloudWatch Logs Subscription Filter



Setting the Base

You can use subscriptions to get access to a real-time feed of log events from CloudWatch Logs and **have it delivered to other services** such as Kinesis stream, Lambda function for custom processing, analysis, or loading to other systems.



Supported Subscription Filter

As of now, following destinations are supported as part of subscription filter.

OpenSearch, Kinesis, Kinesis Firehose, and Lambda

The screenshot shows the AWS CloudWatch Metrics console interface. The top navigation bar includes links for 'Log streams', 'Metric filters', 'Subscription filters' (which is highlighted in orange), 'Contributor Insights', 'Tags', and 'Data protection'. Below the navigation bar, a message states: 'Subscription filters (0) We now support up to 2 subscription filters per log group.' A search bar at the top has placeholder text 'Filter name' and 'Filter pattern'. A note below the search bar says 'There are no subscription filters.' On the right side, there is a 'Create' button with an upward arrow icon, followed by four options: 'Create Amazon OpenSearch Service subscription filter', 'Create Kinesis subscription filter', 'Create Kinesis Firehose subscription filter', and 'Create Lambda subscription filter'. The 'Create Kinesis subscription filter' option is currently selected, indicated by a blue border around its corresponding button.

Point to Note

We can customize on the type of logs that are sent to destination.

Example:

Delivery every logged activity made by “Root” AWS credentials to Kinesis Data Stream.

The screenshot shows the 'Subscription filter pattern' configuration page for AWS CloudTrail. It includes fields for 'Subscription filter pattern' containing the condition `[$.userIdentity.type = Root]`, 'Subscription filter name' set to 'RootAccess', and a 'Test pattern' section with a dropdown for 'Select log data to test' showing '042025557788_CloudTrail_ap-southeast-1_3'. The 'Log event messages' section displays several JSON log entries, all of which begin with the event version '1.08' and reference the principal ID '042025557788'. A 'Test pattern' button is at the bottom.

```
[{"eventVersion": "1.08", "userIdentity": {"type": "Root", "principalId": "042025557788", "arn": "arn:aws:sts::042025557788:assumed-role/CloudWatchLogs-Root/Root"}, {"eventVersion": "1.08", "userIdentity": {"type": "Root", "principalId": "042025557788", "arn": "arn:aws:sts::042025557788:root"}, {"eventVersion": "1.08", "userIdentity": {"type": "AssumedRole", "principalId": "AROAQTSHEUJL5V5Z5H5A", "arn": "arn:aws:sts::042025557788:assumed-role/CloudWatchLogs-Root/Root"}, {"eventVersion": "1.08", "userIdentity": {"type": "AWSService", "invokedBy": "support.amazonaws.com"}, {"eventVersion": "1.08", "userIdentity": {"type": "AWSService", "invokedBy": "support.amazonaws.com"}, {"eventVersion": "1.08", "userIdentity": {"type": "AWSService", "invokedBy": "support.amazonaws.com"}]
```

VPC Flow Logs

Logs are Awesome

Simple Analogy - Visitor Register

In many of the societies across India, whenever a visitor visits, they first have to fill in their information in the visitor register.

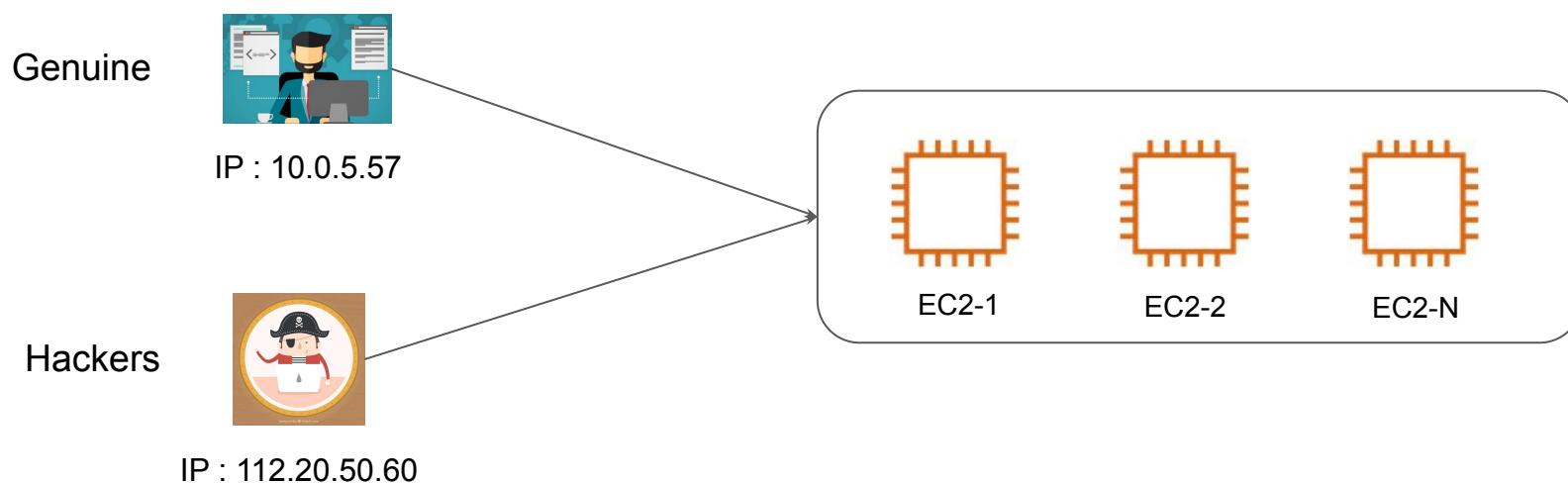
Some of the information includes:

- Name
- Source Place.
- Destination Place.
- Entry and Exit Date/Time
- Purpose of Work

CARD NO.	NAME OF THE VISITOR	VISITOR'S ADDRESS	WHOM TO SEE	PURPOSE OF THE VISIT		TIME	SIGNATORY OF VISITOR	CONTACT & SECURITY	MOBILE NO.
				IN	OUT				
220	Amit Pravin Gajera	9, M.G. Road	4 th	15:00					
221	Shashikant Sengar	Wardha 2 nd							
222	Umesh Patel	Gandhinagar	Chand						
223	Umesh Patel	Gandhinagar	Chand						
224	Umesh Patel	Gandhinagar	Chand						
225	Umesh Patel	Gandhinagar	Chand						
226	R.D. Kothari	Paldi	Prabodh						
227	Dinesh Patel	Kothi	Patel						
228	Dinesh Patel	Kothi	Patel						
229	Dinesh Patel	Kothi	Patel						
230	Dinesh Patel	Kothi	Patel						
231	Dinesh Patel	Kothi	Patel						
232	Dinesh Patel	Kothi	Patel						
233	Dinesh Patel	Kothi	Patel						
234	Dinesh Patel	Kothi	Patel						
235	Dinesh Patel	Kothi	Patel						
236	Dinesh Patel	Kothi	Patel						
237	Dinesh Patel	Kothi	Patel						
238	Dinesh Patel	Kothi	Patel						
239	Dinesh Patel	Kothi	Patel						
240	Dinesh Patel	Kothi	Patel						
241	Dinesh Patel	Kothi	Patel						
242	Dinesh Patel	Kothi	Patel						
243	Dinesh Patel	Kothi	Patel						
244	Dinesh Patel	Kothi	Patel						
245	Dinesh Patel	Kothi	Patel						
246	Dinesh Patel	Kothi	Patel						
247	Dinesh Patel	Kothi	Patel						
248	Dinesh Patel	Kothi	Patel						
249	Dinesh Patel	Kothi	Patel						
250	Dinesh Patel	Kothi	Patel						
251	Dinesh Patel	Kothi	Patel						
252	Dinesh Patel	Kothi	Patel						
253	Dinesh Patel	Kothi	Patel						
254	Dinesh Patel	Kothi	Patel						
255	Dinesh Patel	Kothi	Patel						
256	Dinesh Patel	Kothi	Patel						
257	Dinesh Patel	Kothi	Patel						
258	Dinesh Patel	Kothi	Patel						
259	Dinesh Patel	Kothi	Patel						
260	Dinesh Patel	Kothi	Patel						
261	Dinesh Patel	Kothi	Patel						
262	Dinesh Patel	Kothi	Patel						
263	Dinesh Patel	Kothi	Patel						
264	Dinesh Patel	Kothi	Patel						
265	Dinesh Patel	Kothi	Patel						
266	Dinesh Patel	Kothi	Patel						
267	Dinesh Patel	Kothi	Patel						
268	Dinesh Patel	Kothi	Patel						
269	Dinesh Patel	Kothi	Patel						
270	Dinesh Patel	Kothi	Patel						
271	Dinesh Patel	Kothi	Patel						
272	Dinesh Patel	Kothi	Patel						
273	Dinesh Patel	Kothi	Patel						
274	Dinesh Patel	Kothi	Patel						
275	Dinesh Patel	Kothi	Patel						
276	Dinesh Patel	Kothi	Patel						
277	Dinesh Patel	Kothi	Patel						
278	Dinesh Patel	Kothi	Patel						
279	Dinesh Patel	Kothi	Patel						
280	Dinesh Patel	Kothi	Patel						
281	Dinesh Patel	Kothi	Patel						
282	Dinesh Patel	Kothi	Patel						
283	Dinesh Patel	Kothi	Patel						
284	Dinesh Patel	Kothi	Patel						
285	Dinesh Patel	Kothi	Patel						
286	Dinesh Patel	Kothi	Patel						
287	Dinesh Patel	Kothi	Patel						
288	Dinesh Patel	Kothi	Patel						
289	Dinesh Patel	Kothi	Patel						
290	Dinesh Patel	Kothi	Patel						
291	Dinesh Patel	Kothi	Patel						
292	Dinesh Patel	Kothi	Patel						
293	Dinesh Patel	Kothi	Patel						
294	Dinesh Patel	Kothi	Patel						
295	Dinesh Patel	Kothi	Patel						
296	Dinesh Patel	Kothi	Patel						
297	Dinesh Patel	Kothi	Patel						
298	Dinesh Patel	Kothi	Patel						
299	Dinesh Patel	Kothi	Patel						
300	Dinesh Patel	Kothi	Patel						
301	Dinesh Patel	Kothi	Patel						
302	Dinesh Patel	Kothi	Patel						
303	Dinesh Patel	Kothi	Patel						
304	Dinesh Patel	Kothi	Patel						
305	Dinesh Patel	Kothi	Patel						
306	Dinesh Patel	Kothi	Patel						
307	Dinesh Patel	Kothi	Patel						
308	Dinesh Patel	Kothi	Patel						
309	Dinesh Patel	Kothi	Patel						
310	Dinesh Patel	Kothi	Patel						
311	Dinesh Patel	Kothi	Patel						
312	Dinesh Patel	Kothi	Patel						
313	Dinesh Patel	Kothi	Patel						
314	Dinesh Patel	Kothi	Patel						
315	Dinesh Patel	Kothi	Patel						
316	Dinesh Patel	Kothi	Patel						
317	Dinesh Patel	Kothi	Patel						
318	Dinesh Patel	Kothi	Patel						
319	Dinesh Patel	Kothi	Patel						
320	Dinesh Patel	Kothi	Patel						
321	Dinesh Patel	Kothi	Patel						
322	Dinesh Patel	Kothi	Patel						
323	Dinesh Patel	Kothi	Patel						
324	Dinesh Patel	Kothi	Patel						
325	Dinesh Patel	Kothi	Patel						
326	Dinesh Patel	Kothi	Patel						
327	Dinesh Patel	Kothi	Patel						
328	Dinesh Patel	Kothi	Patel						
329	Dinesh Patel	Kothi	Patel						
330	Dinesh Patel	Kothi	Patel						
331	Dinesh Patel	Kothi	Patel						
332	Dinesh Patel	Kothi	Patel						
333	Dinesh Patel	Kothi	Patel						
334	Dinesh Patel	Kothi	Patel						
335	Dinesh Patel	Kothi	Patel						
336	Dinesh Patel	Kothi	Patel						
337	Dinesh Patel	Kothi	Patel						
338	Dinesh Patel	Kothi	Patel						
339	Dinesh Patel	Kothi	Patel						
340	Dinesh Patel	Kothi	Patel						
341	Dinesh Patel	Kothi	Patel						
342	Dinesh Patel	Kothi	Patel						
343	Dinesh Patel	Kothi	Patel						
344	Dinesh Patel	Kothi	Patel						
345	Dinesh Patel	Kothi	Patel						
346	Dinesh Patel	Kothi	Patel						
347	Dinesh Patel	Kothi	Patel						
348	Dinesh Patel	Kothi	Patel						
349	Dinesh Patel	Kothi	Patel						
350	Dinesh Patel	Kothi	Patel						
351	Dinesh Patel	Kothi	Patel						
352	Dinesh Patel	Kothi	Patel						
353	Dinesh Patel	Kothi	Patel						
354	Dinesh Patel	Kothi	Patel						
355	Dinesh Patel	Kothi	Patel						
356	Dinesh Patel	Kothi	Patel						
357	Dinesh Patel	Kothi	Patel						
358	Dinesh Patel	Kothi	Patel						
359	Dinesh Patel	Kothi	Patel						
360	Dinesh Patel	Kothi	Patel						
361	Dinesh Patel	Kothi	Patel						
362	Dinesh Patel	Kothi	Patel						
363	Dinesh Patel	Kothi	Patel						
364	Dinesh Patel	Kothi	Patel						
365	Dinesh Patel	Kothi	Patel						
366	Dinesh Patel	Kothi	Patel						
367	Dinesh Patel	Kothi	Patel						
368	Dinesh Patel	Kothi	Patel						
369	Dinesh Patel	Kothi	Patel						
370	Dinesh Patel	Kothi	Patel						
371	Dinesh Patel	Kothi	Patel						
372	Dinesh Patel	Kothi	Patel						
373	Dinesh Patel	Kothi	Patel						
374	Dinesh Patel	Kothi	Patel						
375	Dinesh Patel	Kothi	Patel						
376	Dinesh Patel	Kothi	Patel						
377	Dinesh Patel	Kothi	Patel						
378	Dinesh Patel	Kothi	Patel						
379	Dinesh Patel	Kothi	Patel						
380	Dinesh Patel	Kothi	Patel						
381	Dinesh Patel	Kothi	Patel						
382	Dinesh Patel	Kothi	Patel						
383	Dinesh Patel	Kothi	Patel						
384	Dinesh Patel	Kothi	Patel						
385	Dinesh Patel	Kothi	Patel						
386	Dinesh Patel	Kothi	Patel						
387	Dinesh Patel	Kothi	Patel						
388	Dinesh Patel	Kothi	Patel						
389	Dinesh Patel	Kothi	Patel						
390	Dinesh Patel	Kothi	Patel						
391	Dinesh Patel	Kothi	Patel						
392	Dinesh Patel	Kothi	Patel						
393	Dinesh Patel	Kothi	Patel						
394	Dinesh Patel	Kothi	Patel						
395	Dinesh Patel	Kothi	Patel						
396	Dinesh Patel	Kothi	Patel						
397	Dinesh Patel	Kothi	Patel						
398	Dinesh Patel	Kothi	Patel						
399	Dinesh Patel	Kothi	Patel						
400	Dinesh Patel	Kothi	Patel						
401	Dinesh Patel	Kothi	Patel						
402	Dinesh Patel	Kothi	Patel						
403	Dinesh Patel	Kothi	Patel						
404	Dinesh Patel	Kothi	Patel						
405	Dinesh Patel	Kothi	Patel						
406	Dinesh Patel	Kothi	Patel						
407	Dinesh Patel	Kothi	Patel						
408	Dinesh Patel	Kothi	Patel						
409	Dinesh Patel	Kothi	Patel						
410	Dinesh Patel	Kothi	Patel						
411	Dinesh Patel	Kothi	Patel						
412	Dinesh Patel	Kothi	Patel						
413	Dinesh Patel	Kothi	Patel						
414	Dinesh Patel	Kothi	Patel						
415	Dinesh Patel	Kothi	Patel						
416	Dinesh Patel	Kothi	Patel						
417	Dinesh Patel	Kothi	Patel						
418	Dinesh Patel	Kothi	Patel		</td				

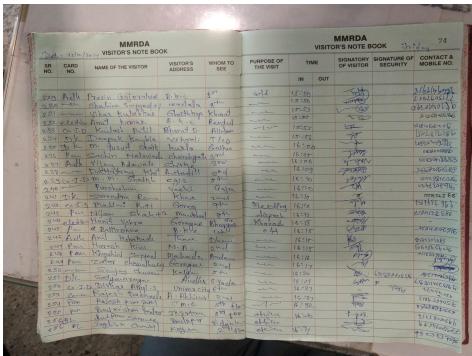
Comparing Analogy with AWS Environment

Even in AWS, there can be thousands of users across the world who might be visiting your environment.

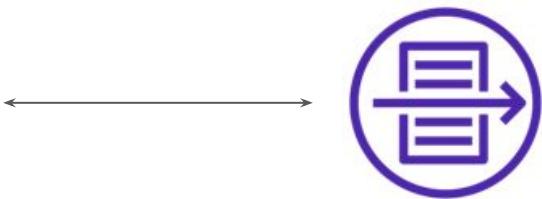


VPC Flow Logs

VPC Flow Logs is a feature that enables you to capture information about the IP traffic going to and from network interfaces in your VPC.



Visitor Register

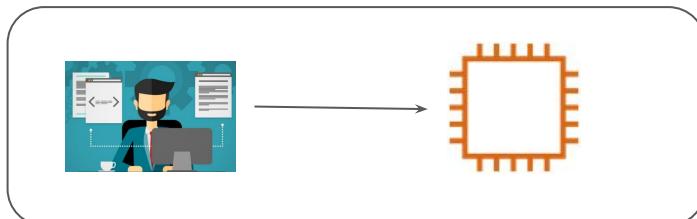


VPC Flow Log

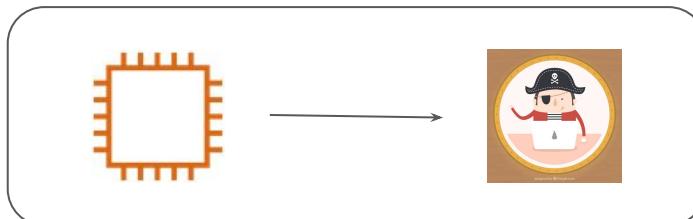
Capture Information Scope

The scope of the VPC Flow logs:

1. Record the traffic information that is visiting the resource (eg EC2)
2. Record data about resource connecting to which outbound endpoint.



10.77.2.50 → EC2 Instance



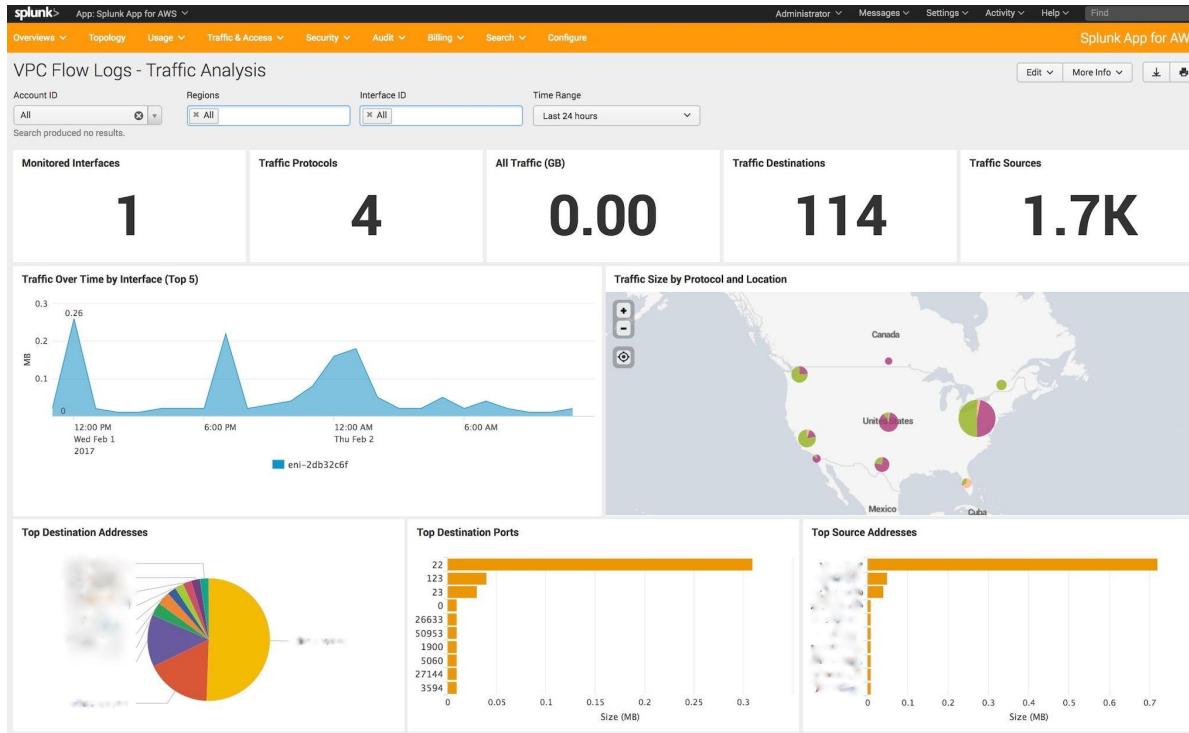
EC2 Instance → 192.168.0.5

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

<input type="checkbox"/> View as text	 Actions ▾	Create Metric Filter
Filter events		Clear 1m 30m 1h 12h Custom 
▶	Timestamp	Message
No older events at this moment. Retry		
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 50.205.244.36 172.31.94.239 123 34874 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 1.116.229.53 80 59807 6 1 40 1623168611 1623168640 AC...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 8.129.43.176 172.31.94.239 48507 2376 6 1 40 1623168611 1623168640 ...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 204.11.201.12 39609 123 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 138.68.201.49 55618 123 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 204.11.201.12 172.31.94.239 123 39609 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 138.68.201.49 172.31.94.239 123 55618 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 69.89.207.199 53680 123 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 1.116.229.53 172.31.94.239 59807 80 6 1 40 1623168611 1623168640 AC...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 69.89.207.199 172.31.94.239 123 53680 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 162.142.125.150 172.31.94.239 62446 9143 6 1 44 1623168611 16231686...
▶	2021-06-08T21:40:11.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 50.205.244.36 34874 123 17 1 76 1623168611 1623168640...
▶	2021-06-08T21:40:46.000+05:30	2 693331494763 eni-025ffffb751de82493 107.173.140.175 172.31.94.239 49640 8088 6 1 44 1623168646 16231687...
▶	2021-06-08T21:40:46.000+05:30	2 693331494763 eni-025ffffb751de82493 50.205.244.36 172.31.94.239 123 35182 17 1 76 1623168646 1623168700...
▶	2021-06-08T21:40:46.000+05:30	2 693331494763 eni-025ffffb751de82493 172.31.94.239 50.205.244.36 35182 123 17 1 76 1623168646 1623168700...

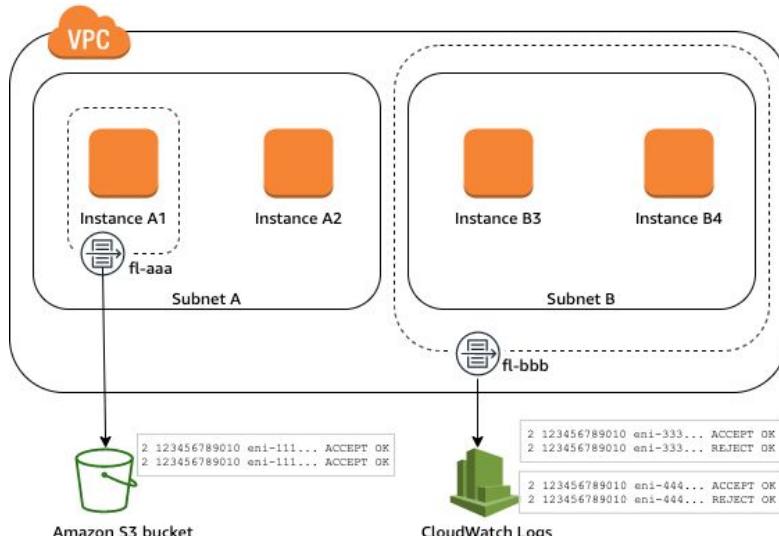
Dashboards Built using VPC Flow Logs Data



Interface Level Flow Logs

VPC Flow Logs captures traffic at an interface level.

Flow logs do not capture real-time log streams for your network interfaces.



High-Level Flow Logs Format

version	- The VPC Flow Logs Version
account-id	- AWS Account ID
interface-id	- The network interface id
srcaddr	- The source address
destaddr	- Destination Address
src port	- Source Port
dest port	- Destination Port
protocol	- The protocol number
packets	- Number of packets transferred
bytes	- Number of bytes transferred
start	- Start time in unix seconds
end	- End time in unix seconds
action	- ACCEPT or REJECT
log status	- Logging status of flow log

2 7742829482 eni-4d788e3d 115.73.149.218 10.0.5.157 12053 23 6 2 88 1485439809 1485440090 REJECT OK

Type of Traffic Not Logged

Flow logs do not capture all IP traffic. Some of these include:

- Traffic generated by instances when they contact the Amazon DNS server. If you use your own DNS server, then all traffic to that DNS server is logged.
- Traffic generated by a Windows instance for Amazon Windows license activation.
- Traffic to and from 169.254.169.254 for instance metadata.
- DHCP traffic.

CloudTrail - Log File Integrity Validation

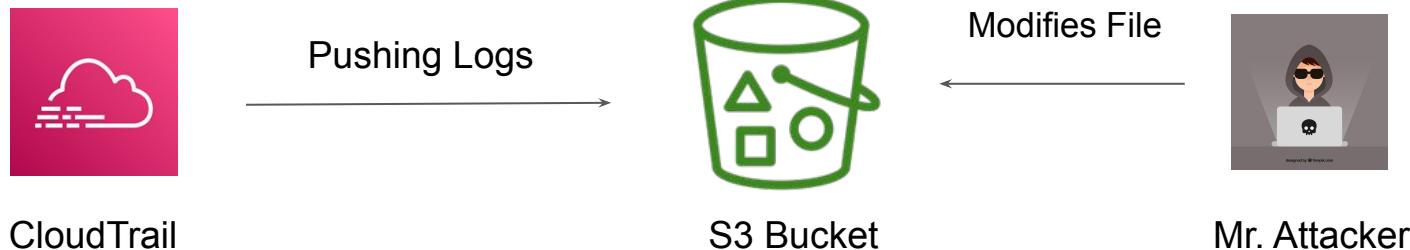
Back to Security!



Getting Started

CloudTrail log file integrity validation allows us to determine whether a log file was modified, deleted, or unchanged after CloudTrail delivered it.

This feature is built using industry standard algorithms: SHA-256 for hashing and SHA-256 with RSA for digital signing.



Basic Working

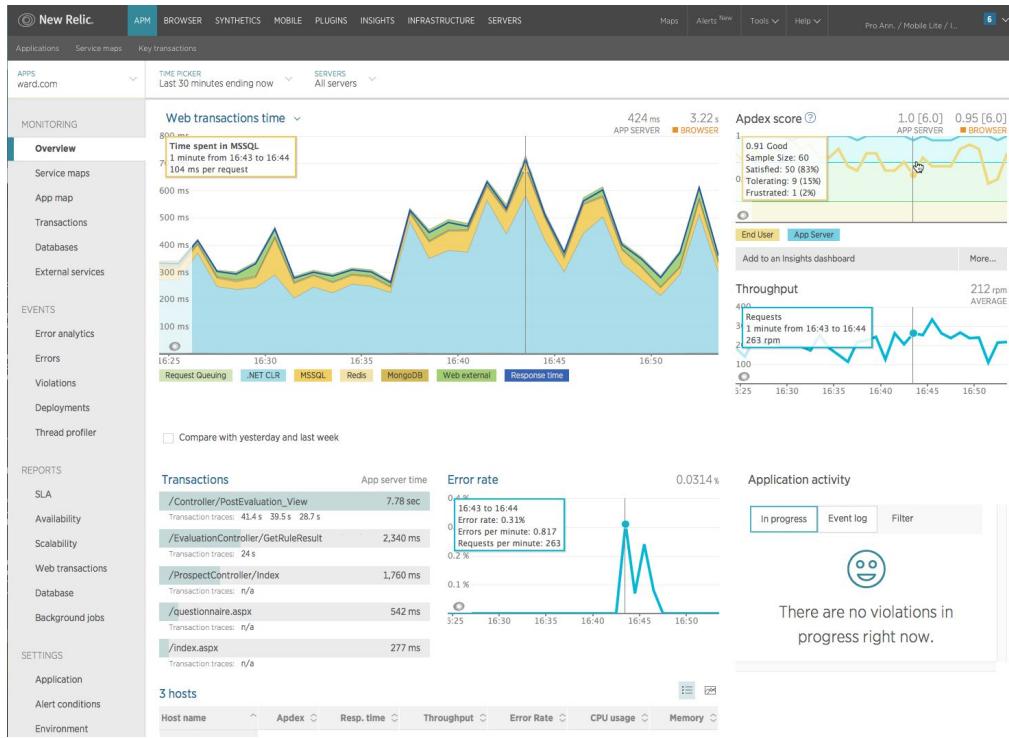
When you enable log file integrity validation, CloudTrail creates a hash for every log file that it delivers.

Every hour, CloudTrail also creates and delivers a file that references the log files for the last hour and contains a hash of each. This file is called a digest file.

X-Ray

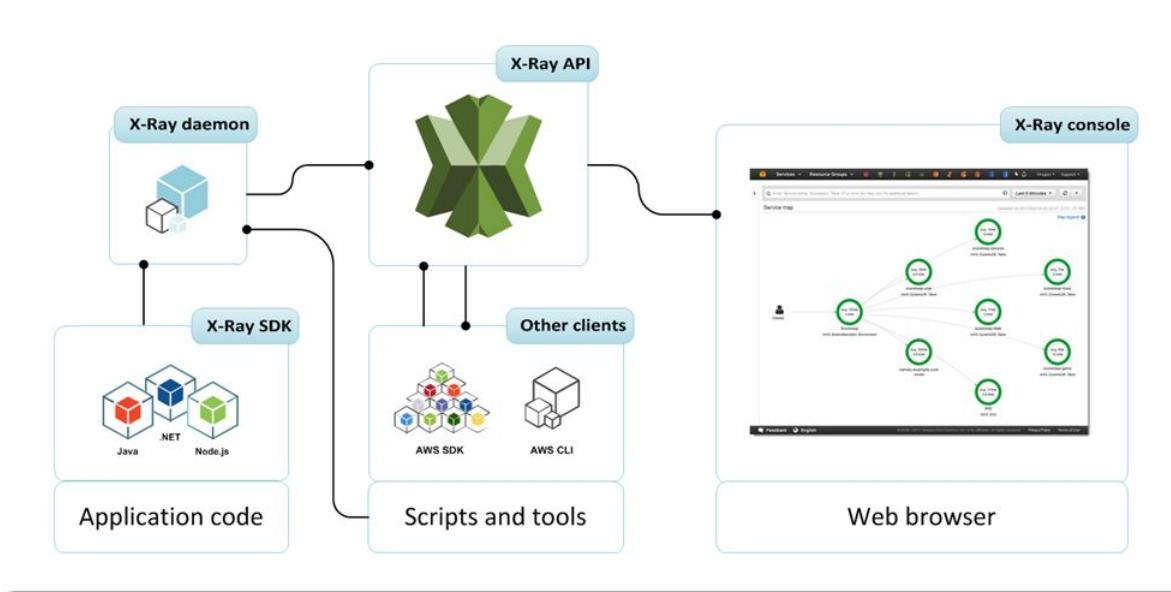
Debugging & Monitoring Applications

Traditional APM



Overview of X-Ray

AWS X-Ray allows us to debug our applications with functionality of request tracing so we can find the root cause and performance issues.



X-Ray Integration

AWS X-Ray provides integration with various AWS services like:

- AWS EC2
- Lambda
- Elastic Load Balancing
- API Gateway
- Elastic Beanstalk

X-Ray Supported Platform

AWS X-Ray provides support for following platforms:

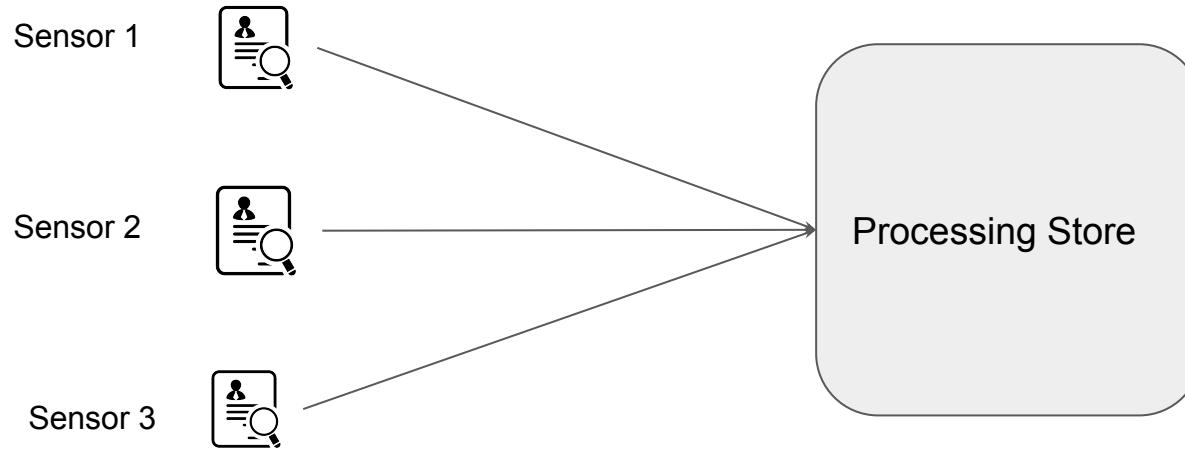
- Java
- Go
- .NET
- Ruby
- Python
- Node JS

Amazon Kinesis

Streaming Data

Basics of Streaming Data.

Streaming data is the continuous flow of data generated by various sources



Examples of Streaming Data

A financial institution tracks changes in the stock market in real time and adjust it's portfolio accordingly.

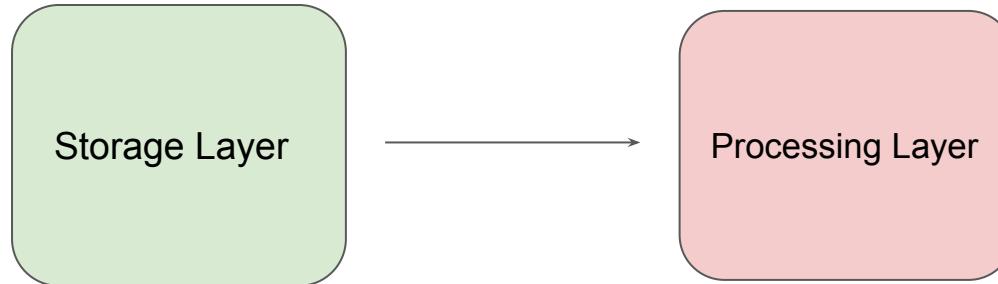
A media publisher streams billions of clickstream records from its online properties



Challenges with Working of Streaming Data

Streaming data processing requires two layers: a storage layer and a processing layer.

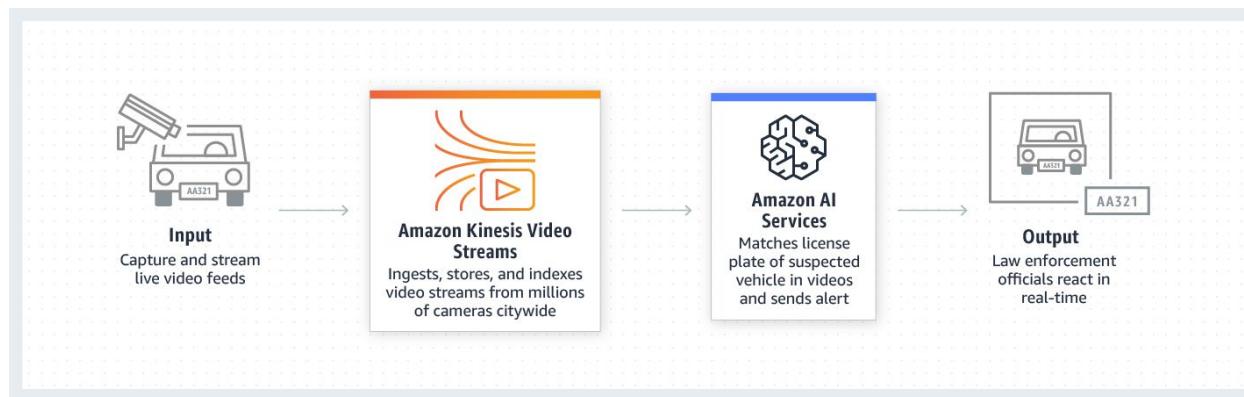
The storage layer needs to support record ordering and strong consistency, replayable reads and the processing layer is responsible for consuming data from the storage layer, running computation on that data and many other tasks.



Basics of Amazon Kinesis

Amazon Kinesis makes it easy to collect, process, and analyze real-time, streaming data so you can get timely insights and react quickly to new information.

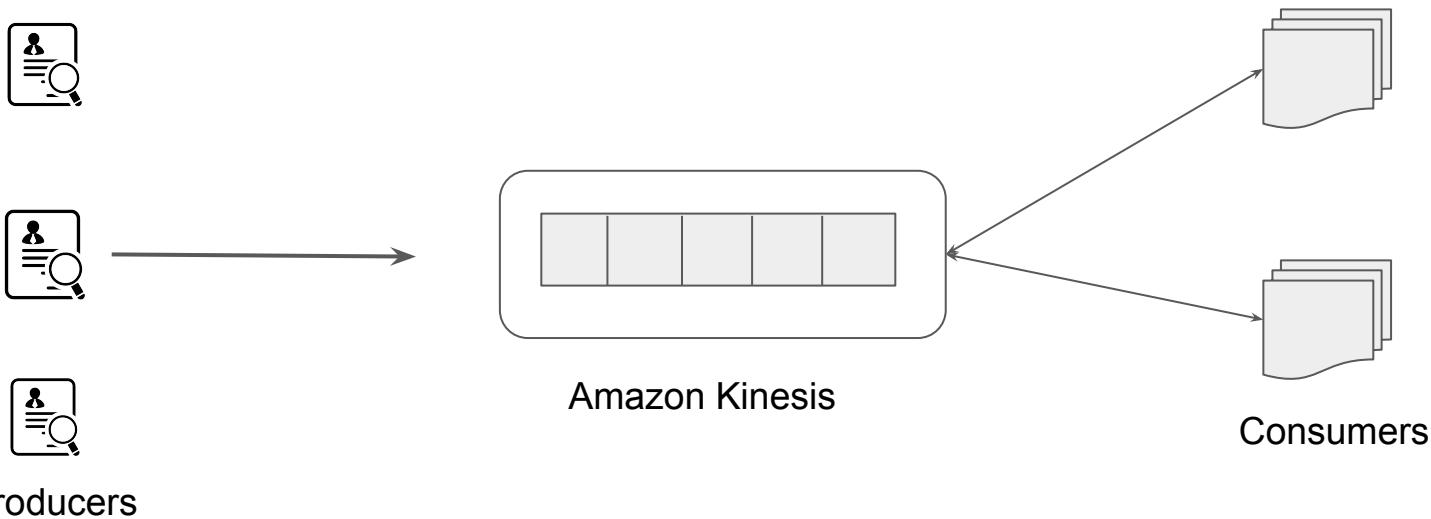
Amazon Kinesis offers key capabilities to cost-effectively process streaming data at any scale



3 entities

There are 3 entities in this kind of use case:

Producer, Stream Store, Consumer



Amazon Kinesis Services

Capabilities of Kinesis Set of Services

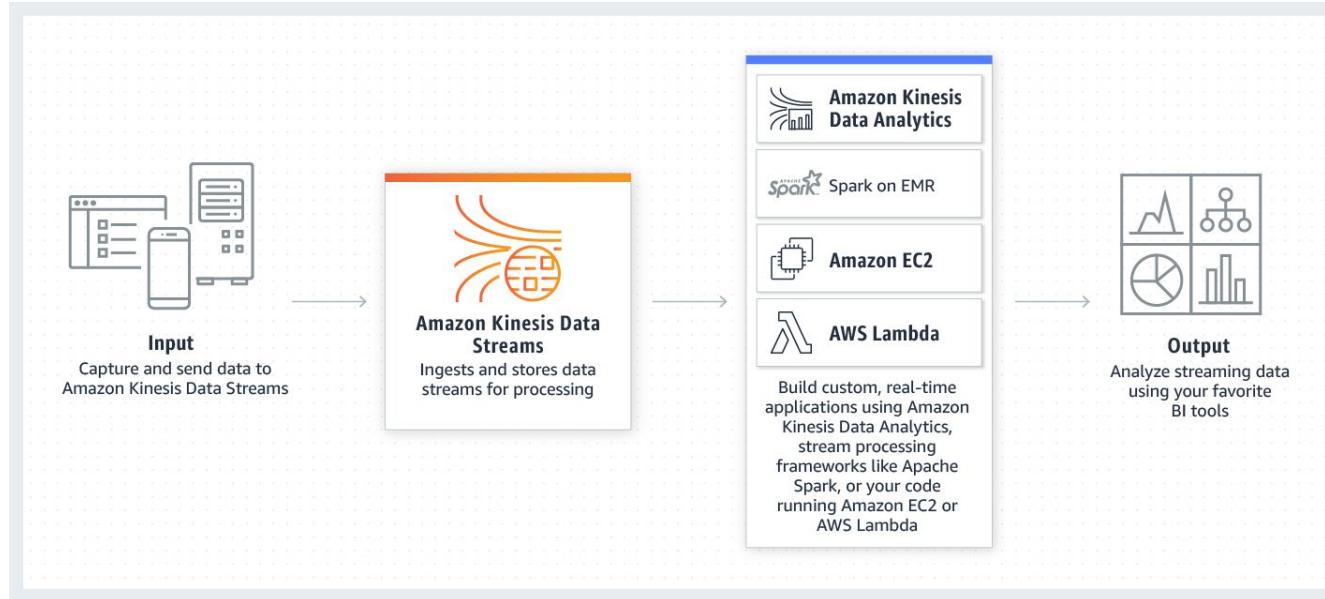
Kinesis Offerings

Amazon Kinesis is a set of services which makes it easy to work with set of streaming data on AWS.

Sr No	Kinesis Services	Description
1	Kinesis Data Stream	Captures, processes and stores data streams in real-time
2	Kinesis Data Firehose	Primary to move data from point A to point B.
3	Kinesis Data Analytics	Analyze streaming data in real-time with SQL / Java code.
4	Kinesis Video Stream	Capture, processes and stores video streams.

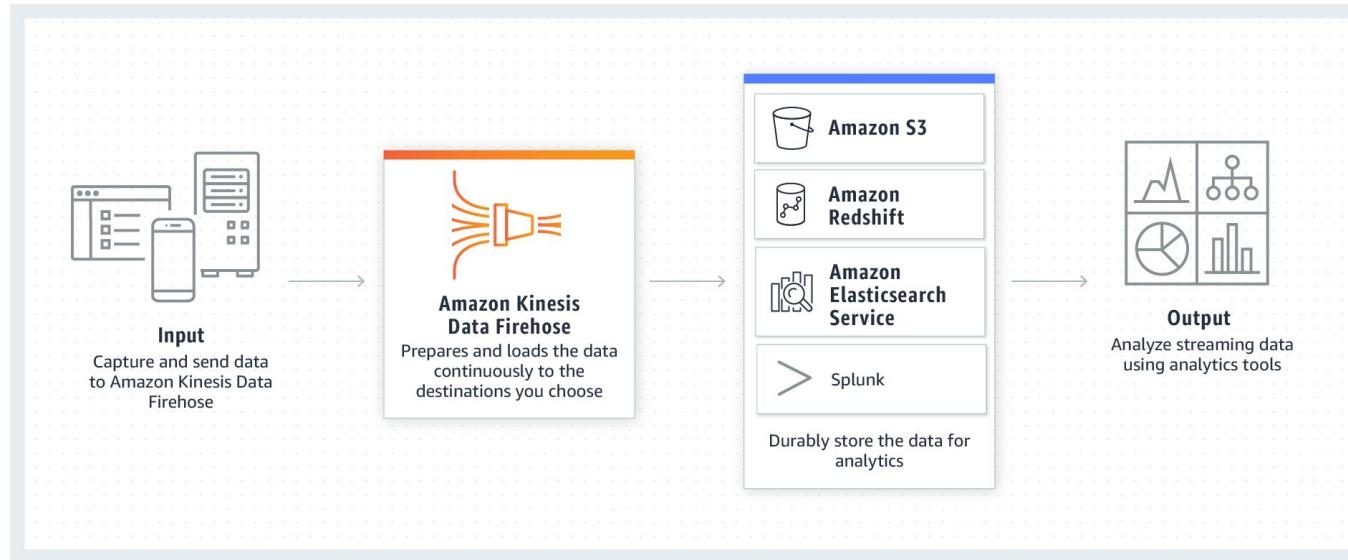
Kinesis Data Stream

It allows us to capture, process and store data streams.



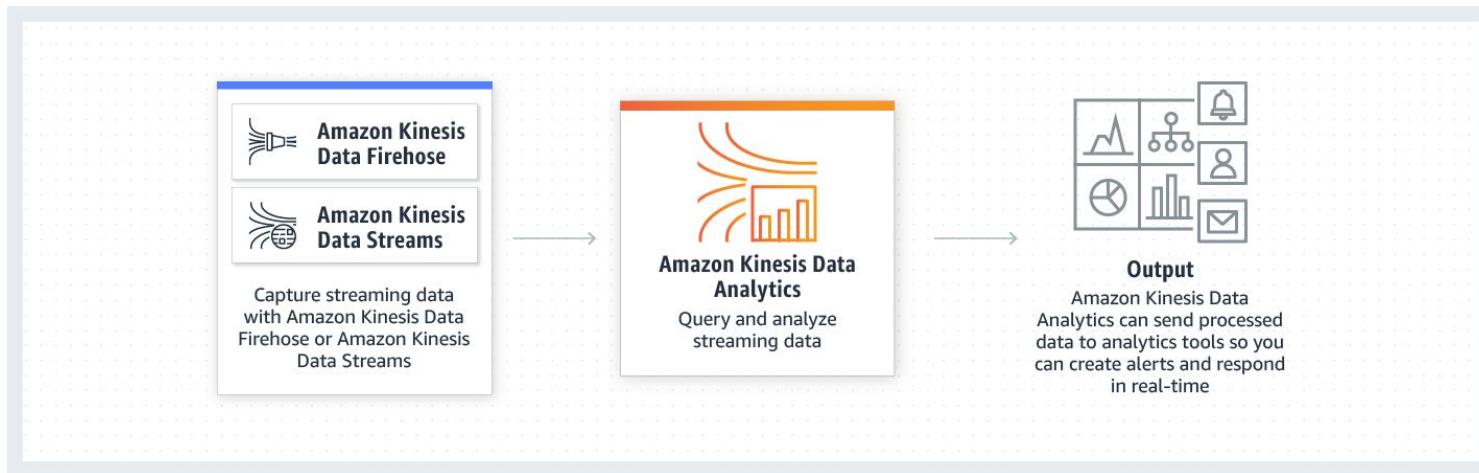
Kinesis Firehose

Kinesis firehose delivers data from point A to point B.



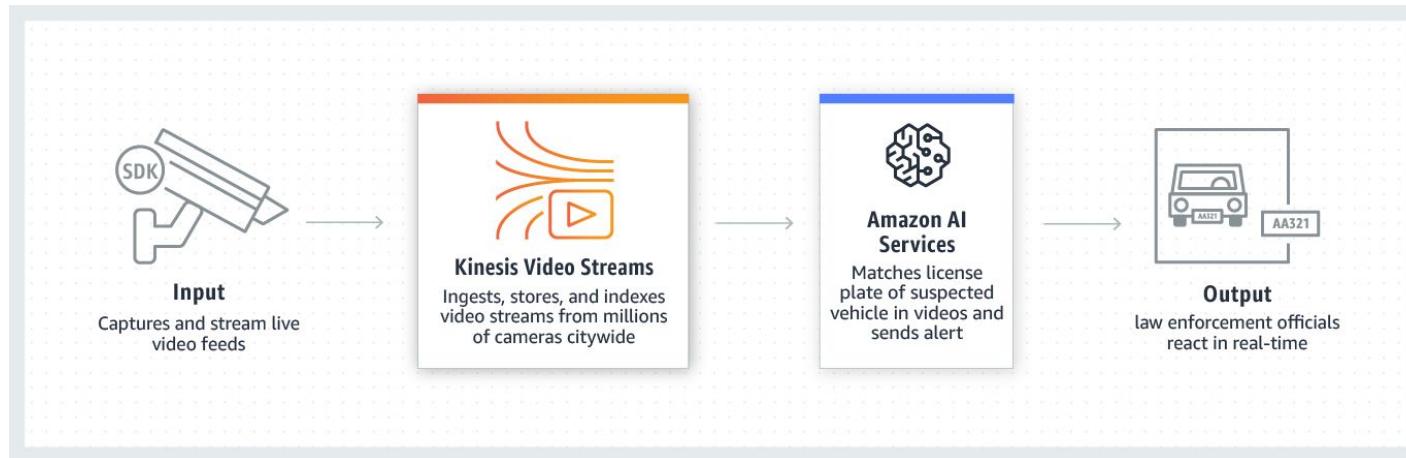
Kinesis Data Analytics

Kinesis Data Analytics has ability to analyze data streams in real time.



Kinesis Video Stream

Amazon Kinesis Video Streams makes it easy to securely stream video from connected devices to AWS

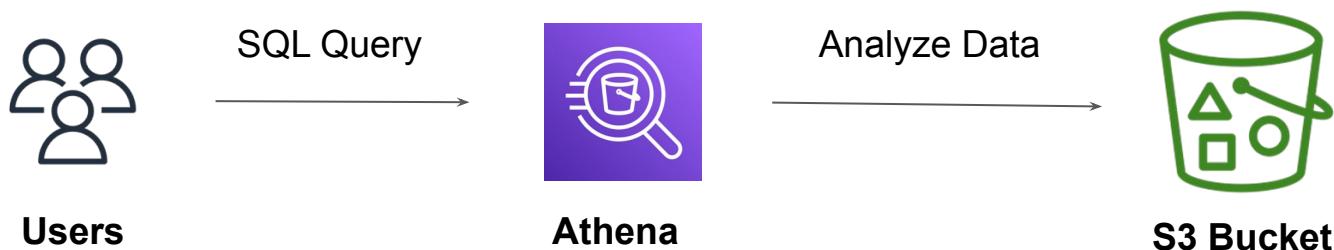


Amazon Athena

Query Logs from S3

Getting the basics right

Amazon Athena is service that allows us to analyze various log files from a data source using standard SQL



Approach Before Athena

You have CloudTrail logs in S3 and you want to see who has logged in, in the past 10 days.

- Create EC2 instances.
- Deploy monitoring stack like Splunk, ELK or others.
- Add the data source from S3 to import CloudTrail logs.
- Begin Analyzing.

Trusted Advisor

Recommendations are always good

What is Trusted Advisor ?

AWS Trusted Advisor analyzes your AWS environment and provides best practice recommendations in five major categories:

Cost Optimization



6 ✓ 3 ▲ 0 !

\$10.63

Potential monthly savings

Performance



10 ✓ 0 ▲

0 !

Security



11 ✓ 1 ▲

5 !

Fault Tolerance



13 ✓ 2 ▲

2 !

Service Limits



48 ✓ 0 ▲

0 !

Trusted Advisor Check Categories

Categories	Description
Cost optimization	Recommendations that can potentially save you money.
Performance	Recommendations that can improve the speed and responsiveness of your applications.
Security	Recommendations for security settings that can make your AWS solution more secure.
Fault tolerance	Recommendations that help increase the resiliency of your AWS solution.
Service limits	Checks the usage for your account and whether your account approaches or exceeds the limit for AWS services and resources.

AWS Config

Overview of Infrastructure Changes

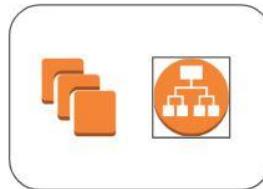
AWS Config - High Level Overview

AWS Config is primarily used to record the resource configuration changes over time.

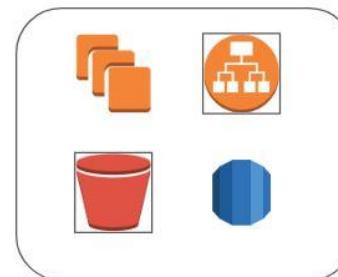
An EC2 instance was hosting website from past 90 days. Suddenly in last one week, there have been a lot of issues with the requests. What was changed?



Week 1



Week 2



Week 3

Audit and Compliance

AWS Config comes with large set of rules that can continuously monitor your AWS environment and report the findings.

Noncompliant rules by noncompliant resource count	
Name	Compliance
RootAccountHardwareMFAEnabled-conformance-pack-zcx0hyuom	⚠ 1 Noncompliant resource(s)
RootAccountMFAEnabled-conformance-pack-zcx0hyuom	⚠ 1 Noncompliant resource(s)
IAMPasswordPolicy-conformance-pack-zcx0hyuom	⚠ 1 Noncompliant resource(s)
approved-amis-by-id	⚠ 1 Noncompliant resource(s)
cloudtrail-security-trail-enabled	⚠ 1 Noncompliant resource(s)

[View all noncompliant rules](#)

Conformance Packs

A conformance pack is a collection of AWS Config rules and remediation actions that can be easily deployed

The screenshot shows the 'Deploy conformance pack' wizard in the AWS Config console. The current step is 'Step 1 Specify template'. A search bar at the top contains the text 'S'. Below it is a dropdown menu showing a list of operational best practices:

- Operational Best Practices for Amazon S3
- Operational Best Practices for Asset Management
- Operational Best Practices for BCP and DR
- Operational Best Practices for BNM RMIT
- Operational Best Practices for CCN ENS Low
- Operational Best Practices for CCN ENS Medium
- Operational Best Practices for CIS AWS v1_3 Level1
- Operational Best Practices for CIS AWS v1_3 Level2
- Operational Best Practices for CIS
- Operational Best Practices for CMMC Level 1
- Operational Best Practices for CMMC Level 2
- Operational Best Practices for Compute Services
- Operational Best Practices for Data Resiliency
- Operational Best Practices for Amazon S3

To the right of the dropdown, there is descriptive text about AWS accounts and a note about creating your own template. At the bottom, there is a link to 'Conformance Pack Sample Templates' and two buttons: 'Cancel' and 'Next'.

Pricing of AWS Config

You pay \$0.003 per configuration item recorded in your AWS account per AWS Region. A configuration item is recorded whenever a resource undergoes a configuration change or a relationship change.

Based on rule evaluation. A rule evaluation is recorded every time a resource is evaluated for compliance against an AWS Config rule.

You are charged per conformance pack evaluation in your AWS account per AWS Region based on the tier below.

AWS Config Aggregator



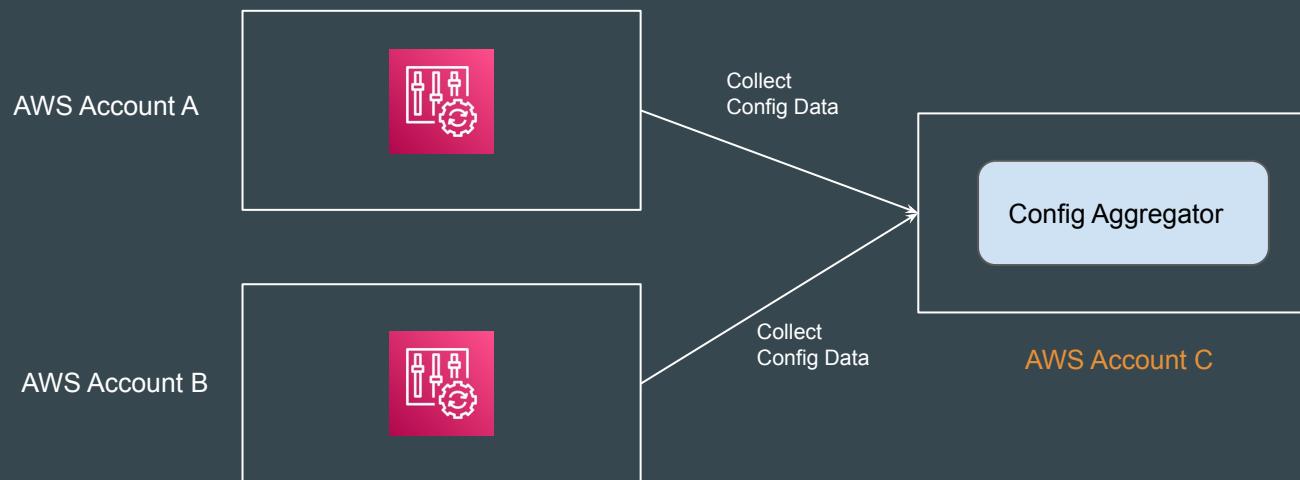
Understanding the Basics

An **aggregator** is an AWS Config resource type that collects AWS Config configuration and compliance data from the following:

1. Multiple accounts and multiple regions.
2. Single account and multiple regions.
3. An organization in AWS Organizations and all the accounts in that organization which have AWS Config enabled.

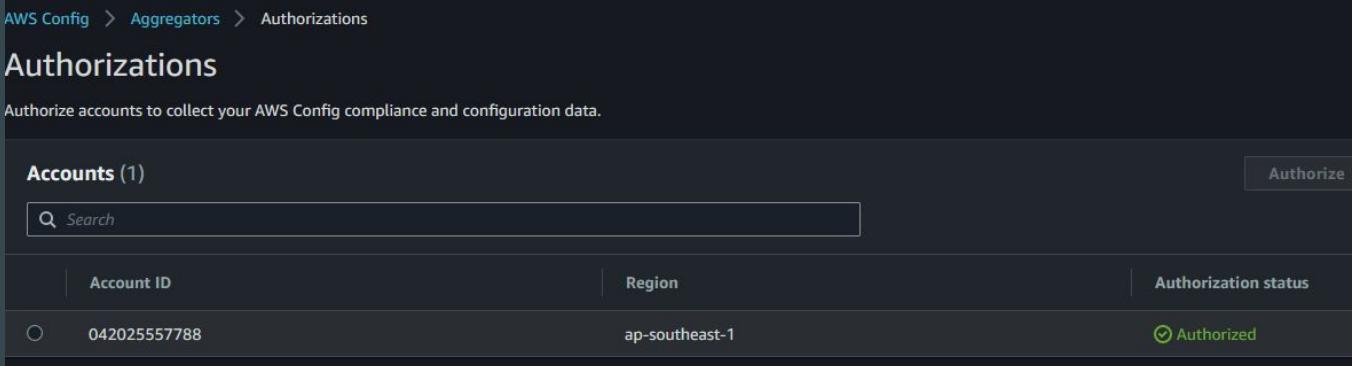
Understanding the Workflow

Config Aggregator can collect Config Data from multiple external accounts.



Important Step - Authorization

In the external accounts, you need to allow a specified aggregator account and Region to collect AWS Config configuration and compliance data from your current account.



The screenshot shows the AWS Config Authorizations page. The navigation bar at the top includes links for AWS Config, Aggregators, and Authorizations. The main title is "Authorizations" with the subtitle "Authorize accounts to collect your AWS Config compliance and configuration data." Below this, there is a section titled "Accounts (1)" with a search bar. A single account is listed in the table:

Account ID	Region	Authorization status
042025557788	ap-southeast-1	Authorized

External AWS Account

Remediate Non-Compliant Config Rules with SSM Automation



Setting the Base Right

AWS Config Rules can be created to audit the compliance of your environment.

Automation, a capability of AWS System manager, simplifies common maintenance, deployment, and remediation tasks for AWS services like Amazon EC2, RDS,S3 and many more.



Let's be Friends!

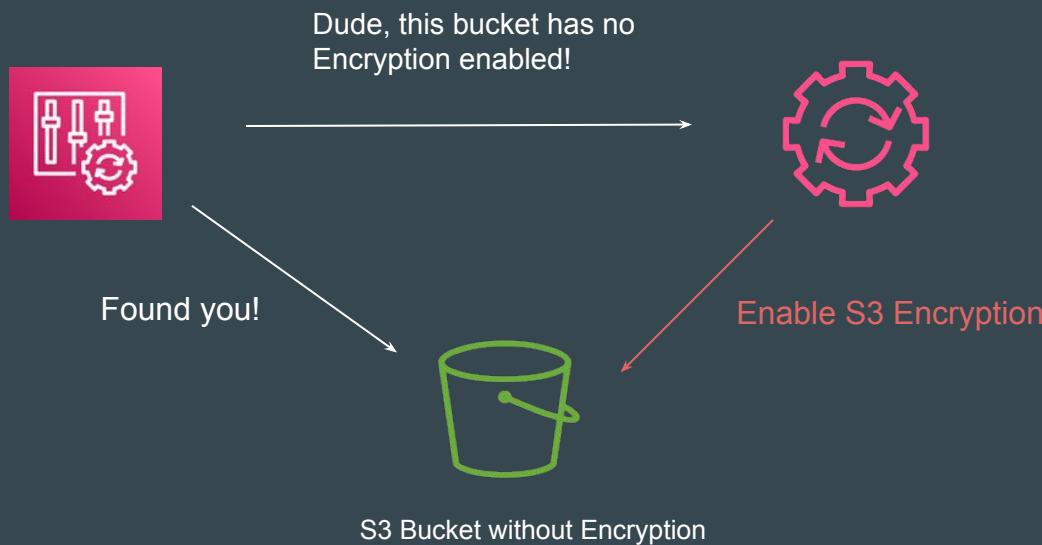


AWS Config

SSM Automation

Simple Use-Case

AWS Config Rule has identified that a specific S3 bucket does not have S3 Encryption enabled.



Point to Note

Automations can be initiated under the context of a service role (or assume role).

This allows the service to perform actions on your behalf.

Point to Note

AWS Config > Rules > restricted-ssh > Manage remediation

Edit: Remediation action

▼ Select remediation method

Automatic remediation
The remediation action gets triggered automatically when the resources in scope become noncompliant.

Manual remediation
You have to manually choose to remediate the noncompliant resources.

If a resource is still non-compliant after auto-remediation, you can set this rule to try again. Note, there are costs associated with running a remediation script.

Retries in Seconds

5 60

▼ Remediation action details

The execution of remediation actions is achieved using AWS Systems Manager Automation

Choose remediation action

AWS-DisablePublicAccessForSecurityGroup ▾

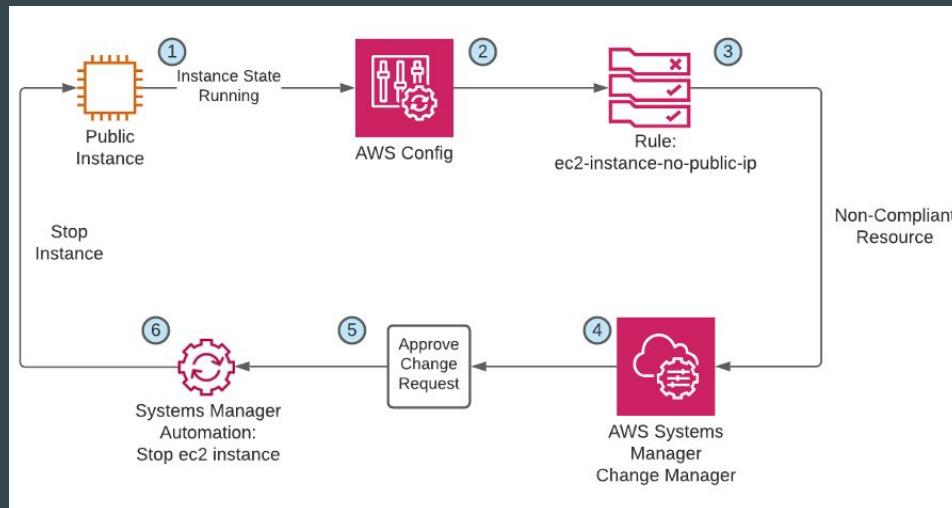
Remediation Action in AWS Config

Automations can be initiated under the context of a service role (or assume role).

This allows the service to perform actions on your behalf.

Architecture with SSM Change Manager

Change Manager, a capability of AWS Systems Manager for requesting, approving, implementing, and reporting on operational changes to your application configuration and infrastructure



Amazon CodeGuru



Understanding the Challenge

Development code **can contain wide variety of issues** that needs to be addressed and optimized.

Code Blame 12 lines (10 loc) · 311 Bytes

```
1 package com.main;
2
3 ✓ public class Main {
4     public Main() {
5         configureApp();
6     }
7
8     private void configureApp() {
9         GoSellSDK.init(this, "sk_test_kovrMB0mupFJXfNZWx6Etg5y", "company.tap.goSellSDKExample"); // to be replaced by merchant
10        GoSellSDK.setLocale("en");// language to be set by merchant
11    }
12 }
```



Sample Code

What is Needed

Customers **need tools** that can scan the code from repository and **quickly identify the issues** so that they can be addressed in development stage itself.



Looks like there is
hardcoded secret here!

Security Guy



Code Blame 12 lines (10 loc) - 311 Bytes

```
1 package com.main;
2
3 public class Main {
4     public Main() {
5         configureApp();
6     }
7
8     private void configureApp() {
9         GoSellSDK.init(this, "sk_test_kovrMB0mupFJXFNZwx6Etg5y", "company.tap.goSellSDKExample"); // to be replaced by merchant
10        GoSellSDK.setLocale("en");// language to be set by merchant
11    }
12 }
```

Sample Code

Setting the Base

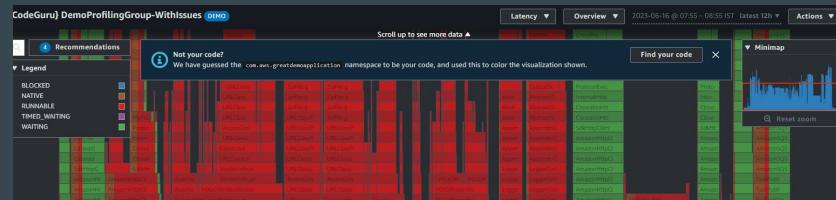
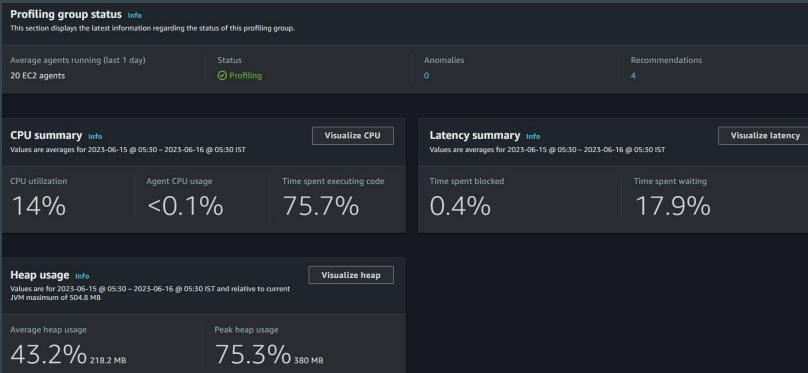
Amazon CodeGuru provides set of tools to **improve application code security, quality, and performance with ML**

CodeGuru Features	Description
CodeGuru Security	Identify Vulnerabilities in Code & Provide Recommendations
CodeGuru Profiler	Visualize & Provide Recommendation on App Performance.
CodeGuru Reviewer	Evaluates Code Against Best Practices

CodeGuru Profiler

CodeGuru Profiler **visualizes your application performance**, showing you the methods that take the most time and CPU capacity to execute.

This helps you diagnose and isolate root causes of application issues during operational events much faster.



CodeGuru Reviewer

CodeGuru Reviewer connects to code repositories such as GitHub, AWS CodeCommit and Bitbucket.

It evaluates your code against best practices observed in popular open source code repositories and Amazon's own code base

The screenshot shows the CodeGuru Reviewer interface with the title "Recommendations (4)". It displays two separate findings for different lines of code:

- src/resources/application.conf Line: 1**:
A hardcoded Database Connection String is identified as a security risk. It advises revoking access to resources using this credential and storing future credentials in a management service like AWS Secrets Manager. A link to "Learn more about the use of hardcoded credentials" is provided.
- src/resources/application.conf Line: 5**:
A hardcoded Stripe Live Secret Key is identified as a security risk. Similar advice is given regarding AWS Secrets Manager and a link to learn more about hardcoded credentials.

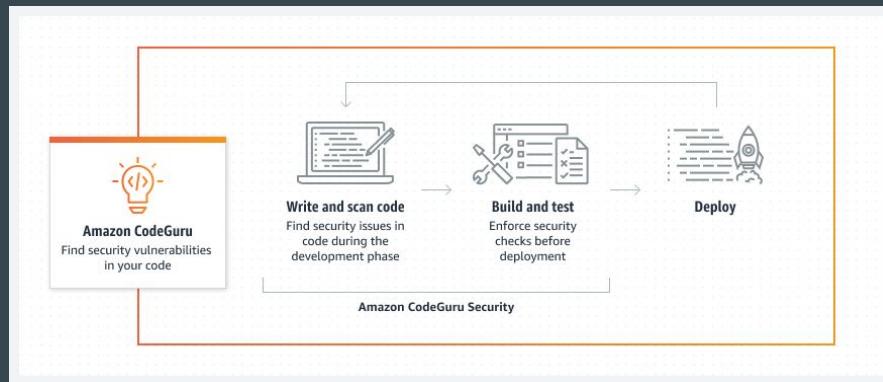
The interface includes a search bar, navigation controls (back, forward, first, last), and a refresh button.

CodeGuru Security

CodeGuru Security is an ML and program analysis-based code scanning tool that **finds security vulnerabilities** in your Java, Python, and JavaScript code.

CodeGuru Security detects OWASP Top 10 issues and many others.

CodeGuru Security is a static application security testing (**SAST**) tool.



Intro to Amazon SQS

Message Queuing Service

Use-Case: Restoring Image Application

Medium Corp is designing an application that will enhance and restore the images that users submit through the online portal.



Current Architecture

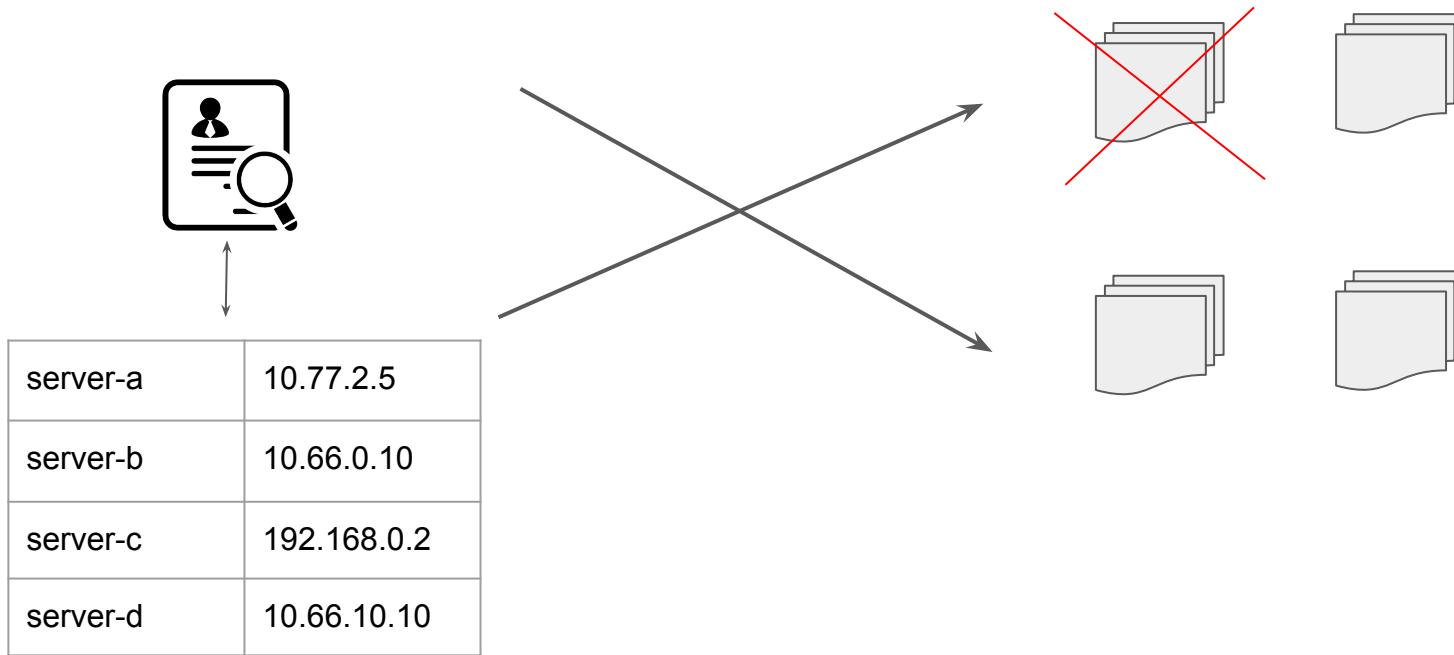
The overall architecture involves two components:

1. Image Gatherer - Takes the Images from the user via Upload button.
2. Imager Enhancer - Receives the Image from Image Gatherer.



Challenges

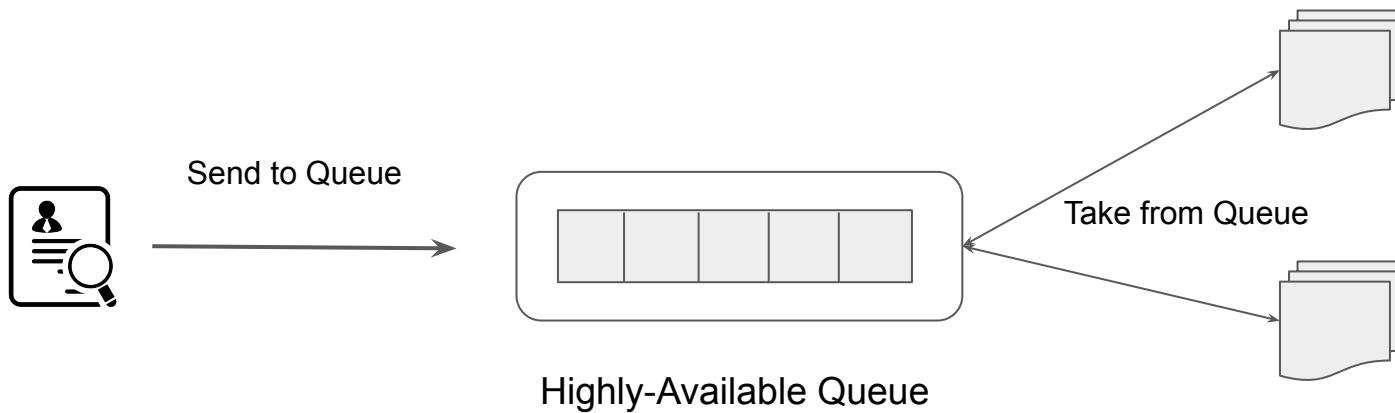
Due to popularity of the application and huge traffic spike, Medium Corp has decided to add more image enhancer servers.



Better Architecture

One of the main function of message queue service is to take message from a Publisher and forward that to a consumer.

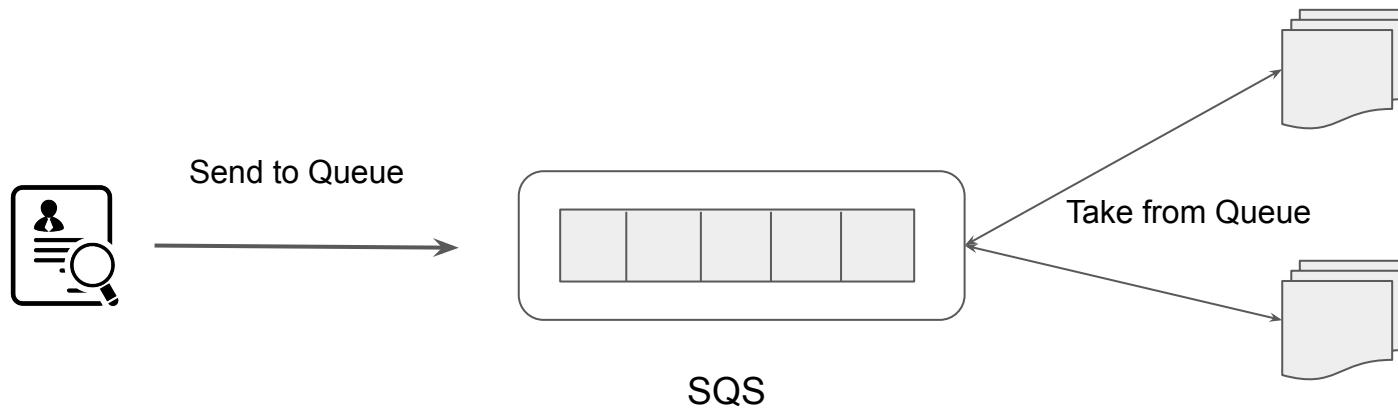
The queue stores these messages internally.



Introduction to SQS

Amazon SQS is a fast reliable, scalable, and fully managed message queuing service.

Amazon SQS makes it simple and quiet cost effective to decouple the components of a specific application.



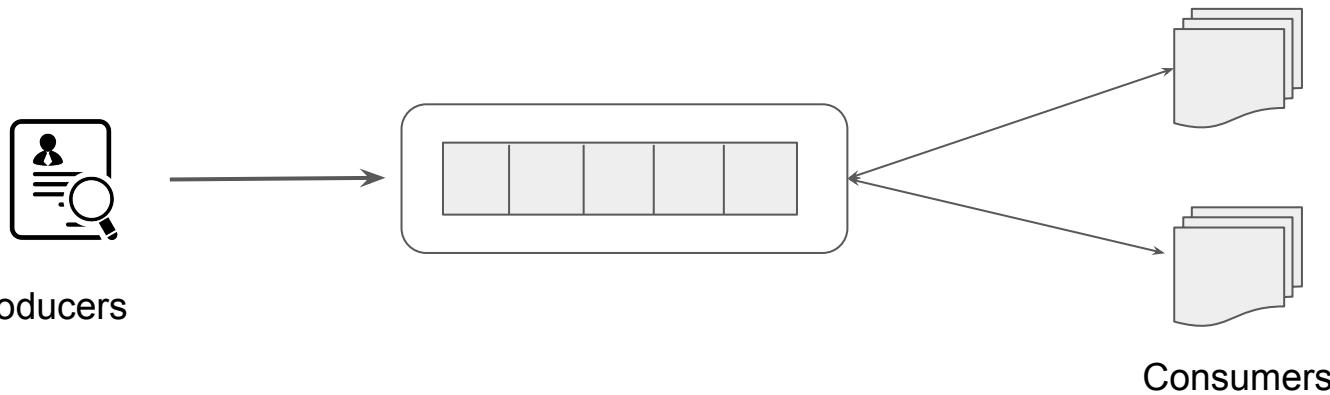
Tightly Coupled Systems

Components of system architecture directly communicate with each other and have hard-dependency on each other.



Loosely Coupled System

Components of system architecture that can process the information without being directly connected.

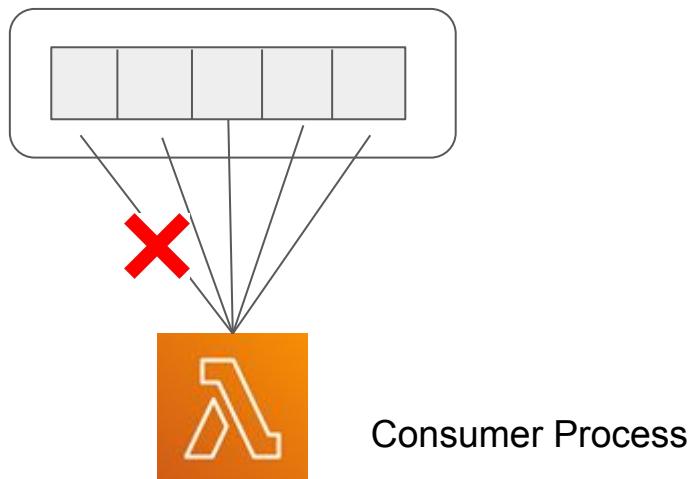


SQS Dead-Letter Queues

Troubleshooting Problematic messages

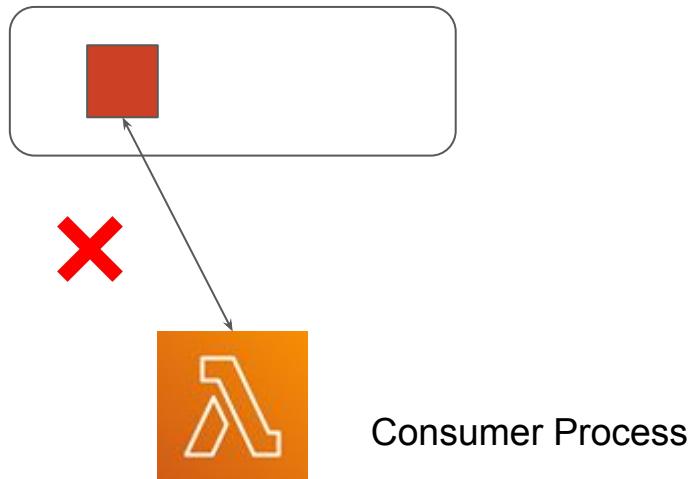
Understanding the Challenge

Amazon SQS supports dead-letter queues, which other queues (source queues) can target for messages that can't be processed (consumed) successfully.



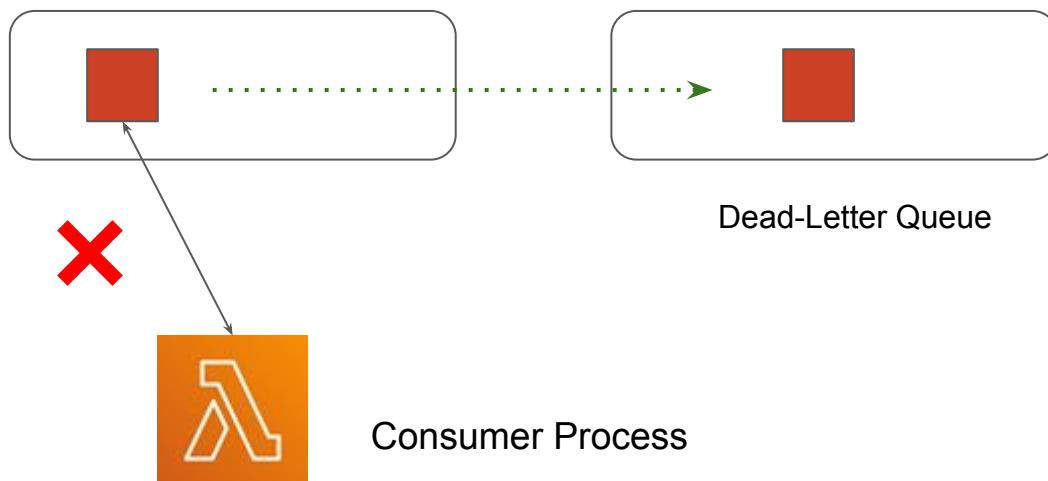
Understanding the Challenge

Amazon SQS supports dead-letter queues, which other queues (source queues) can target for messages that can't be processed (consumed) successfully.



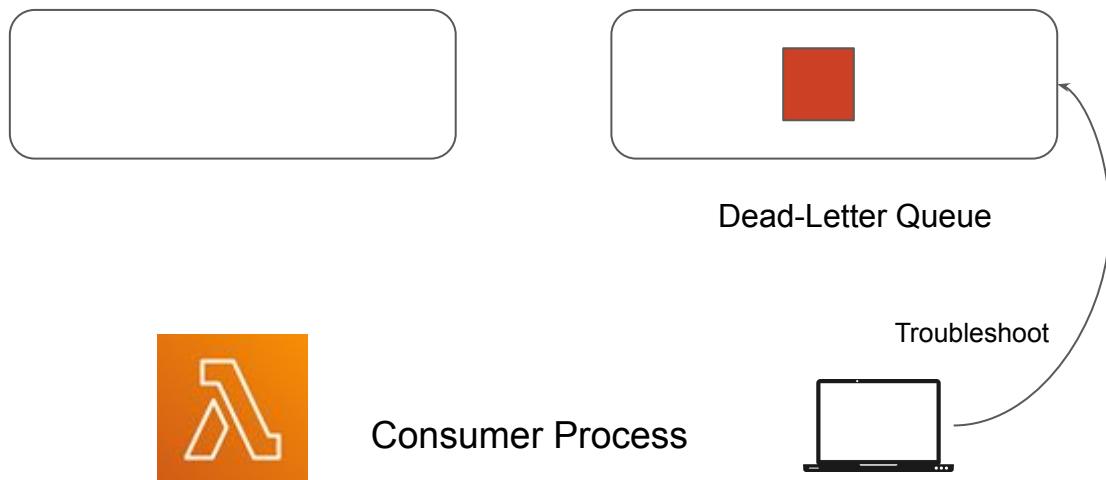
Moving to Dead-Letter Queue

Move the message that cannot be processed to dead letter queue.



Moving to Dead-Letter Queue

Move the message that cannot be processed to dead letter queue.



Overview of Dead Letter Queue

Amazon SQS supports dead-letter queues, which other queues (source queues) can target for messages that can't be processed (consumed) successfully

Dead-letter queues are useful for debugging your application or messaging system because they let you isolate problematic messages to determine why their processing doesn't succeed.

The messages are sent to the dead letter queue after exceeding maximum receives.

Important Pointers to Remember

When a message moves to a dead-letter queue, the timestamp remains unchanged.

Let's understand this with an example:

- Message has been in the source queue for 1 day and moved to dead-letter queue.
- Message Retention Period in Dead Letter Queue is 4 days.
- Message will be deleted from the Dead Letter queue after 3 days.

Best practice is to have higher retention period for dead-letter queues than the source queue.

Relax and Have a Meme Before Proceeding

me: i'll do it at 6

time: 6:05

me: wow looks like i gotta wait til 7 now

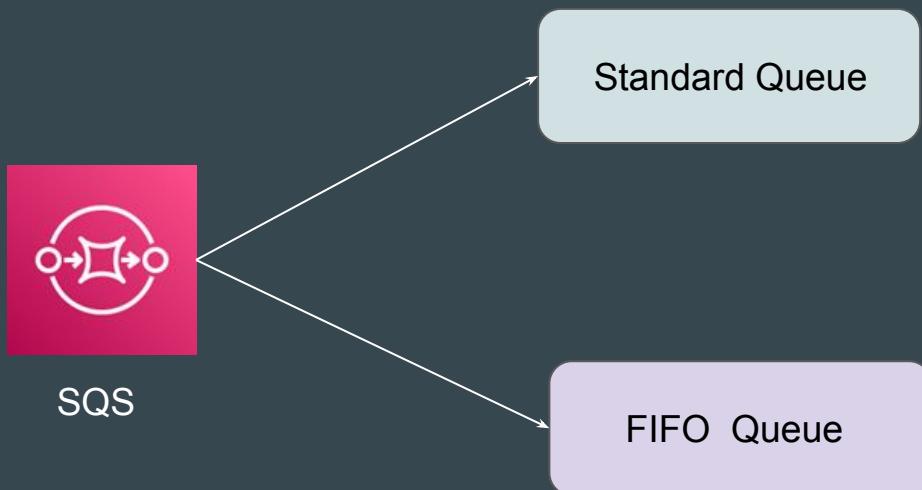


Amazon SQS queue types



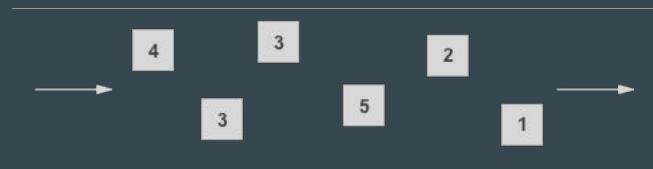
Types of SQS Queue

There are two primary types of SQS queues.



Message Ordering

Standard Queue	Occasionally, messages are delivered in an order different from which they were sent.
FIFO Queue	The order in which messages are sent and received is strictly preserved



Standard Queue



FIFO Queue

Difference

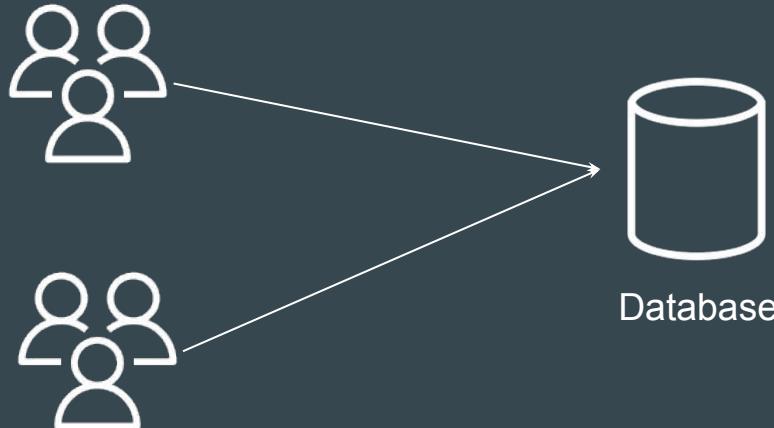
Characteristic	Standard Queue	FIFO Queue
Throughput	Standard queues support a nearly unlimited number of API calls per second, per API action (SendMessage, ReceiveMessage, or DeleteMessage).	FIFO queues support up to 300 API calls per second, per API method (SendMessage, ReceiveMessage, or DeleteMessage).
Delivery	A message is delivered at least once, but occasionally more than one copy of a message is delivered.	A message is delivered once and remains available until a consumer processes and deletes it. Duplicates aren't introduced into the queue.
Ordering	Messages can be out of order.	Messages are in Order.

Message Queues in Database Transactions



Understanding with a Use-Case

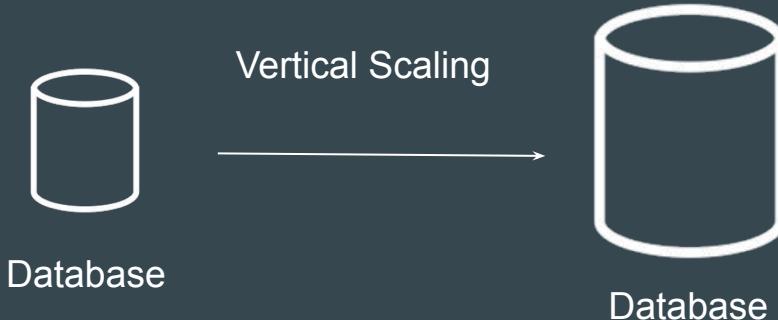
- Let's assume you have a single database hosted on RDS.
- Due to sales, the number of write transactions has reached 20x normal load.
- Many requests are failing regularly.
- New sale promotions are scheduled every alternate month.



Possible Solution - Vertical Scaling

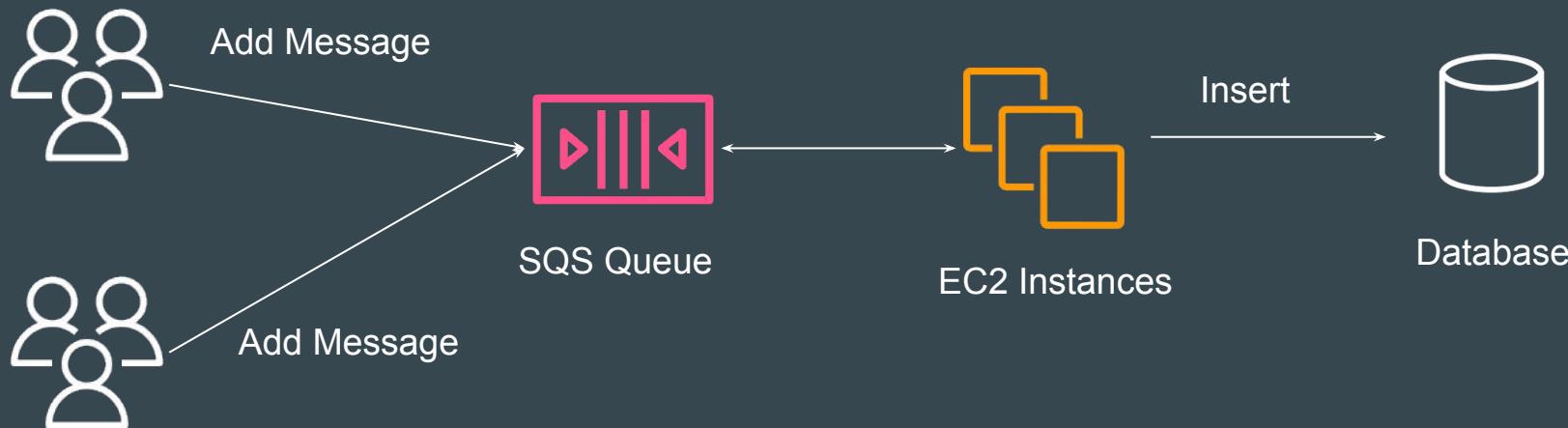
Increase the DB Instance Size of RDS and Provisioned IOPS to handle 20x capacity

Challenge: Downtime + Increased Cost

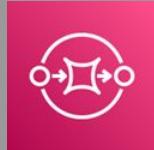


Better Approach - Add a Queue

In this approach, the messages are temporarily stored in SQS queue which can handle nearly infinite messages.



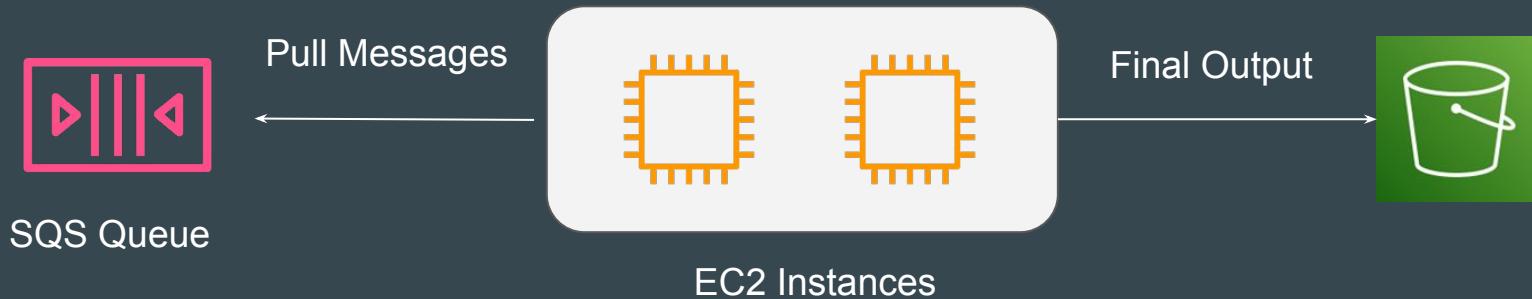
Scaling based on Amazon SQS



Setting the Base

In many scenarios, the number of EC2 instances that are required directly correlates with number of messages in SQS queue.

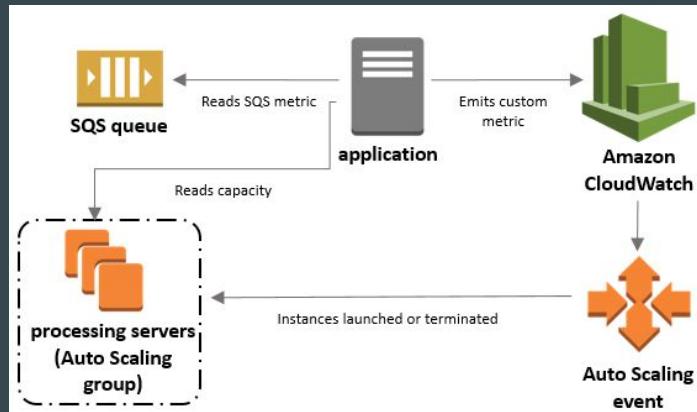
If number of messages increases in SQS, there would be a need to increase EC2 instances.



Scaling based on Amazon SQS

We can configure Auto-Scaling group to launch or terminate EC2 instances based on the number of messages in the SQS queue.

SQS Attribute to check: ApproximateNumberOfMessages



```
PS C:\Users\zealv> aws sqs get-queue-attributes --queue-url https://sqs.ap-southeast-1.amazonaws.com/693331494763/queue-1 --attribute-names ApproximateNumberofMessages --region ap-southeast-1
{
    "Attributes": {
        "ApproximateNumberOfMessages": "2"
    }
}
```

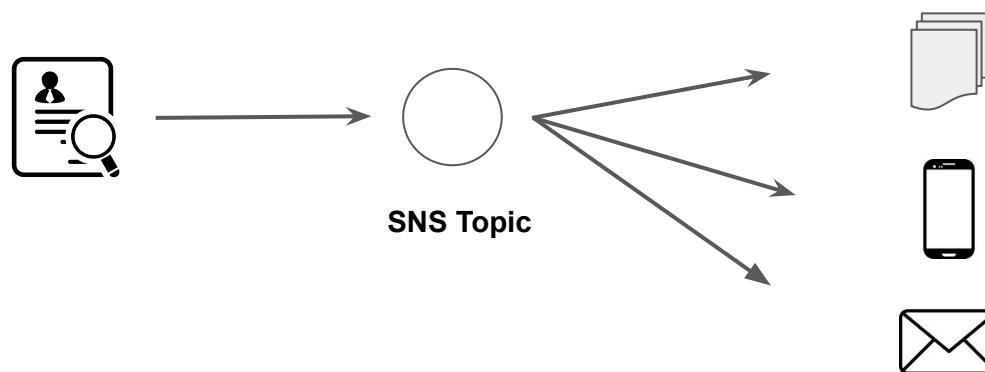
Simple Notification Service

Notification Service

Let's Message

SNS stands for simple notification service.

SNS is a fully managed messaging and mobile notification service for delivering messages to the subscribed endpoints.



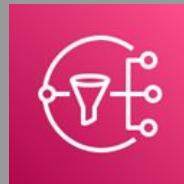
Use-Cases for SNS

AWS CloudWatch integrates well with SNS.

Whenever a disk usage of a server exceeds 95%, send an EMAIL and SMS notification to the NOC team.

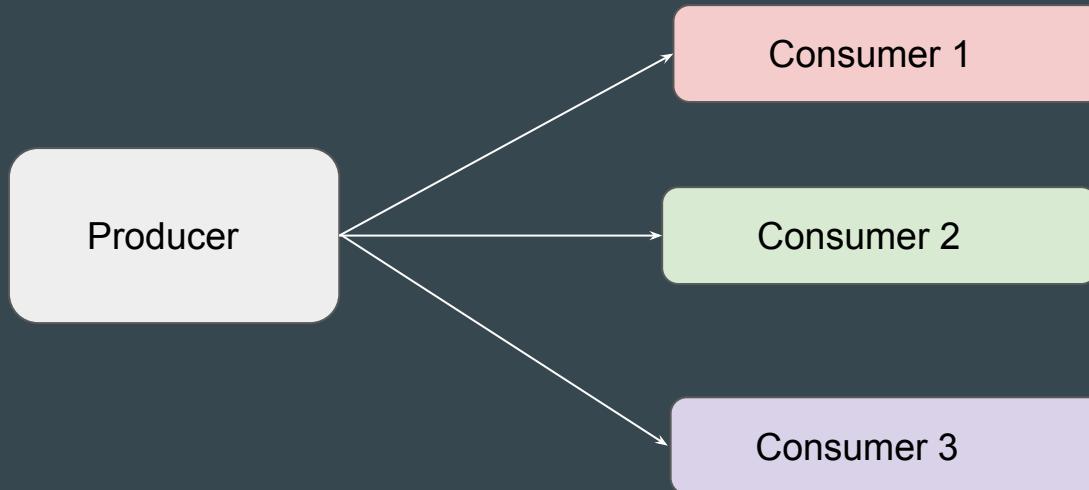
Whenever a server load in production is more than 90%, send and email and SMS notification.

SNS Fanout



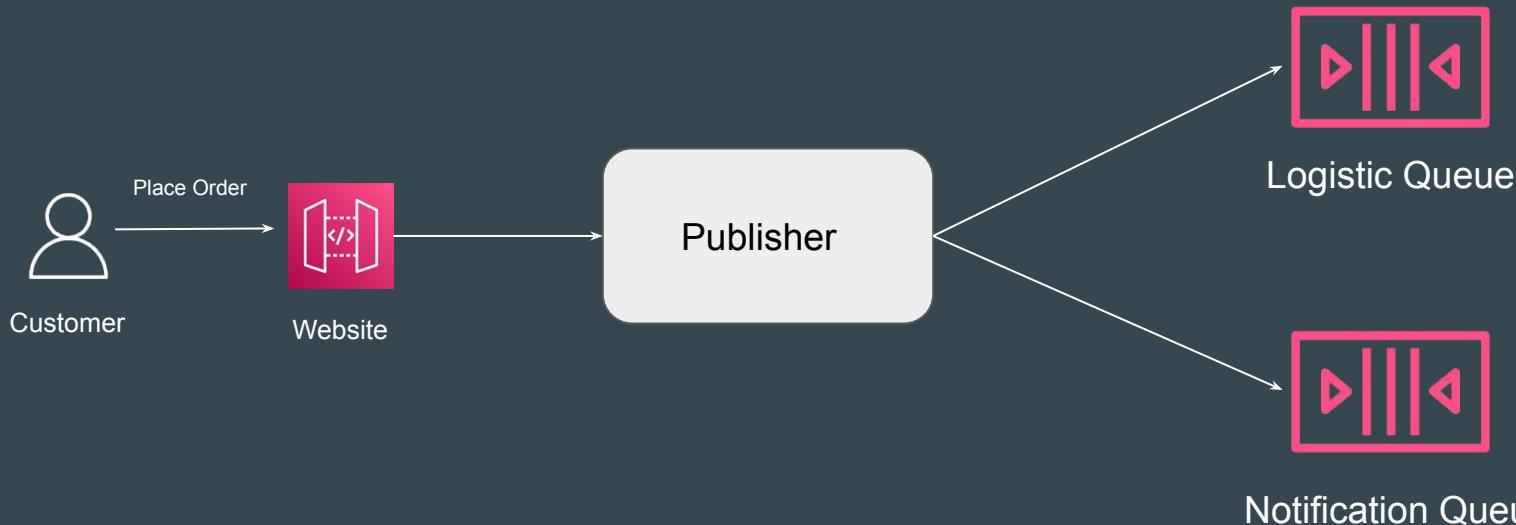
Basics of Fanout Pattern

Fanout is a pattern in which message is delivered to multiple destinations.



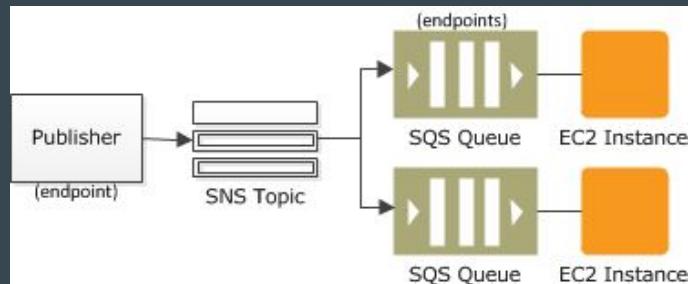
Simple Use-Case: Ordering a Product

Fanout is a pattern in which message is delivered to multiple destinations.



SNS Fanout Pattern

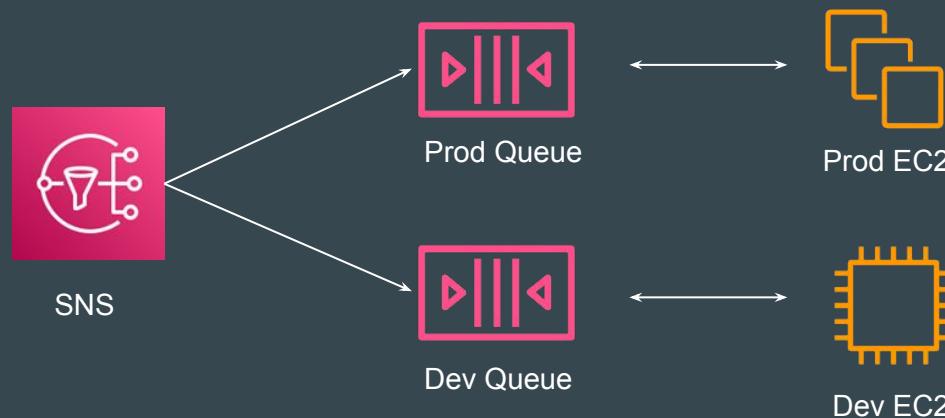
The Fanout scenario is when a message published to an SNS topic is replicated and pushed to multiple endpoints, such as Kinesis Data Firehose delivery streams, Amazon SQS queues, HTTP(S) endpoints, and Lambda functions.



Another Use-Case

You can also use fanout to replicate data sent to your production environment with your test environment

In production, you can attach a new SQS queue for test environment and can continue to improve and test your application using data received from your production environment.

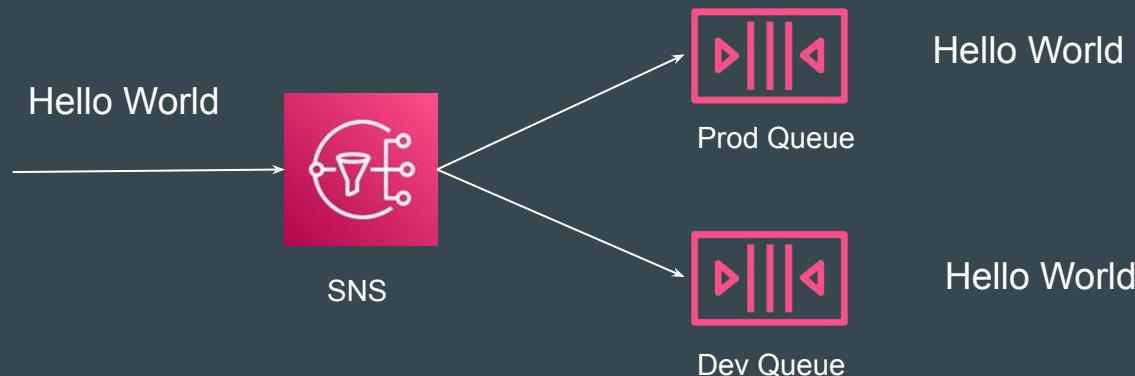


SNS Message Filtering



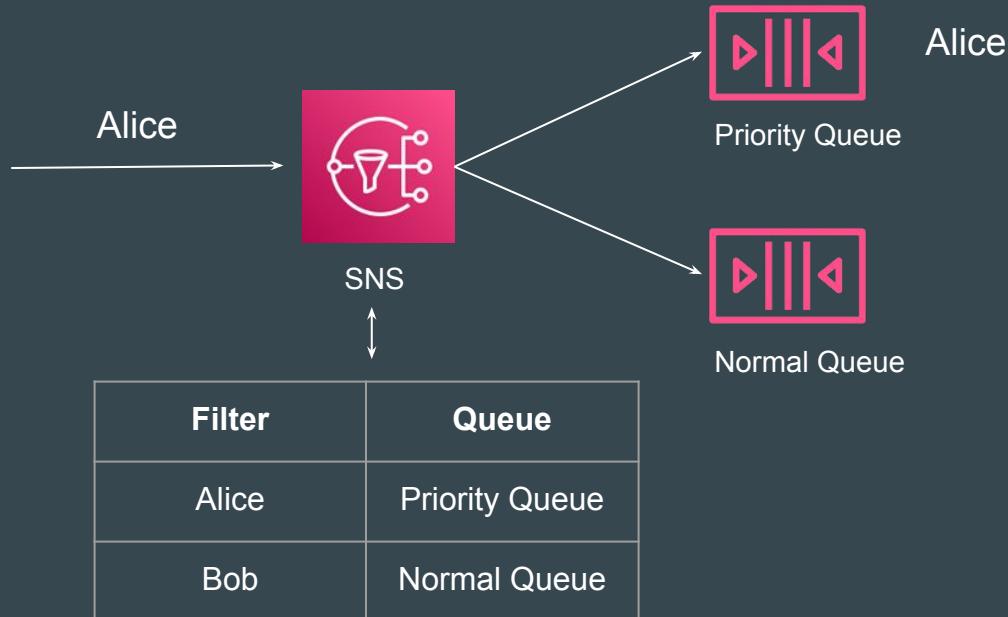
Revising the Basics

By default, an Amazon SNS topic subscriber receives every message that's published to the topic.



Basics of SNS Filtering

A filter policy is a JSON object containing properties that define which messages the subscriber receives

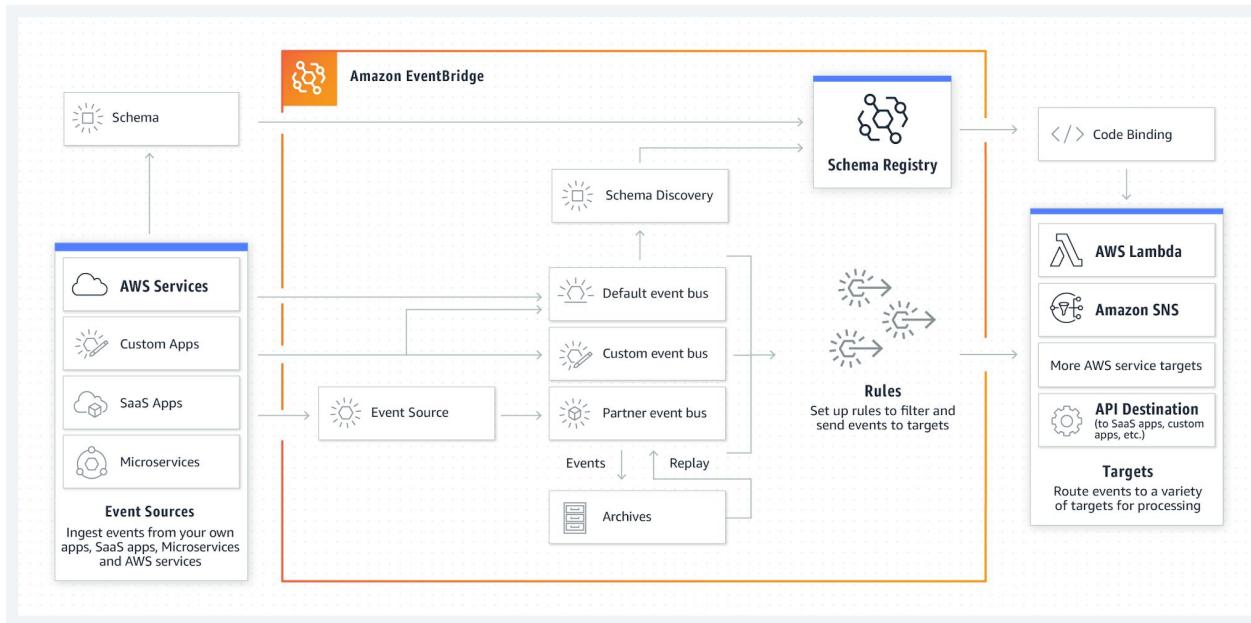


Amazon EventBridge

Connecting Services

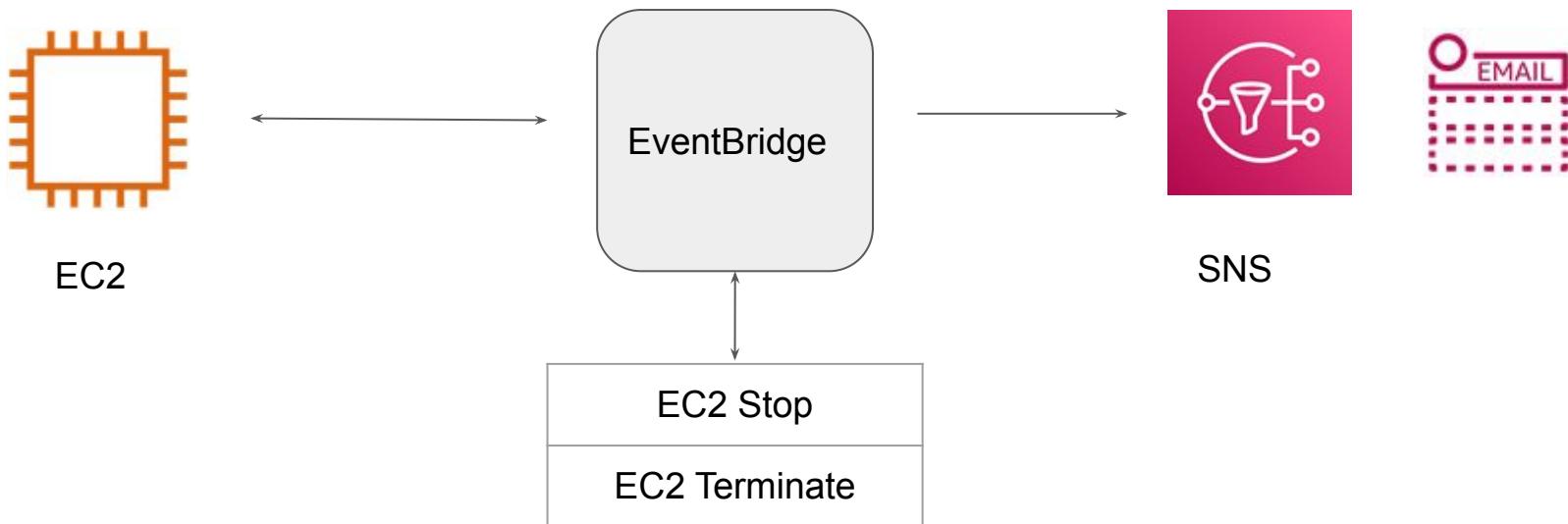
Overview of Amazon Event Bridge

EventBridge delivers a stream of real-time data from event sources to targets.



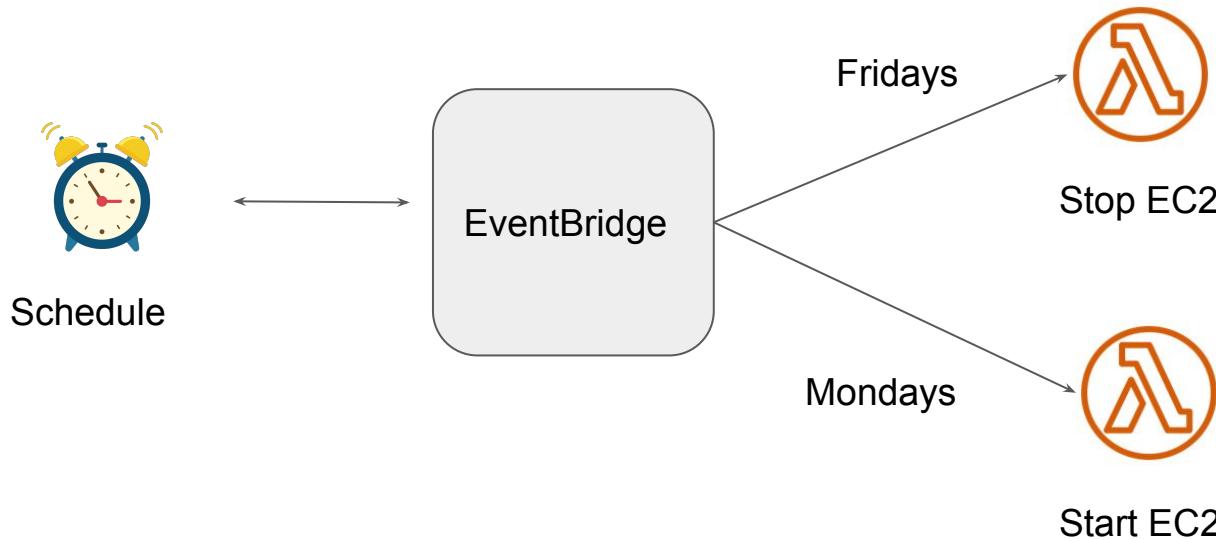
Use-Case 1: EC2 to SNS

Whenever a EC2 instance is stopped, Administrator should be notified.



Use-Case 2: Stop Dev EC2 Instances

Stop all DEV instances at 8PM on Fridays and Start at 9 AM on Mondays.



EC2 Auto-Recovery

Automated Recovery

Getting Started

You can create an Amazon CloudWatch alarm that monitors an Amazon EC2 instance and automatically recovers the instance if it becomes impaired due to an underlying hardware failure or a problem that requires AWS involvement to repair.

Examples of problems that cause system status checks to fail include:

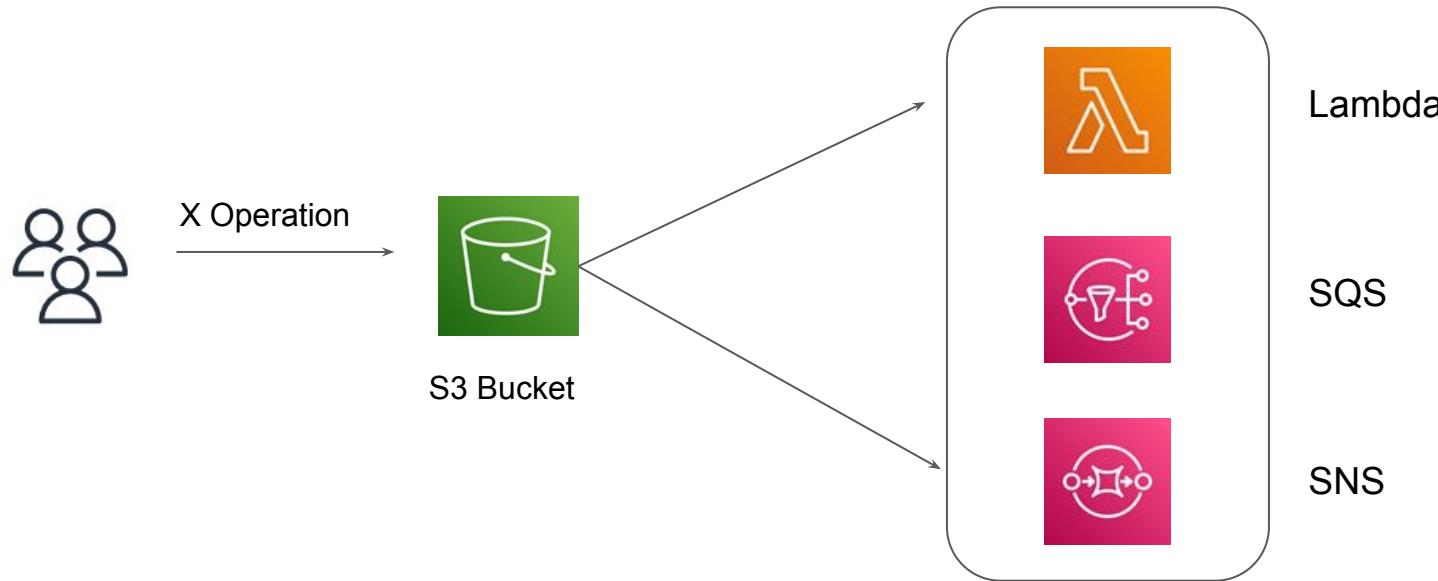
- Loss of network connectivity
- Loss of system power
- Software issues on the physical host
- Hardware issues on the physical host that impact network reachability

S3 Event Notification

S3 is more than just storage

Overview of S3 Event Notification

The Amazon S3 notification feature enables you to receive notifications when certain events happen in your bucket.



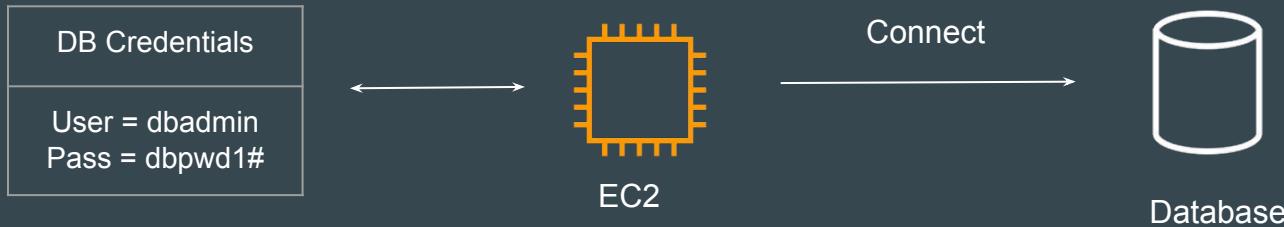
AWS Secrets Manager



Understanding the Challenge

In many organizations, secrets are hard coded directly as part of the application.

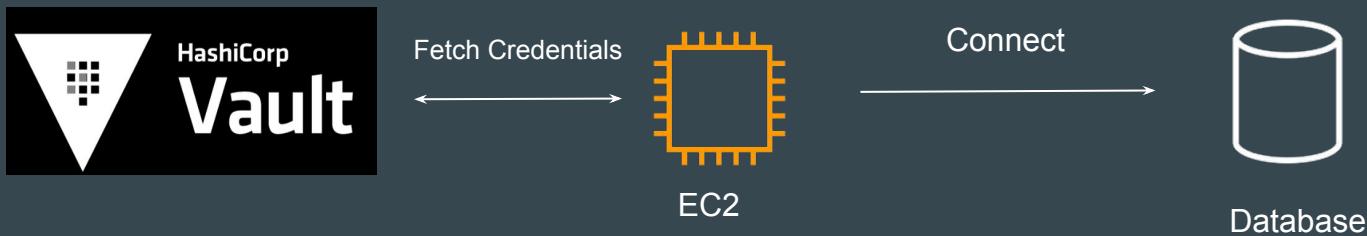
If you want to rotate the secret credential, all the application server needs to be updated. If you miss one, the production can go down.



Introducing Secrets Management

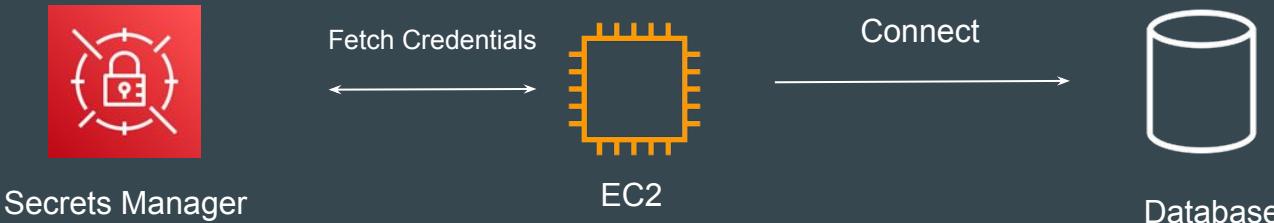
Secret management is a practice that allows developers to securely store sensitive data, such as passwords, keys, and tokens, in a secure environment with strict access controls.

Popular Tools: HashiCorp Vault, AWS Secrets Manager

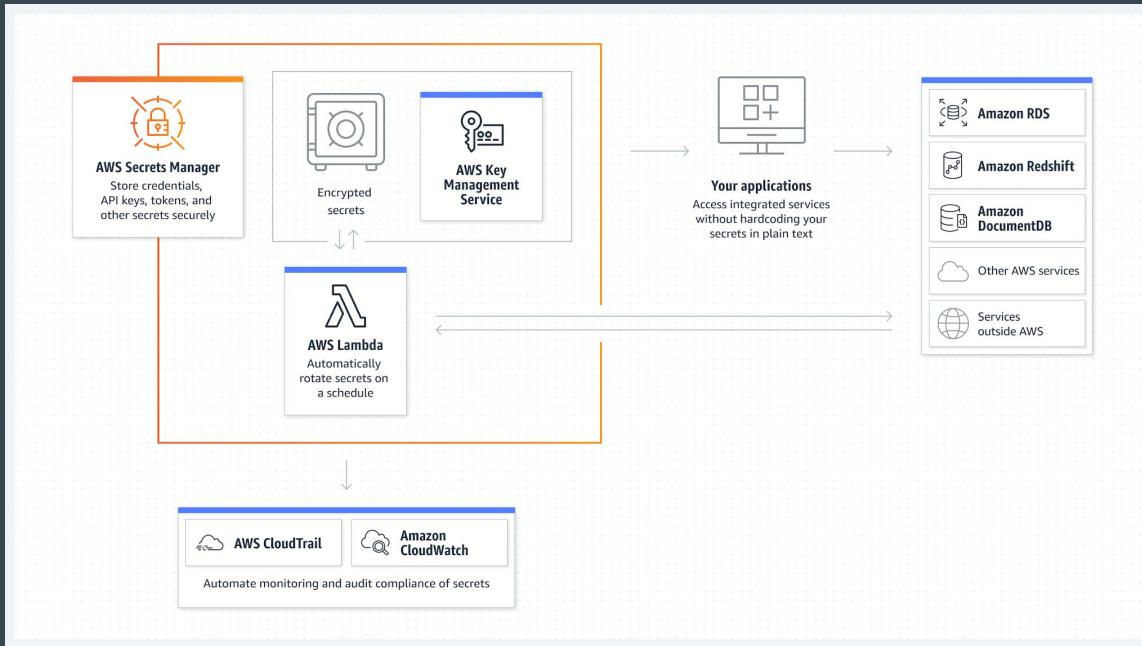


Introduction to Topic

AWS Secrets Manager helps you manage, retrieve, and rotate database credentials, API keys, and other secrets throughout their lifecycles.



Referenced from Docs



Rotate AWS Secrets Manager secrets

Rotation is the process of periodically updating a secret.

Secrets Manager rotation uses an AWS Lambda function to update the secret and the database.

To rotate a secret, Secrets Manager calls a Lambda function according to the schedule you set up. You can set a schedule to rotate after a period of time, for example, every 30 days.

Relax and Have a Meme Before Proceeding



Rotating Secrets



Basics of Rotation

Rotation is the process of periodically updating a secret.

Secrets Manager rotation uses an AWS Lambda function to update the secret and the database.



Points to Note

To rotate a secret, Secrets Manager calls a Lambda function according to the schedule you set up. You can set a schedule to rotate after a period of time, for example, every 30 days.

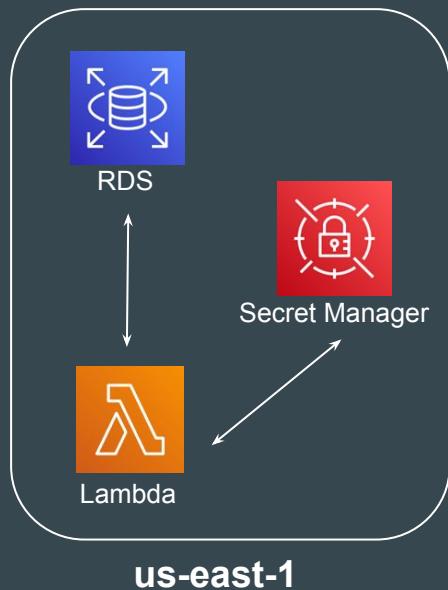
Secrets Manager provides rotation function templates for various use-cases related to RDS, DocumentDB, RedShift etc.

Replicate AWS Secrets Manager secrets



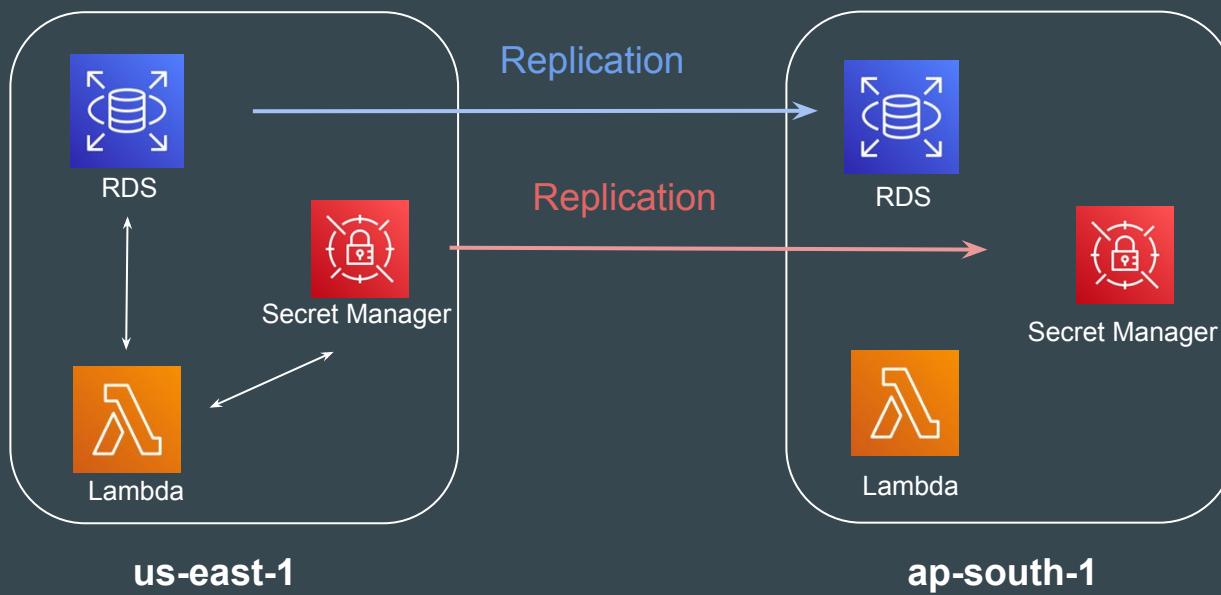
Understanding with Use-Case

In a Disaster Recovery based architecture, it is necessary to setup necessary level of replication across regions for failover.



Replicating Data Across Regions

In this architecture, the data and secrets are replicated across regions.



Points to Note

You can replicate your secrets in multiple AWS Regions to support applications spread across those Regions to meet Regional access and low latency requirements.

If you later need to, you can promote a replica secret to a standalone and then set it up for replication independently.

If you turn on rotation for your primary secret, Secrets Manager rotates the secret in the primary Region, and the new secret value propagates to all of the associated replica secrets.

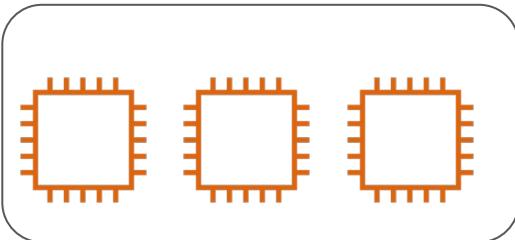
Network ACL

Multiple Layers for Defense

Understanding the Basics

A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.

- Security Group works at an EC2 instance level.
- Network ACL works at a Subnet Level.



Security Group

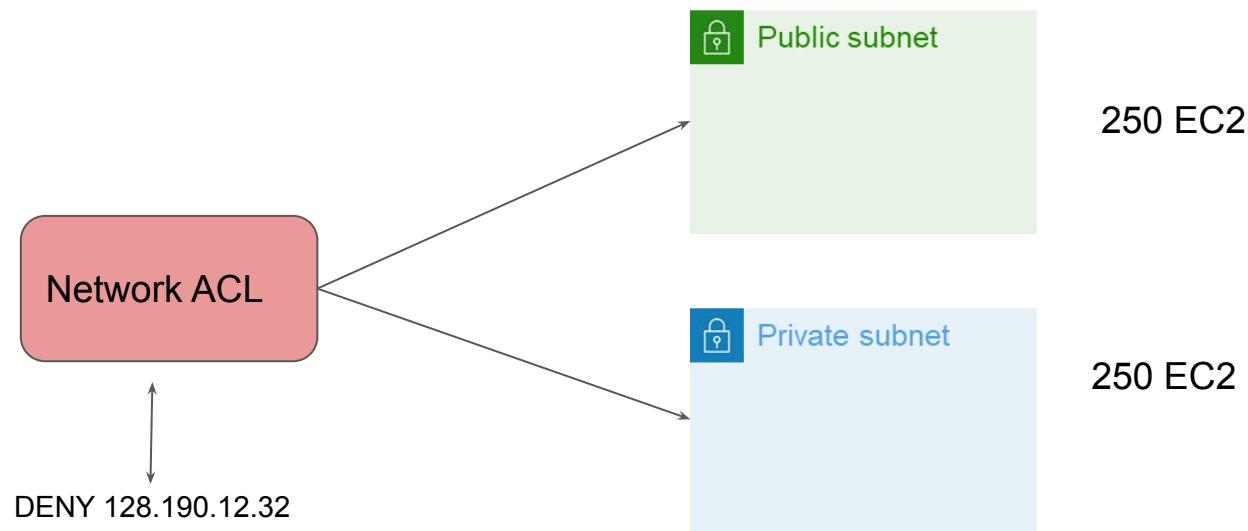


Network ACL

Understanding with Use-Case

Company XYZ is getting **lot of attacks** from a random IP **128.190.12.32**. The company has more than 500 servers and Security team decided to block that IP in firewall for all the servers.

How to go ahead and achieve that goal ?



Important Pointers

Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.

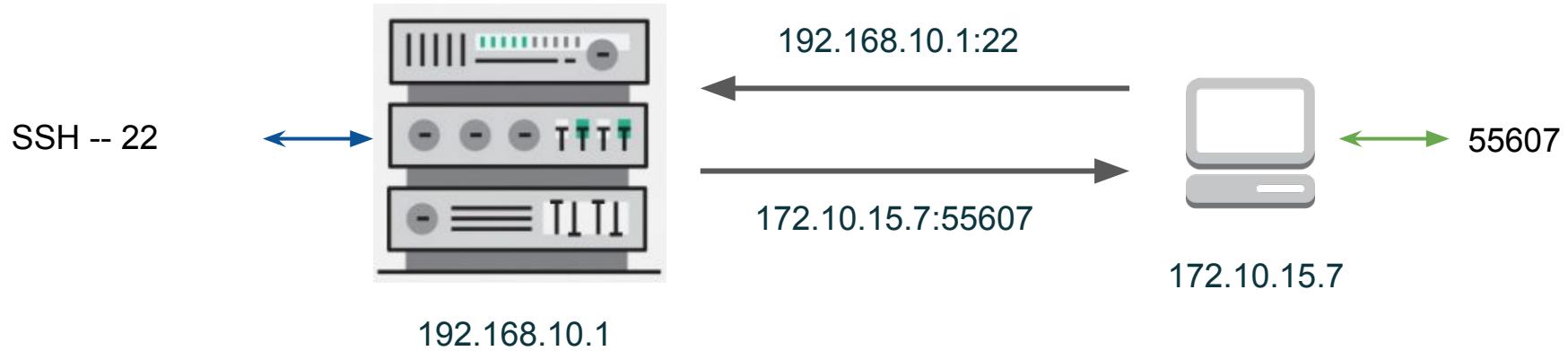
Default NACL allows all inbound and outbound IPv4 traffic and, if applicable, IPv6 traffic.

You can associate a network ACL with multiple subnets. However, a subnet can be associated with only one network ACL at a time.

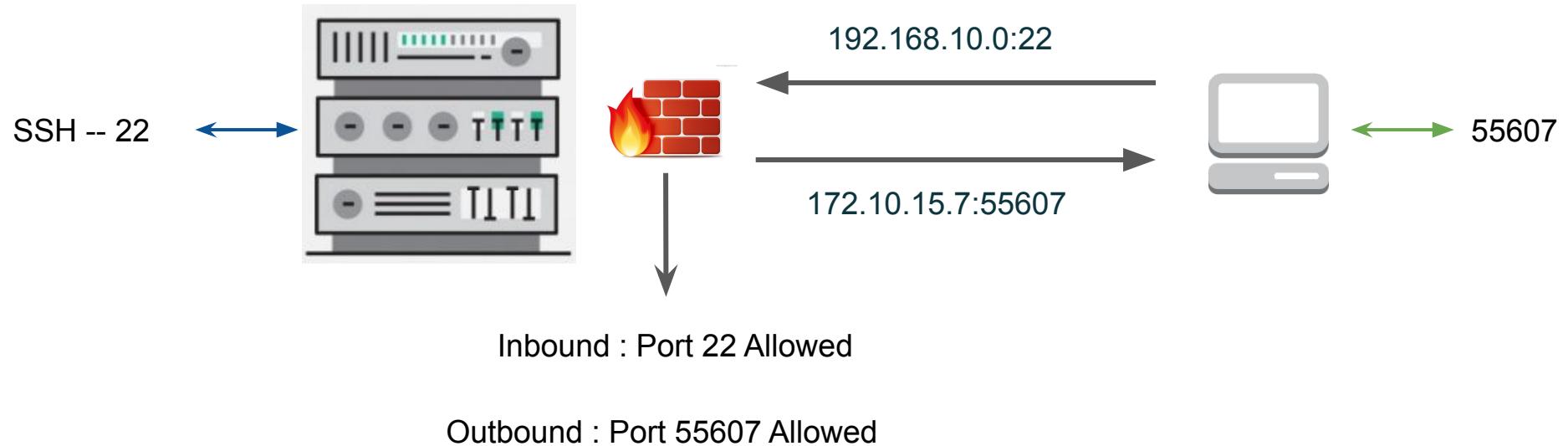
Stateful vs Stateless Firewalls

2 types of Firewall

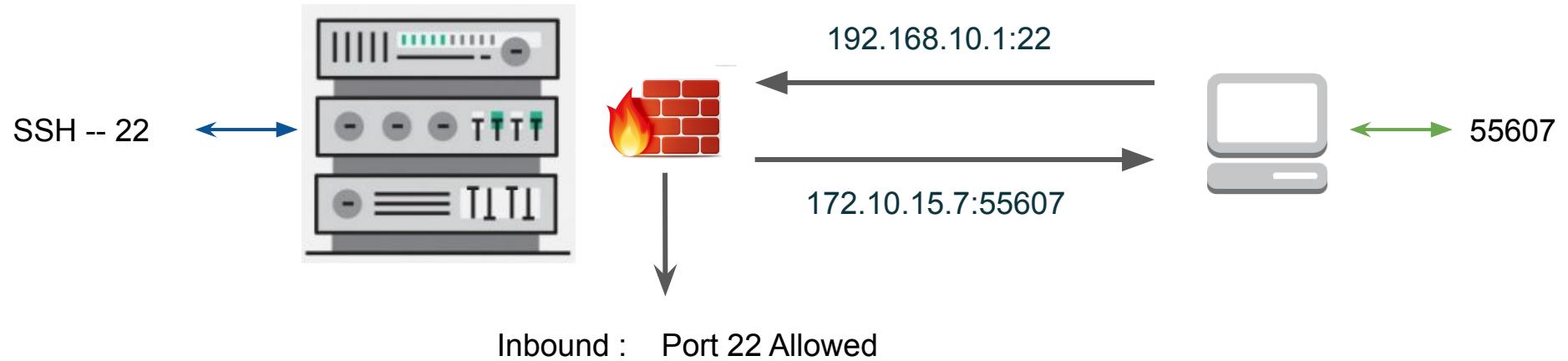
Basic TCP/IP Communication



When Stateless Firewall is Involved



Stateful Firewall



The Finale

There are 2 main types of Firewall :-

- Stateful Firewall
- Stateless Firewall

Stateful firewall maintains the connection state and knows which packets to allow Outbound even when outbound is restricted.

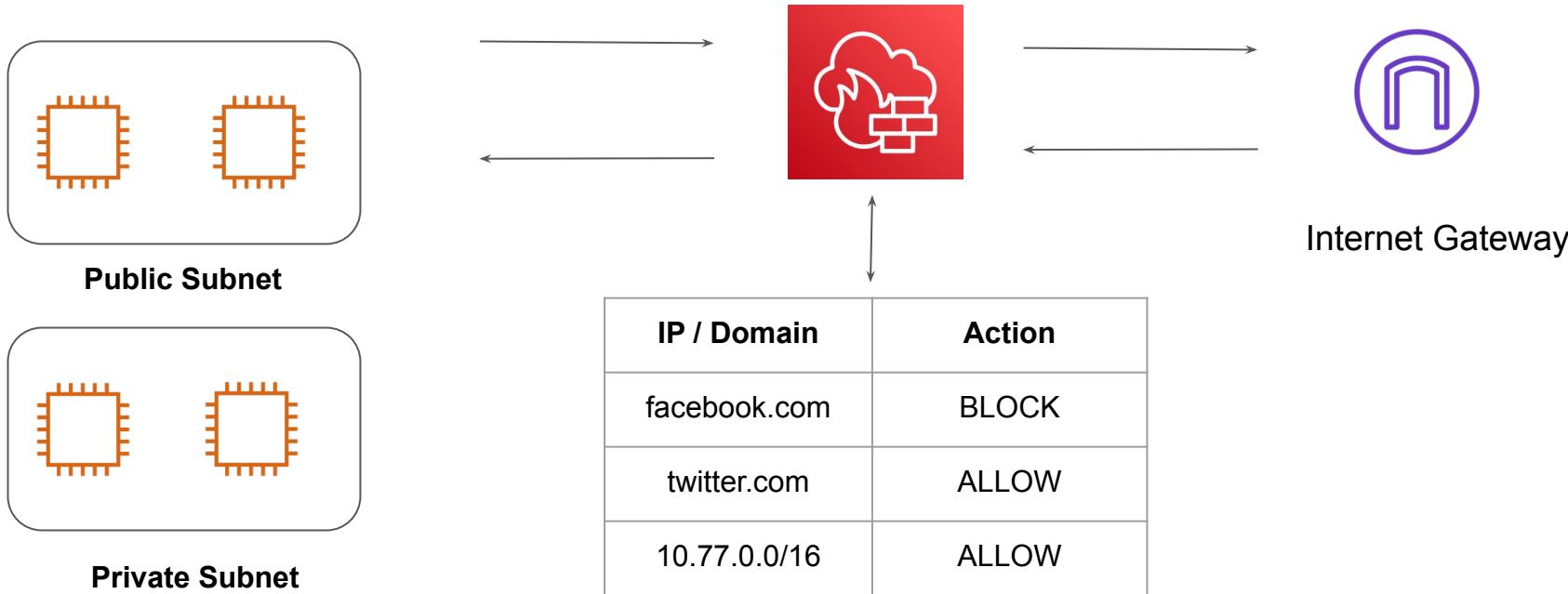
Stateless firewall does not maintain the connection state and for them each packet traversing inbound or outbound is a new separate packet.

AWS Network Firewall

Yet Another Firewall

Basics of Network Firewall

AWS Network Firewall is a stateful, managed, network firewall and intrusion detection and prevention service for your virtual private cloud (VPC)



Benefits of Network Firewall

You can use Network Firewall to monitor and protect your Amazon VPC traffic in a number of ways, including the following:

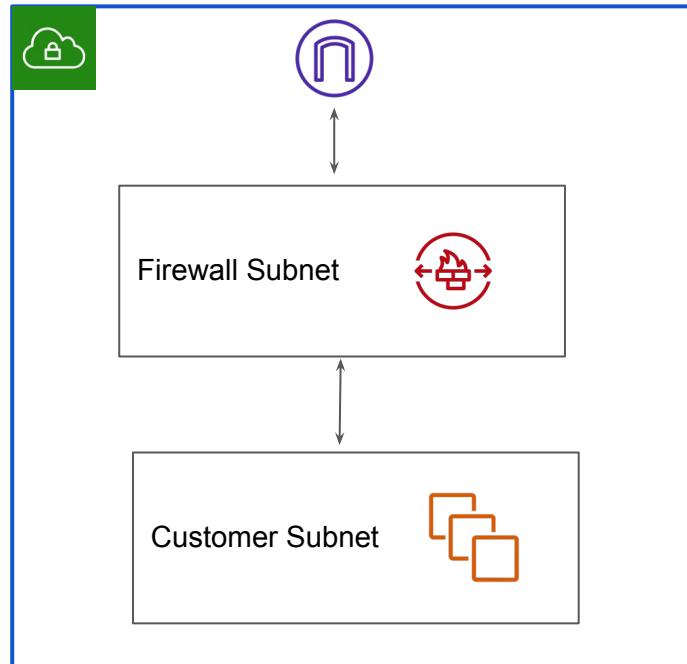
1. Pass traffic through only from known AWS service domains or IP address endpoints, such as Amazon S3.
2. Use custom lists of known bad domains to limit the types of domain names that your applications can access
3. Perform deep packet inspection on traffic entering or leaving your VPC

Deploying Network Firewall

Let's Deploy Network Firewall

Basic Deployment Architecture

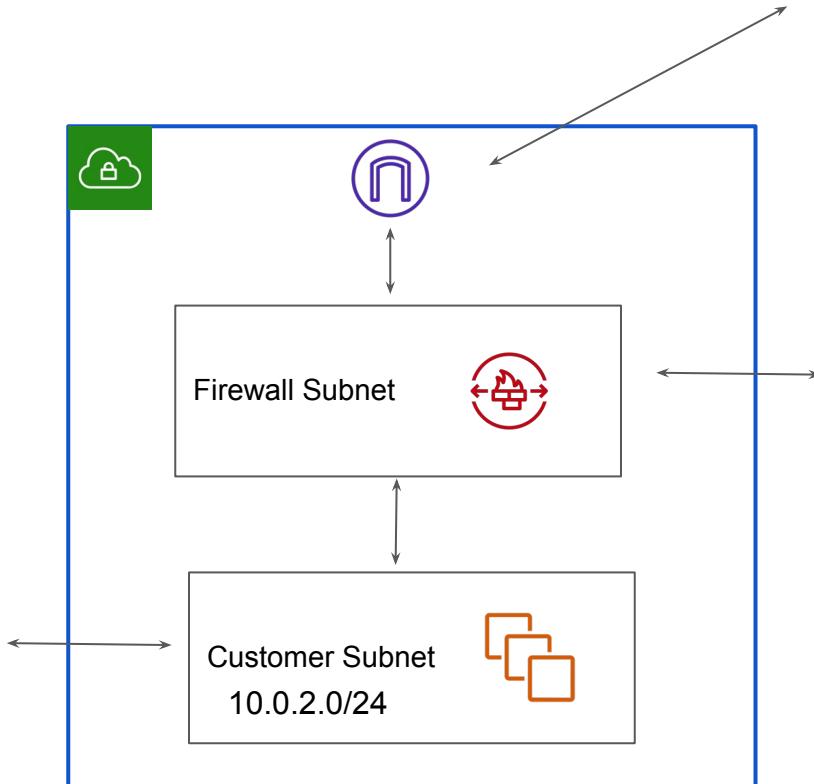
The Network firewall protects the subnets within your VPC by filtering traffic going between the subnets and locations outside of your VPC



Route Table Entries

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	vpce-1234

Customer Subnet



Destination	Target
10.0.2.0/24	vpce-1234

IGW

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-1234

Firewall Subnet

Configuration Steps

Following are the 3 resource types that Network Firewall Manages.

Resource Type	Description
RuleGroup	Defines a set of rules to match against VPC traffic, and the actions to take when Network Firewall finds a match.
FirewallPolicy	Allows adding multiple rule groups and configure other settings.
Firewall	Provides traffic filtering logic for the subnets in a VPC.

Mitigating DDOS

The stronghold for Fort

Mitigating DDOS

- Be ready to scale as traffic surges.
- Minimize the attack surface area.
- Know what is normal and abnormal.
- Create a Plan for Attacks.



Be Ready to Scale

1. Be Ready to Scale

- Your infrastructure should be designed to scale when the traffic increases.
- It not only helps in Business but also during DDOS Attacks.

Example :

Whenever CPU load is more than 70% in Application servers, automatically add one more Application server to meet the needs.

AWS Services : ELB, Auto Scaling

Let's Minimizing is the Key

2. Minimize the attack surface area.

Decouple your infrastructure.

Example :

Application and Database should not be on the same server.

AWS Services : SQS, Elastic BeanStalk

Normal and Abnormal

3. Know what is normal and abnormal

- Key metrics need to be defined to understand the behavior.

Example :

Website getting a huge surge in traffic in the middle of the night at 3 AM

AWS Services :- CloudWatch, SNS.

Create a Plan

4. Create a Plan for Attacks.

For example :

- Check whether the Source IP Address is the same.
- Check from which country the increased traffic is coming from.
- Nature of the attack (SYN Flood, Application Level)
- Can it be blocked with NACL or Security Group level.



It is recommended to have AWS Support. At-least Business Support.

AWS Services for DDoS Attack Mitigation

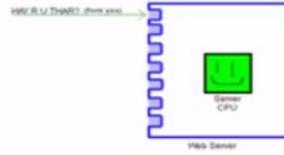
Following are some of the key AWS services involved in DDoS attack mitigation

- **AWS Shield**
- **Amazon CloudFront**
- **Amazon Route53**
- AWS WAF
- Elastic Load Balancing
- VPC & Security Groups

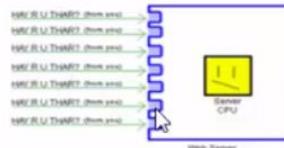
Denial of Service

Attack difficult to mitigate

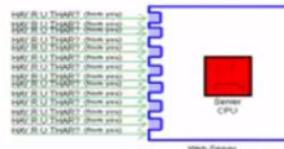
normal service →



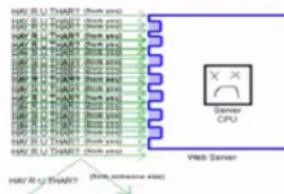
high traffic →



single DOS →



LOL DDOS'D →



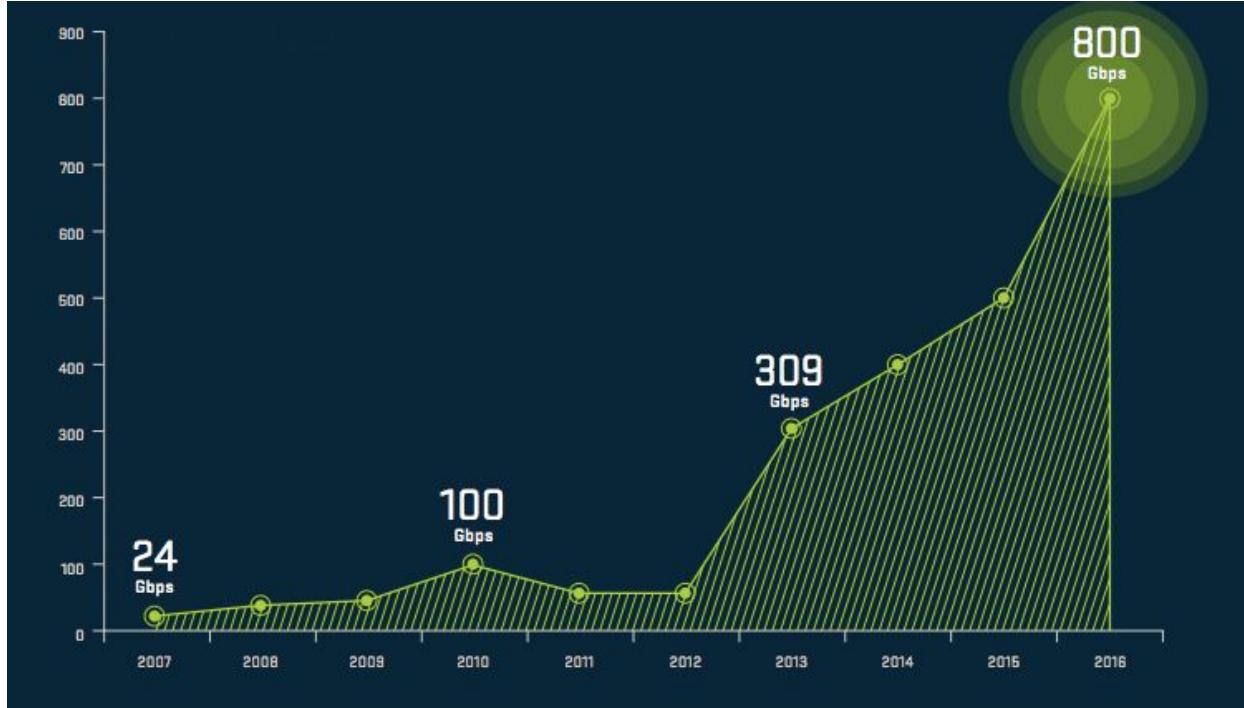
DOS and DDoS are part and parcel of servers life

DOS and DDoS attacks are very common attack vectors used nowadays to bring down the servers or flood the network.

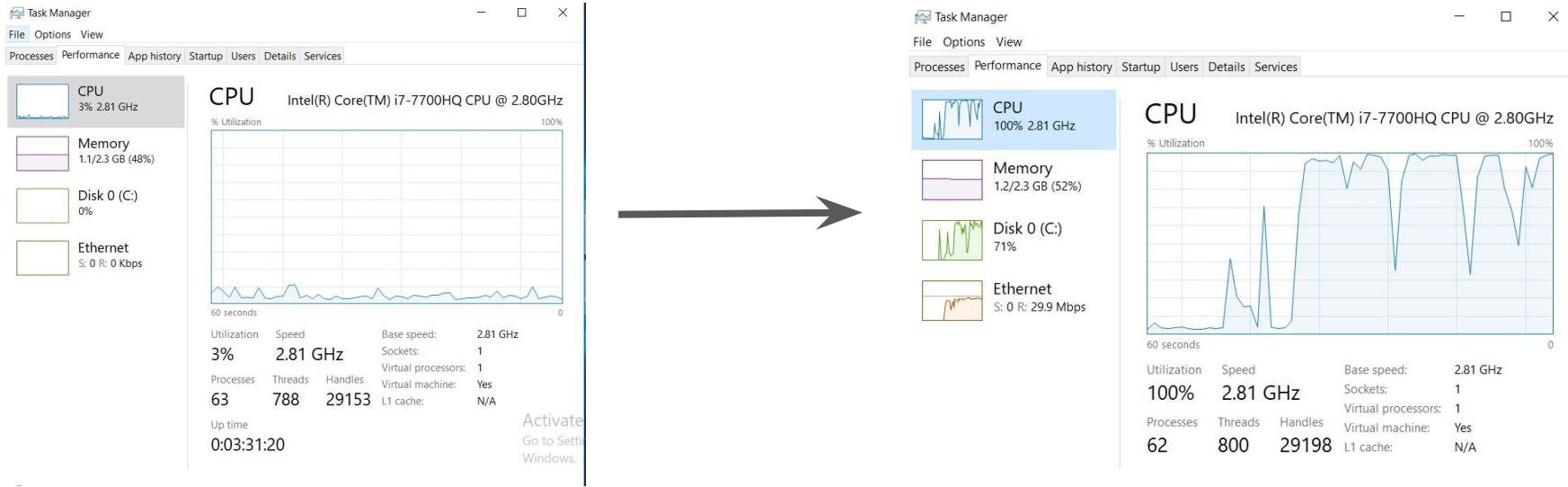
The reason why they are so successful is because of ease of ability to launch the attack and most of the protection mechanisms are based on expensive hardware.



DDOS attacks are going really big!



Before vs After (DOS Attack)



DDOS Attacks Crush Twitter, Hobble Facebook

Posted Aug 6, 2009 by Michael Arrington (@arrington)



The image shows a screenshot of the Twitter homepage. At the top, there's a navigation bar with links for Home, Profile, Find People, Settings, Help, and Sign out. Below this, a large white box contains a message: "We had network issues today related to a denial-of-service attack. Service now is restored for most people and we're investigating further." This message was posted 8 minutes ago from the web. At the bottom of the page, there's a Facebook logo with the word "Facebook" next to it. The footer contains links for © 2009 Twitter, About Us, Contact, Blog, Status, Goodies, API, Business, Help, Jobs, Terms, and Privacy.

Crunchbase

Facebook	
FOUNDED	2004
OVERVIEW	
LOCATION	Menlo Park, California
CATEGORIES	
WEBSITE	http://www.facebook.com

AWS Shield

DDoS Protection

Understanding AWS Shield

AWS Shield is a managed Distributed Denial of Service (DDoS) service that safeguards the workloads running on AWS against DDoS attacks.

There are two tiers of AWS Shield:

- Shield Standard
- Shield Advanced

Understanding AWS Shield

AWS Shield standard provides basic level protection against most common network and transport layer DDoS attacks.

For a higher level of protection, we can subscribe to the Shield Advanced. Shield Advanced protects against large and sophisticated DDoS attacks with near-real-time visibility into the attacks that might be occurring.

AWS Shield Advanced also gives customers 24x7 access to the AWS DDoS Response Team (DRT) during ongoing attacks.

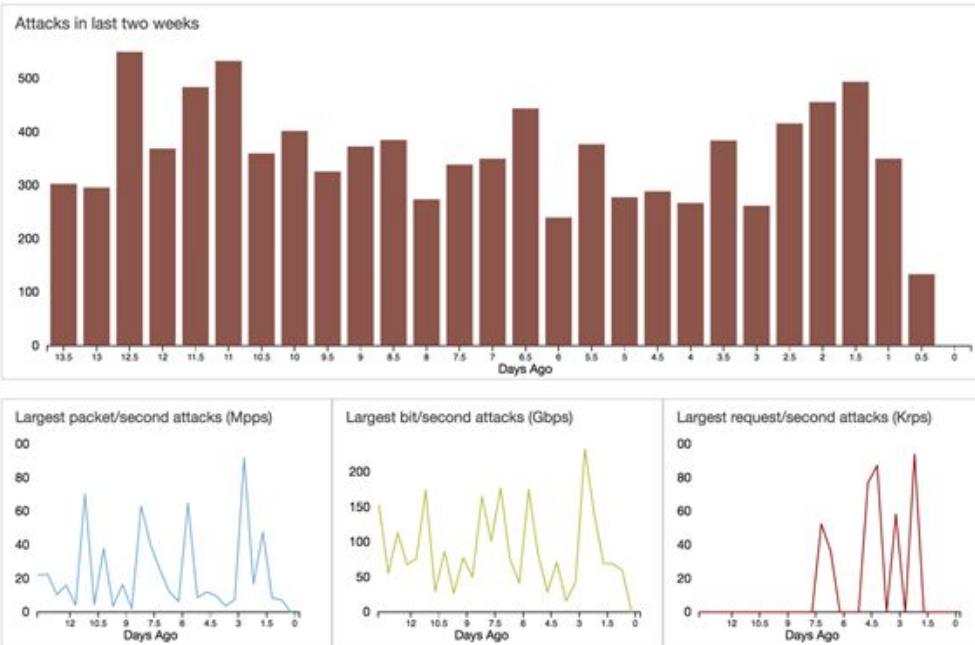
AWS Shield Costs and Credits

AWS Shield Advanced costs 3000\$ per organization and requires Business or Enterprise Support.

One interesting part about AWS Shield Advanced is that during the attack, if your infrastructure has scaled, AWS will return you the amount occurred during scaling in the form of credits. This is also referred to as Cost protection.



AWS Shield Dashboard



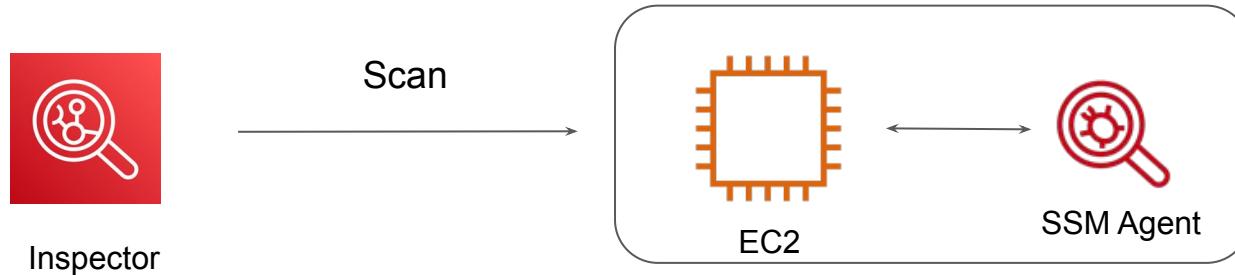
AWS Inspector

Vulnerability Scanner

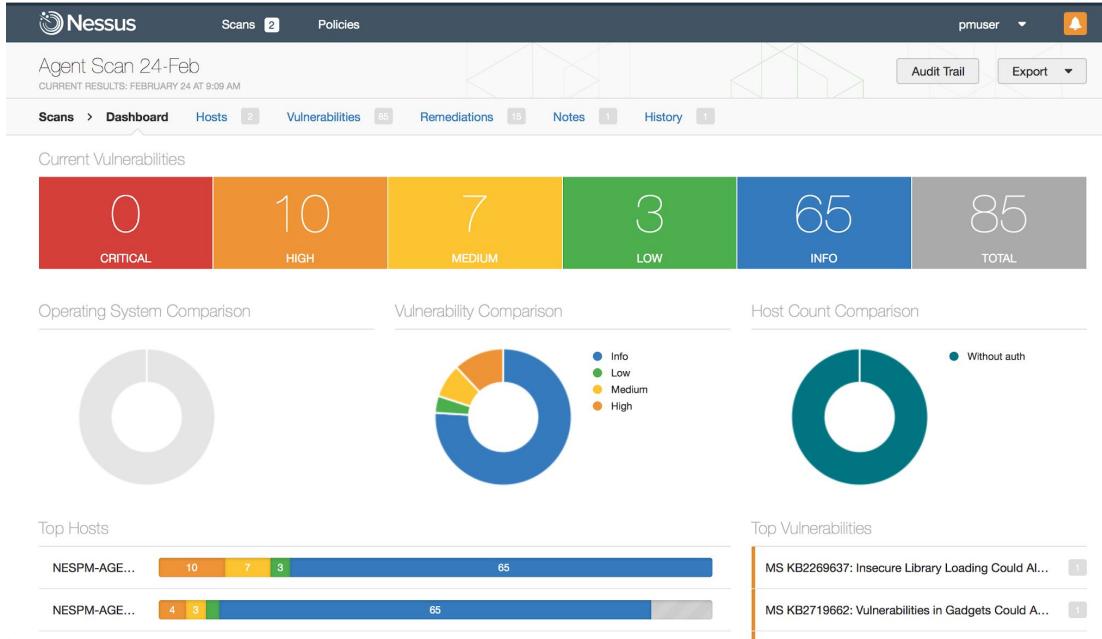
Basics of AWS Inspector

AWS Inspector is similar to a vulnerability scanner which will scan the system for specific vulnerabilities and provide the results.

It relies on the agent installed on the server to scan the server.

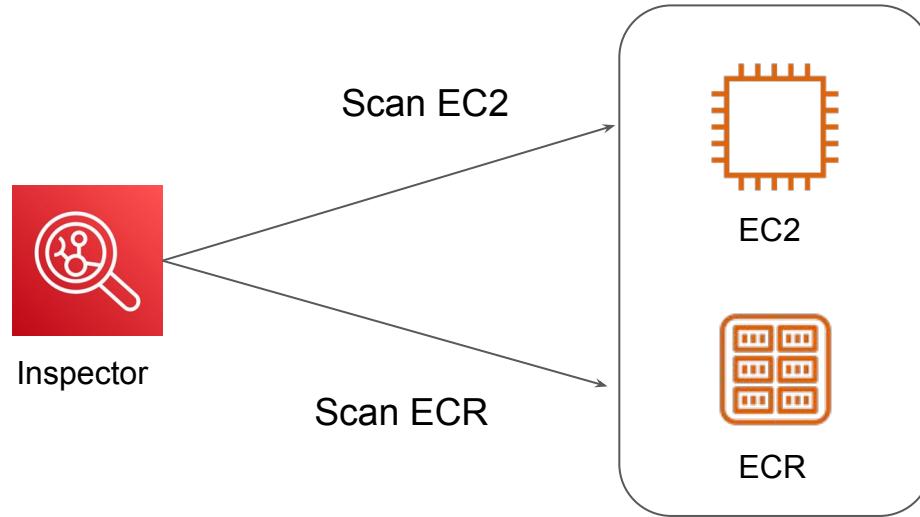


Similar to Nessus



Supported Scans

Amazon Inspector gives you the flexibility to enable either EC2 scanning or ECR container image scanning, or both.

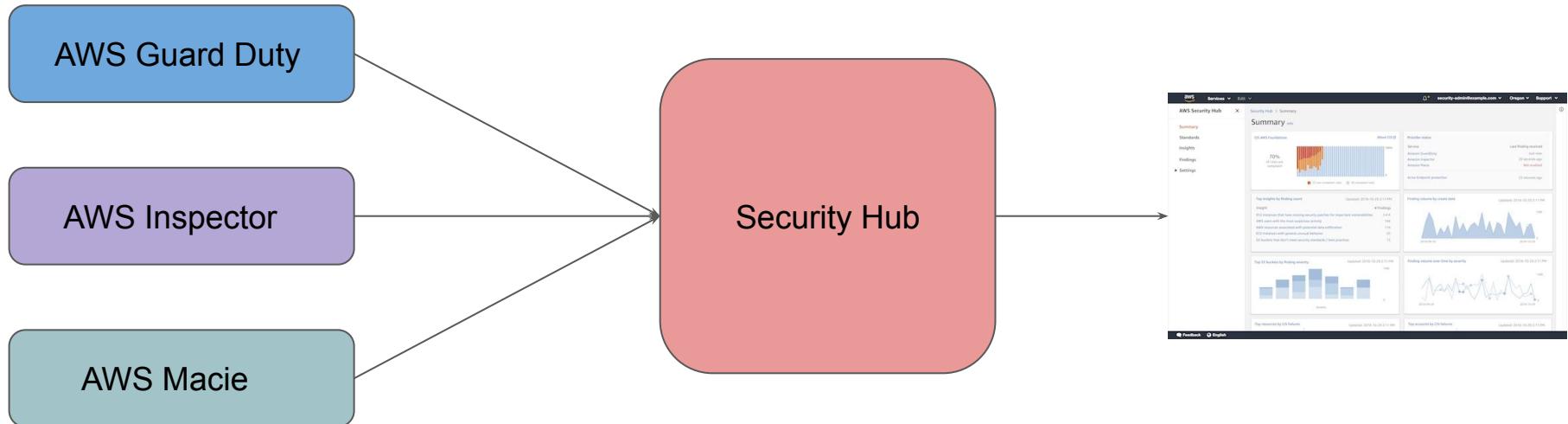


Security Hub

Centralized Security Hub

Overview of Security Hub

AWS Security Hub gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.



Supported Compliance Standard

AWS Security Hub also has ability to generate its own findings by running automated and continuous checks against the rules in a set of supported security standards.

Following Standards are supported:

- CIS AWS Foundation
- PCI DSS



Standard	Passed	Failed	Score ▲
CIS AWS Foundations Benchmark v1.2.0	12	30	29%
PCI DSS v3.2.1	22	9	69%

Web Application Firewall

Next generation firewalls

Getting started

We all know about Firewalls and in some way might have worked as well.

Firewall works on the Layer 3 and Layer 4 of the OSI model.

Main aim of firewall: Block malicious and unauthorized traffic.

However what about malicious traffic like SQL Injection attacks, XSS and many more ?

Introducing WAF

A Web Application Firewall is an application level firewall for HTTP applications.

It applies set of rules for the HTTP based conversations.

WAF generally are deployed to protect against attacks targeted towards application, specifically the ones defined in the OWASP Top 10 metrics.



WAF Vendors

There are lot of ways in which you can implement WAF and various vendors as well.

Naxsi and Modsecurity are some of the famous open sources ones.

Signal Sciences, Akamai, AWS WAF are some of the commercial vendors that offer WAF related functionalities.



AWS WAF

Protection against Layer 7 Attacks

Understanding AWS WAF Concepts

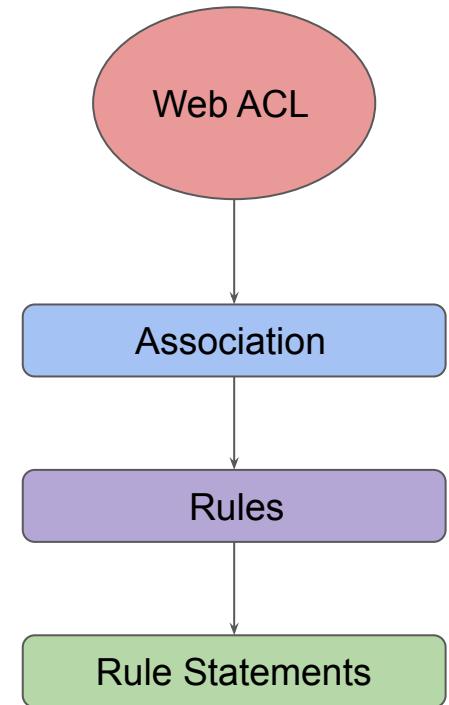
I live in a place A in Bangalore and want to meet my friend living in place B in Bangalore.

Rule Statement: If traffic is less on the roads?
Are there any Uber / OLA available?

Rules: If traffic is less AND uber ola available then yes or no

WebACL: Container for all the things + default action.

Association: Zeal



Rule Statements

Rule Statements define basic characteristics that would be analyzed within a web request.

These can be custom-defined or you can use ready-made ones from AWS and marketplace.

1. Block all the requests which are coming from out of India.
2. Block request which has a URI Path of /admin

You can even build custom condition based on:

Headers, HTTP Method, Query Strings, URI Path, Geo-Location, Body.

Rules in WAF

We can combine multiple statements into rules to precisely target requests.

WAF provides two primary rule types: **Regular Rule & Rate-Based rule**

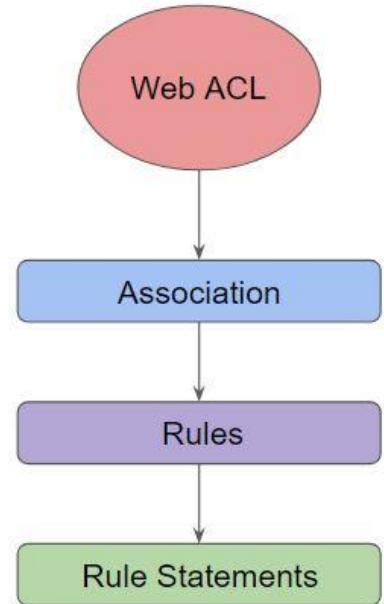
Let's look into sample regular rule:

1. If a request comes from 172.30.0.50
2. Request has SQL-like code

Rules with multiple statements can be AND, OR, NOT

Rate-Based rule = Regular Rule + Rate limiting feature

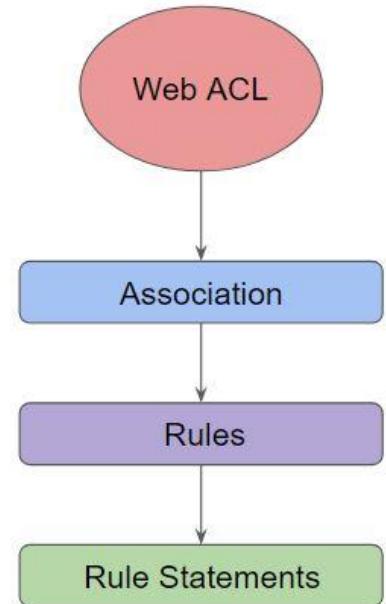
1. If a request comes from 172.30.0.50
2. If requests exceeds 1000 request in 10 minutes



Web ACL in WAF

Web ACL is a centralized place that contains the rules, rule statements and associated configuration.

It is used to define the protection strategy.

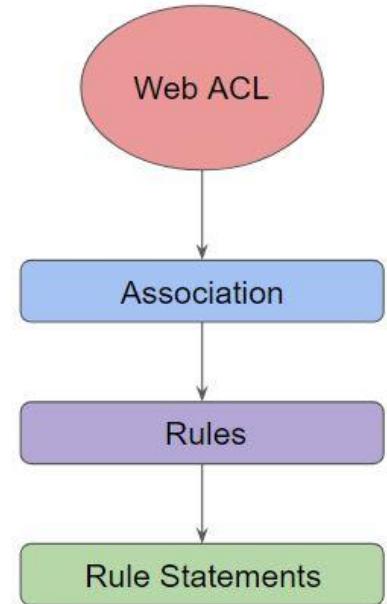


Association in WAF

Association defines to which entity WAF is associated to.

WAF cannot be associated with EC2 instances directly.

Support Association: ALB and CloudFront, API Gateway



Important Pointers

Rule Groups can be configured which has multiple rules that can be used across multiple Web ACLs.

Customers can decide to use ready-made AWS-Managed rules or even rules from AWS Marketplace.

Every Rule has a priority. If a request matches Priority 0 rule, none of the other rules will inspect the request

Pricing Aspect:

Web-ACL (\$5 per month), Rule (\$1 per month), Requests (\$0.60 / 1 million)

Relax and Have a Meme Before Proceeding

When someone says
you look nice and it
makes you feel nice.



Identity Account Architecture

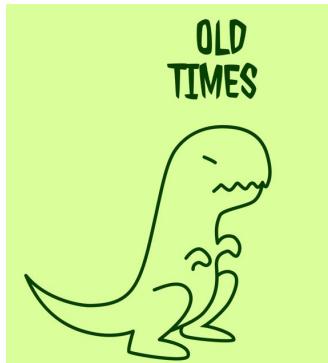
Multiple Accounts are Good

The Initial Start

During the earlier days of AWS, most of the organizations had a single AWS account.

Management was simple.

User would have had a single set of username/password AND access/secret keys.

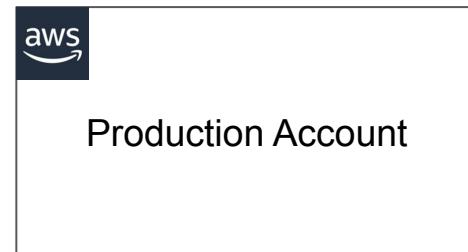
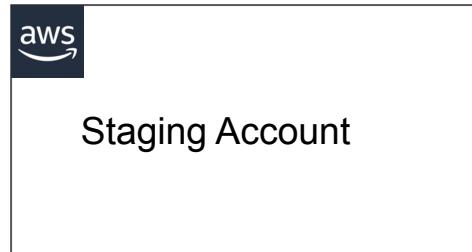
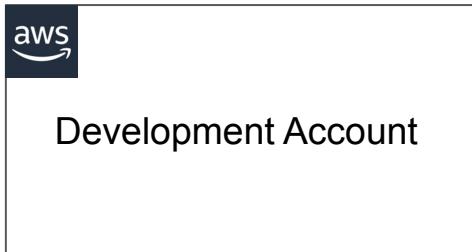


Organizations Became Big

A better architecture with multiple AWS account per function was adopted.

Each user had different username/password AND access/secret keys for each account.

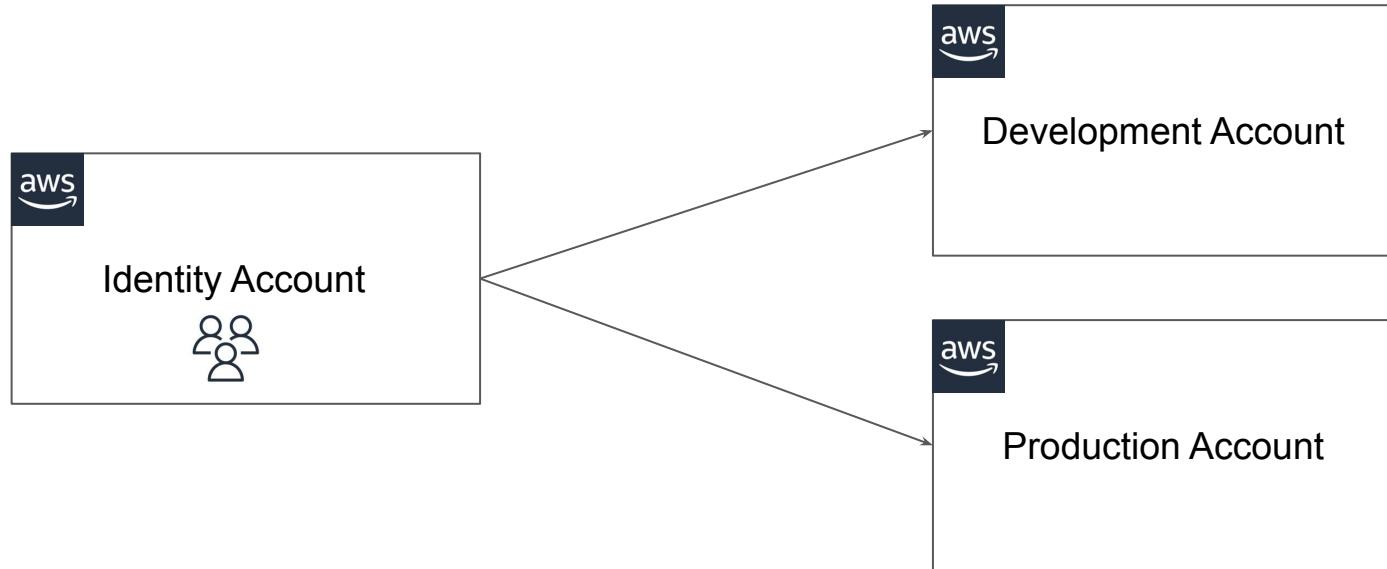
Difficult to Work with. Required a lot of Bookmarks



Rise of Identity Account

In Identity Account architecture, all the IAM Users are stored in central AWS Account.

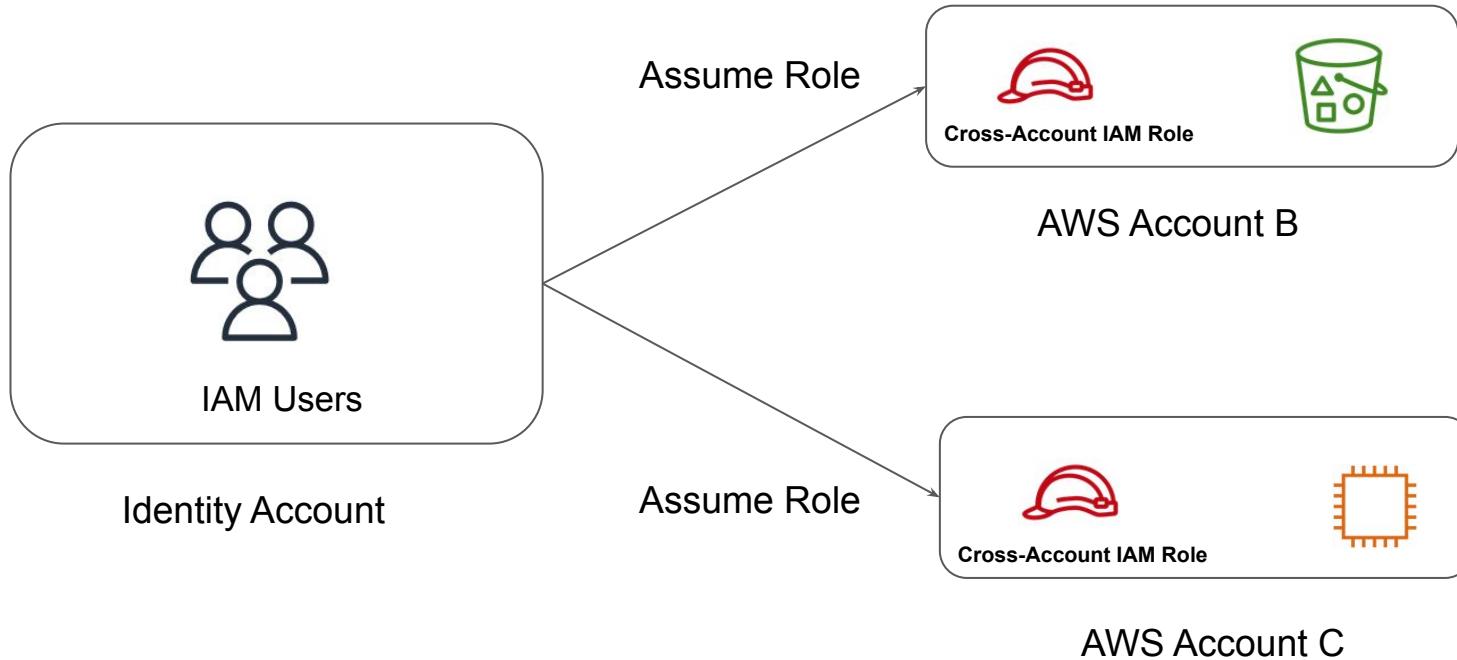
They could easily connect to Dev/Prod accounts without separate credentials.



The Architecture

- i) Create a user in Account A.
- ii) Create a Cross-Account role in Account B.
- iii) Allow User to switch to Account-B Role.

The Practical Architecture

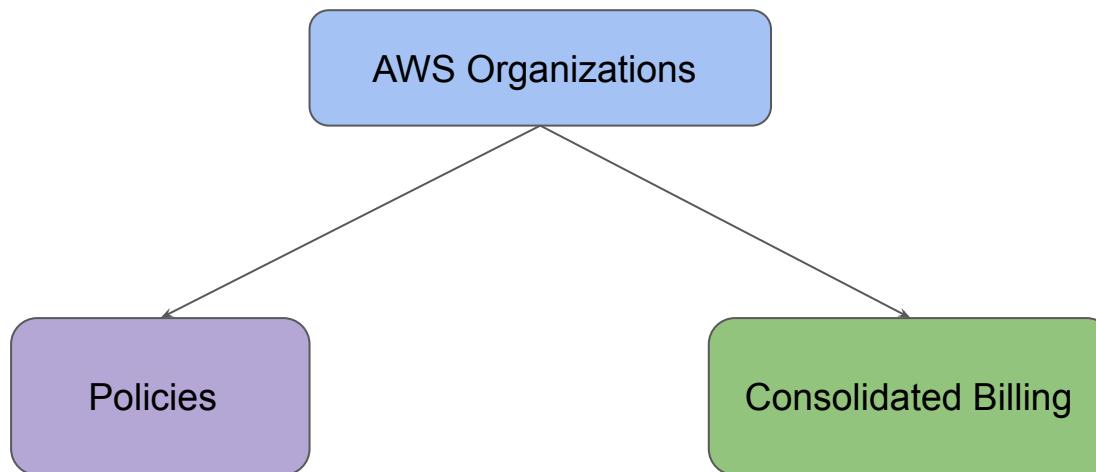


AWS Organizations

Centralized Control

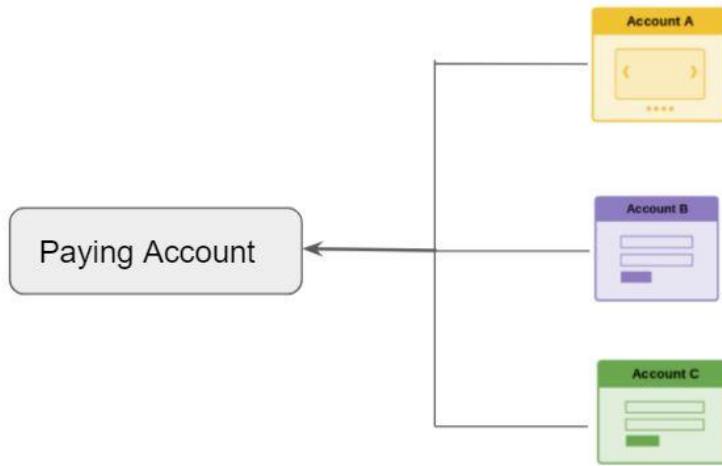
Getting the basics right

AWS offers centralized policy-based management as well as the feature of consolidated billing for multiple AWS accounts through the feature of AWS Organizations.



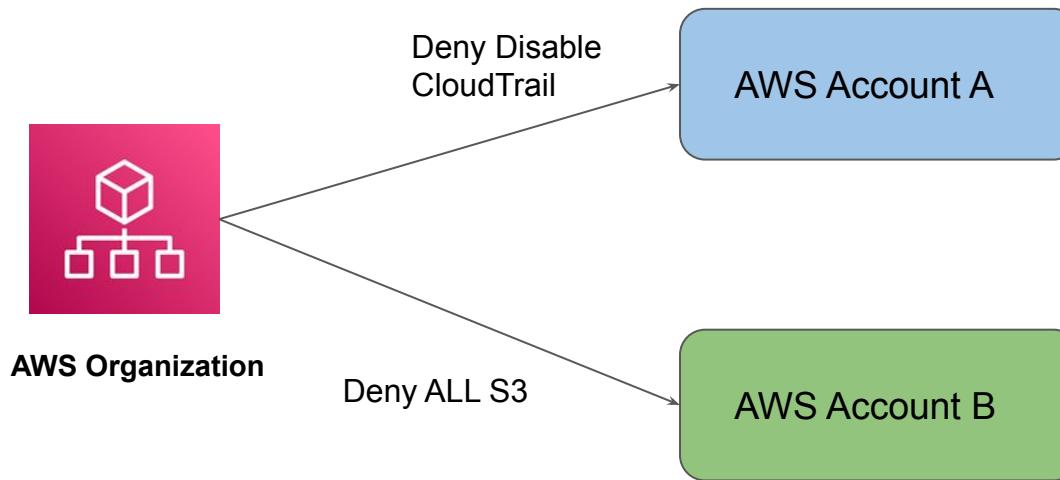
Part 1 - Consolidated Billing

In consolidated billing, management account to access the billing information and pay for all member accounts.



Part 2 - Policies

Policies in AWS Organizations enable you to apply additional types of management to the AWS accounts in your organizations.



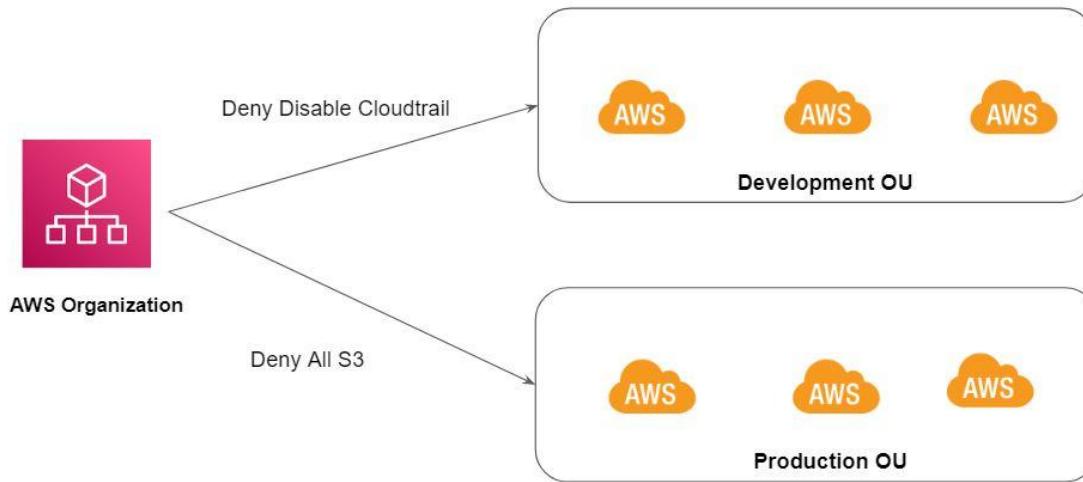
OU in AWS Organization

Were Complex becomes Easy

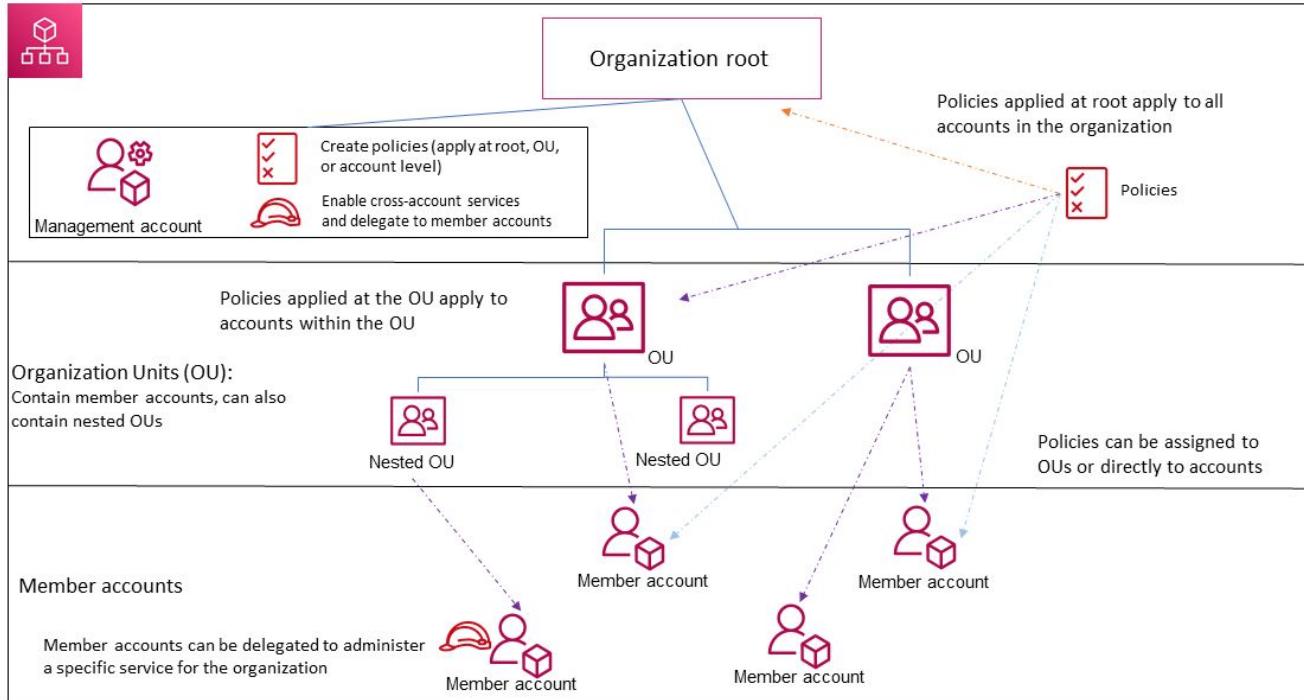
Getting the basics right

Organizational units (OUs) to group accounts together to administer as a single unit

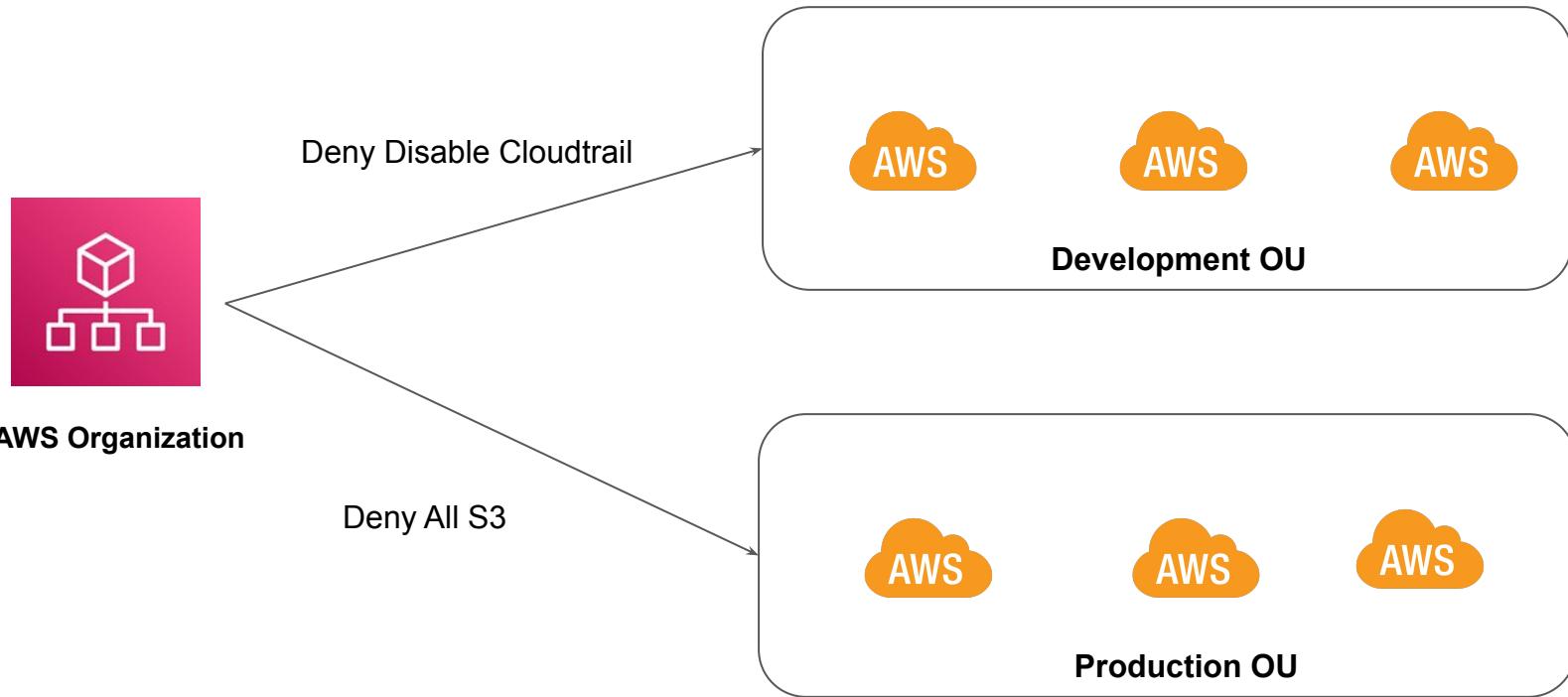
This greatly simplifies the management of your accounts. For example, you can attach a policy-based control to an OU, and all accounts within the OU automatically inherit the policy.



Important Concepts



Grouping AWS Accounts



Important Pointers

SCPs don't affect users or roles in the management account. They affect only the member accounts in your organization.

By default, AWS Organizations attaches an AWS managed policy called FullAWSAccess to all roots, OUs, and accounts. This helps ensure that, as you build your organization, nothing is blocked until you want it to be.

IAM Permissions Boundaries

Advanced IAM

Getting Started

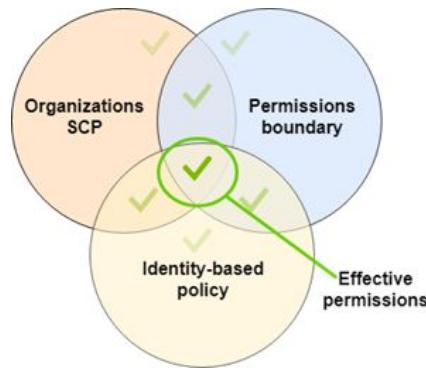
A permissions boundary is an advanced feature in which you use a managed policy to set the maximum permissions that an identity-based policy can grant to an IAM entity.

When you set a permissions boundary for an entity, the entity can perform only the actions that are allowed by both its identity-based policies and its permissions boundaries.

Evaluating Effective Permission with Boundaries

The **effective permissions** for an entity are the permissions that are granted by all the policies associated with the user/role/account.

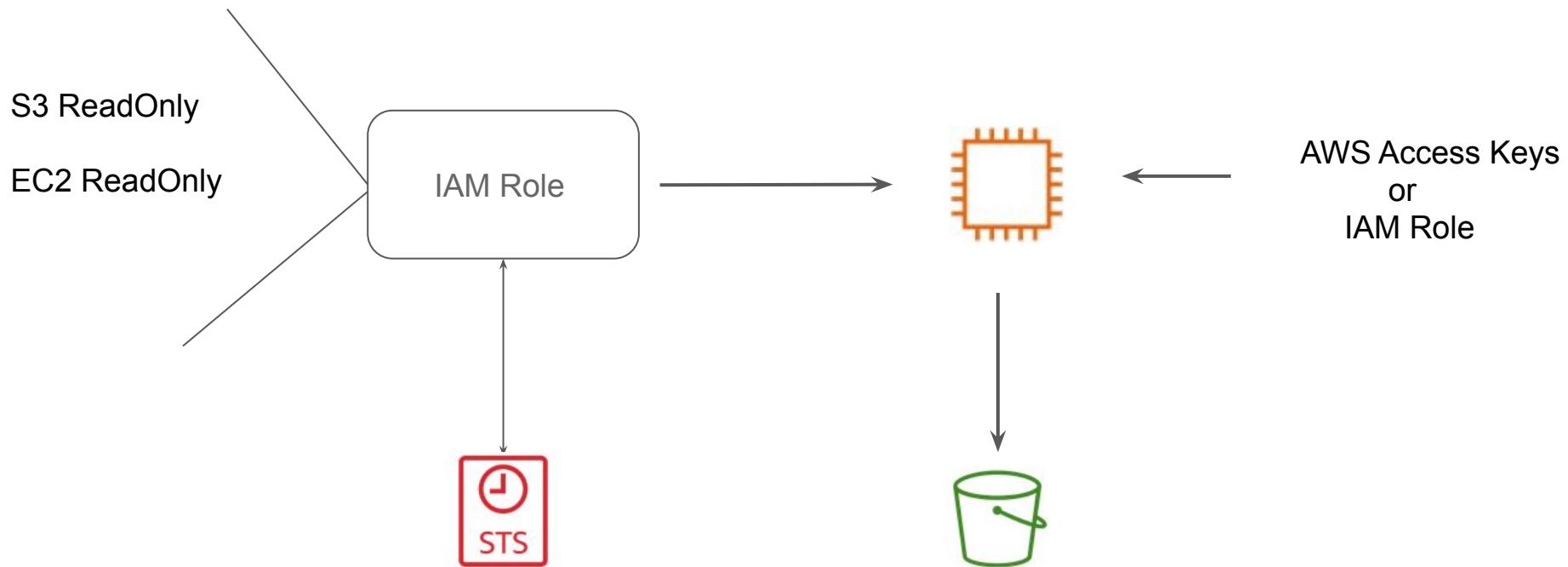
Within an AWS account, the permissions for an entity **can be affected** by identity-based policies, resource-based policies, permissions boundaries, Organizations SCPs, or session policies.



AWS Secure Token Service (STS)

Credentials Management

How IAM Role

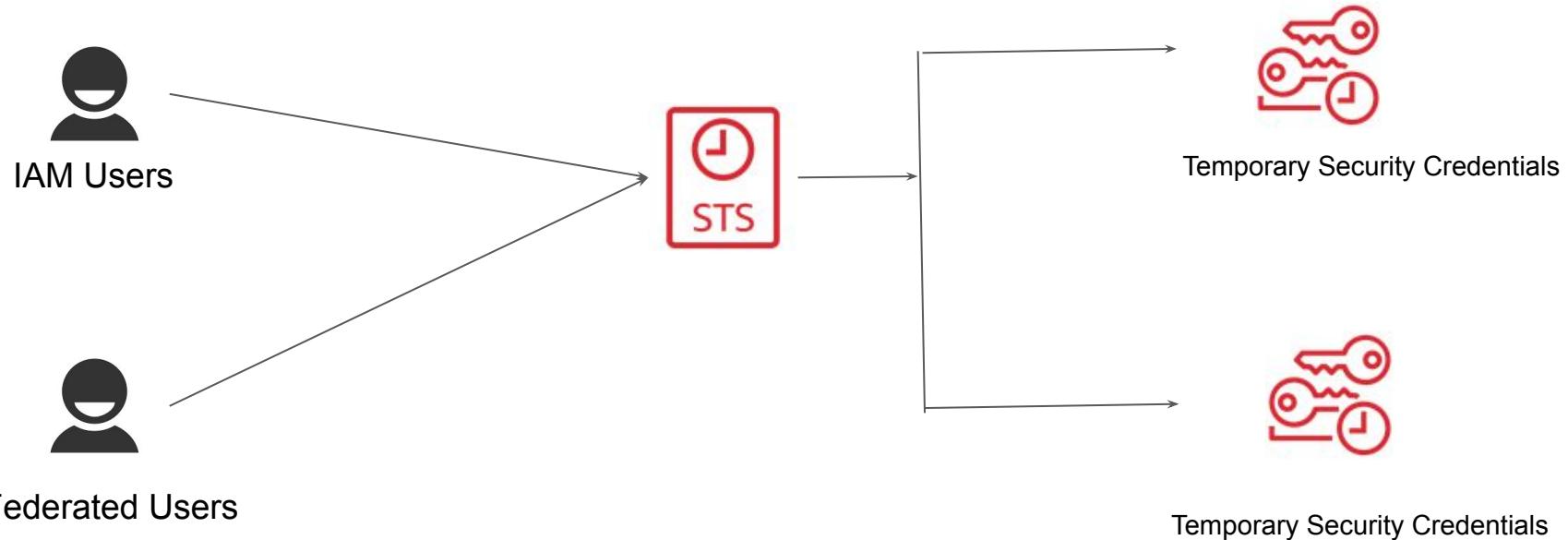


Overview of STS

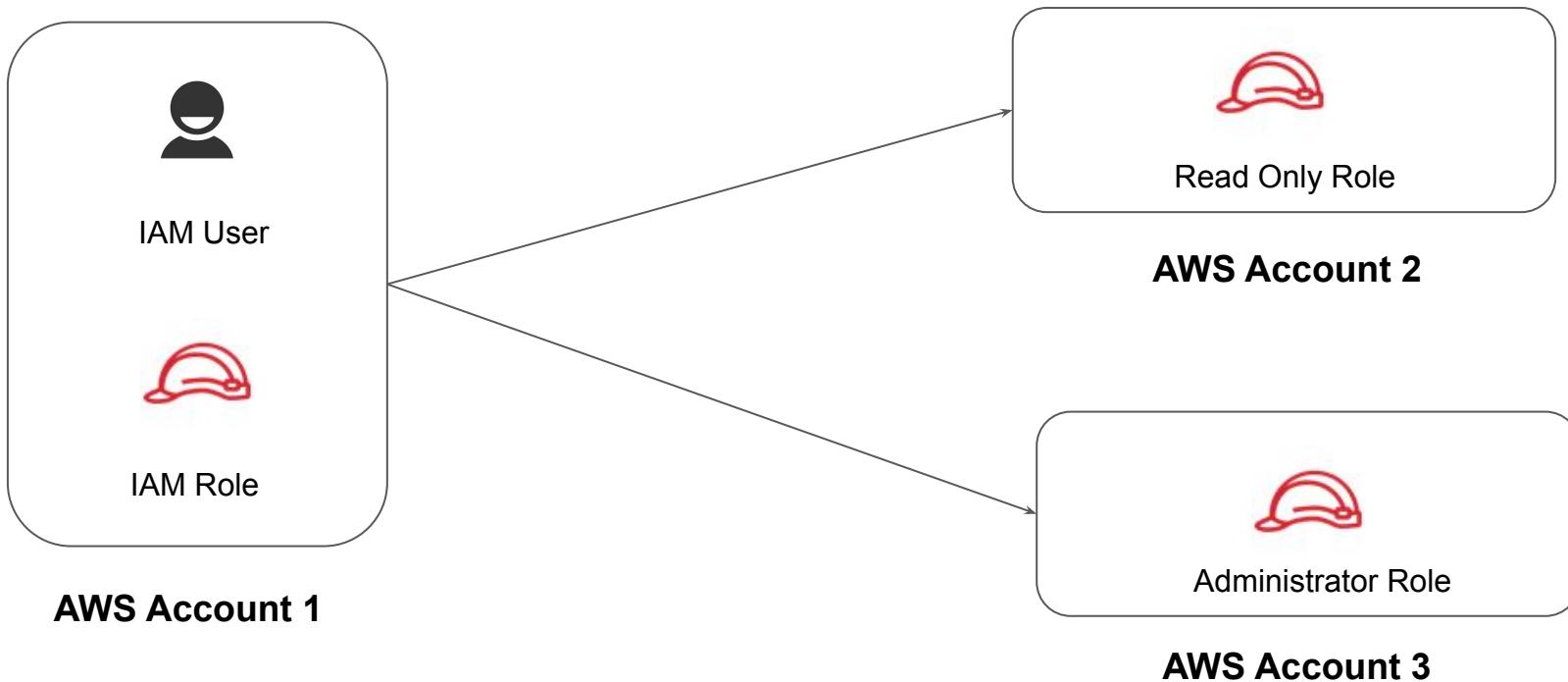
- The AWS STS is a web service that enables you to request temporary, limited-privilege credentials for AWS Identity and Access Management (IAM) users or for users that you authenticate (federated users).
- Temporary security credentials are short term and expire after a certain duration.
- Since they have a limited lifetime, the key rotation is no longer explicitly needed.

```
{  
    "AssumedRoleUser": {  
        "AssumedRoleId": "AROAJOTOADWSDZD53Z7VS:temp",  
        "Arn": "arn:aws:sts::836802967410:assumed-role/CA-EC2RO/temp"  
    },  
    "Credentials": {  
        "SecretAccessKey": "LKtyaWrhxGnBNP3tx7dMK2nv0H1VdwMP1RVP5Sob",  
        "SessionToken": "FQoDYXdzEMj//////////wEaDHwScBw1Hmr5eGqKXyLHAdeXEJZ0oSuJxFd/PGtU  
Z5F3XhjgIawg7ytJXXWRgpyvaq9eMKNfUqmiDca/NM+FLwqy5iek5VKPGkPut+/pAz0WH3ddVmcuhsJowHxaDGHa  
d6S21yhyMFAF9bk9FQjMFHNt1/oD174KvkAV6xAE4Q0cPZ4sDGes130Im4r5Tu1KT/I2qvg0w/LVRjraJ8UBnopMu  
gU=",  
        "Expiration": "2017-09-20T23:36:41Z",  
        "AccessKeyId": "ASIAJWPD367QI4GHCNAQ"  
    }  
}
```

Overview Architecture of STS



STS Architecture - Cross-AWS Account Access



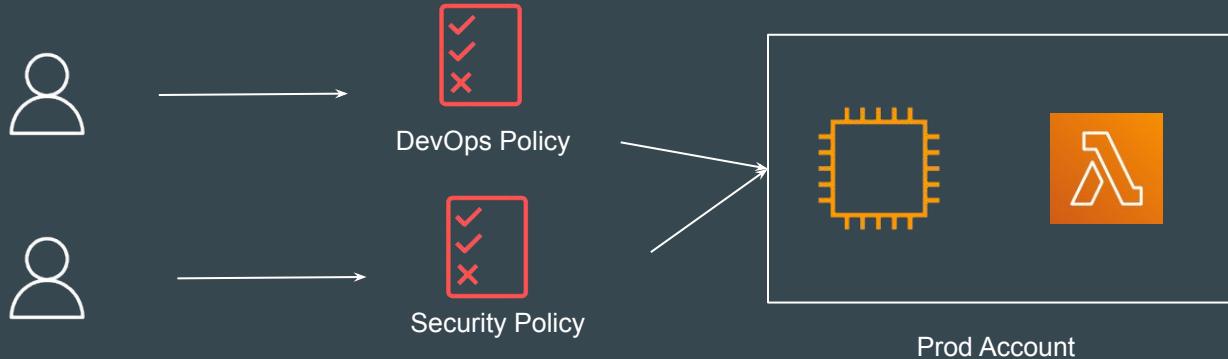
Attribute-based access control



Basics of RBAC

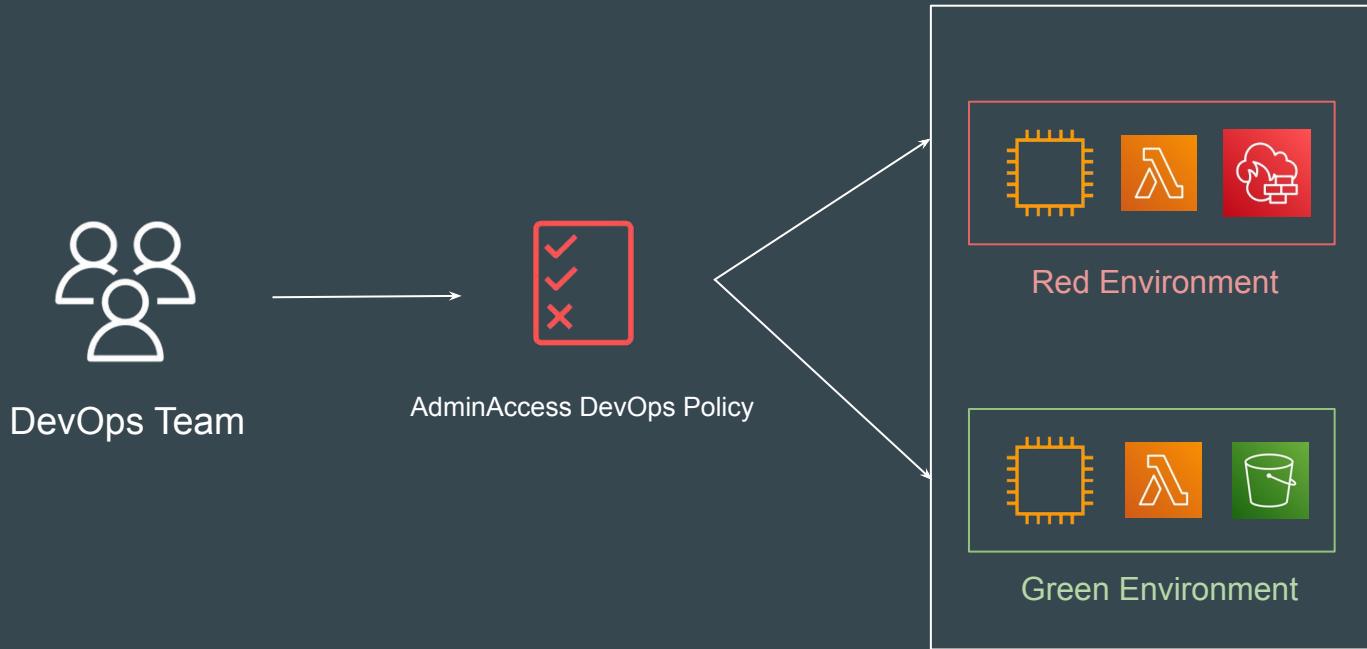
Role-based access control (RBAC) restricts access based on a person's role within an organization

In IAM, you implement RBAC by creating different policies for different job functions



Understanding the Challenge

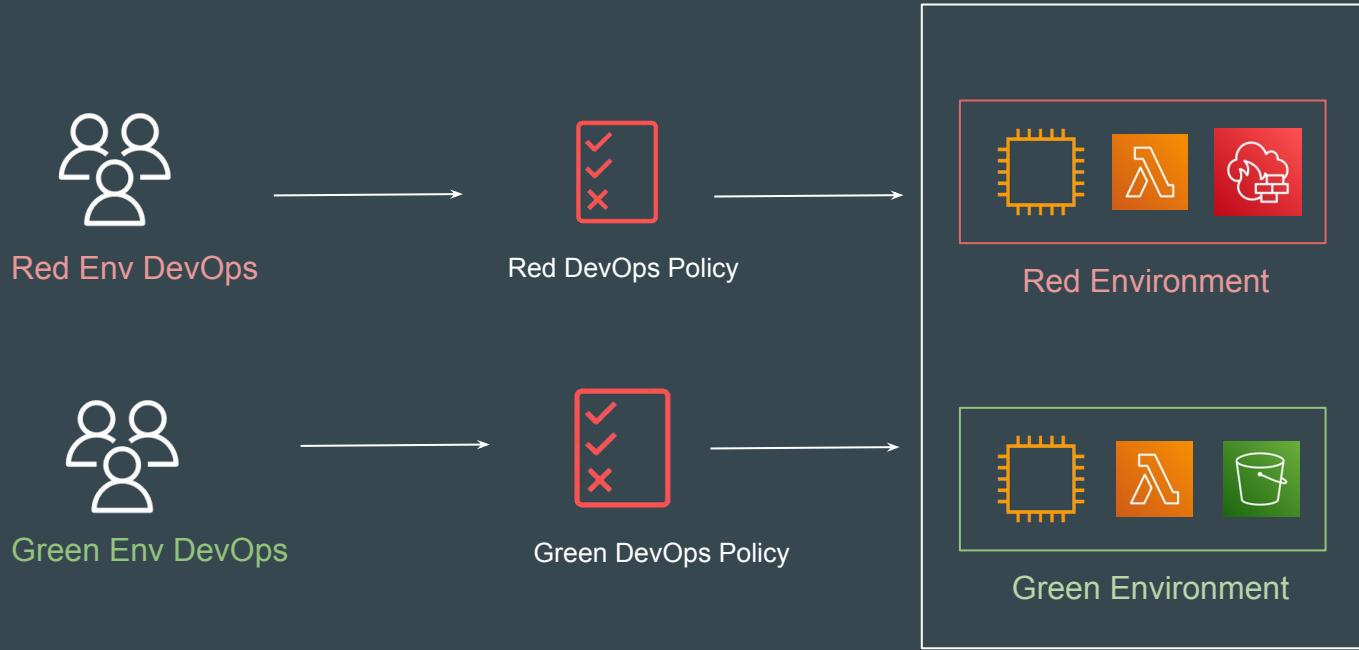
DevOps Team has access to Red and Green Environment



Possible Approach of Separation

Red Env DevOps only has access to Red Environment

Green Env DevOps only has access to Green Environment



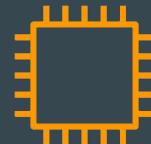
Basics of Attributes

Attributes are key-value pairs.

In AWS, these attributes are called tags.

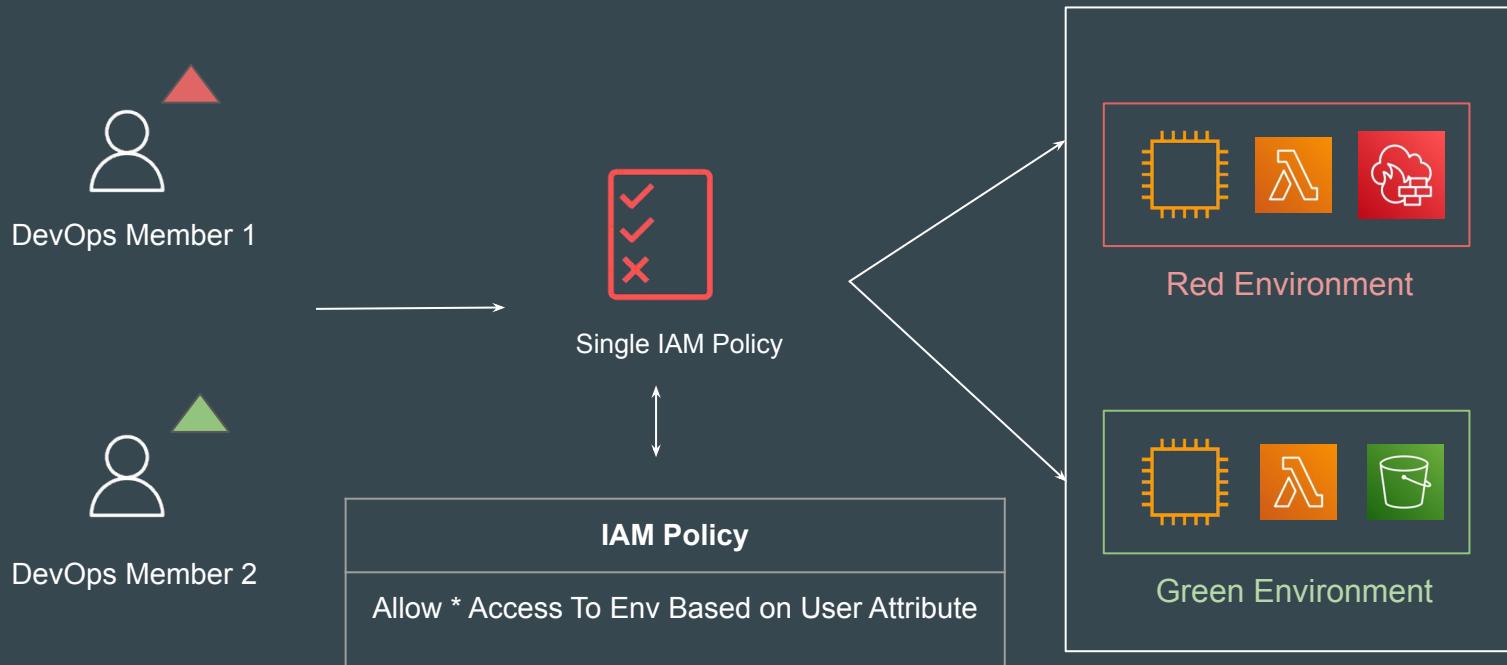


Team: DevOps
Location: India



Env: Prod
Project: Green

Scalable Permission Model based on Attributes



Attributes for IAM User

You can use IAM tag key-value pairs to add **custom attributes** to an IAM user.

The screenshot shows the AWS IAM User Details page for a user named "user01-green-team". The user was created on January 28, 2023, at 19:35 UTC+05:30. It has an ARN of arn:aws:iam::042025557788:user/user01-green-team. Console access is enabled without MFA. There are two Access keys: one is not enabled, and the other is also not enabled. The "Tags (1)" section shows a single tag named "Team" with the value "Green". Other tabs visible include "Permissions", "Groups (1)", "Tags (1)", "Security credentials", and "Access Advisor".

IAM > Users > user01-green-team

user01-green-team

Summary

ARN	Console access	Access key 1
arn:aws:iam::042025557788:user/user01-green-team	⚠ Enabled without MFA	Not enabled
Created	Last console sign-in	Access key 2
January 28, 2023, 19:35 (UTC+05:30)	⌚ Today	Not enabled

Permissions | Groups (1) | **Tags (1)** | Security credentials | Access Advisor

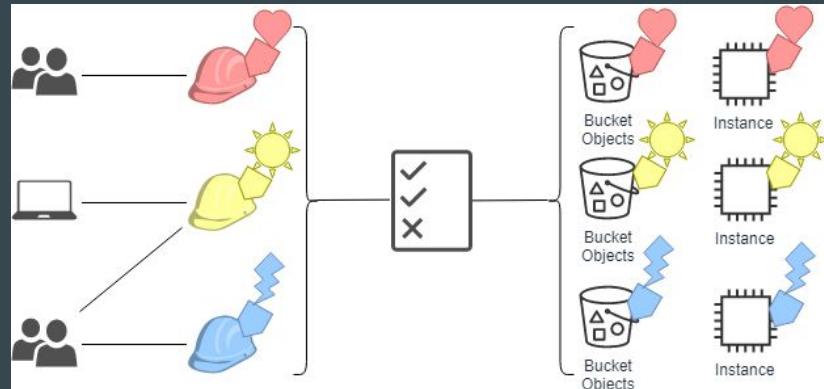
Tags (1)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

Key	Value
Team	Green

Attribute-Based Access Control

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes.



Permissions Based on ABAC

This example shows an IAM policy that allows a principal to start or stop an Amazon EC2 instance when the instance's resource tag and the principal's tag have the same value for the tag key Team



Key	Value
Team	Green

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeInstances"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:StartInstances",  
                "ec2:StopInstances"  
            ],  
            "Resource": "*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/Team": "${aws:PrincipalTag/Team}"  
                }  
            }  
        }  
    ]  
}
```

Benefits of ABAC

ABAC requires fewer policies. Because you don't have to create different policies for different job functions, you create fewer policies. Those policies are easier to manage.

Permissions can easily be granted and revoked based on user's tags.

You can even use attributes of users from **corporate directory** to allow / deny permissions to AWS resources.

Federation

Connecting Identities

Understanding the Challenge

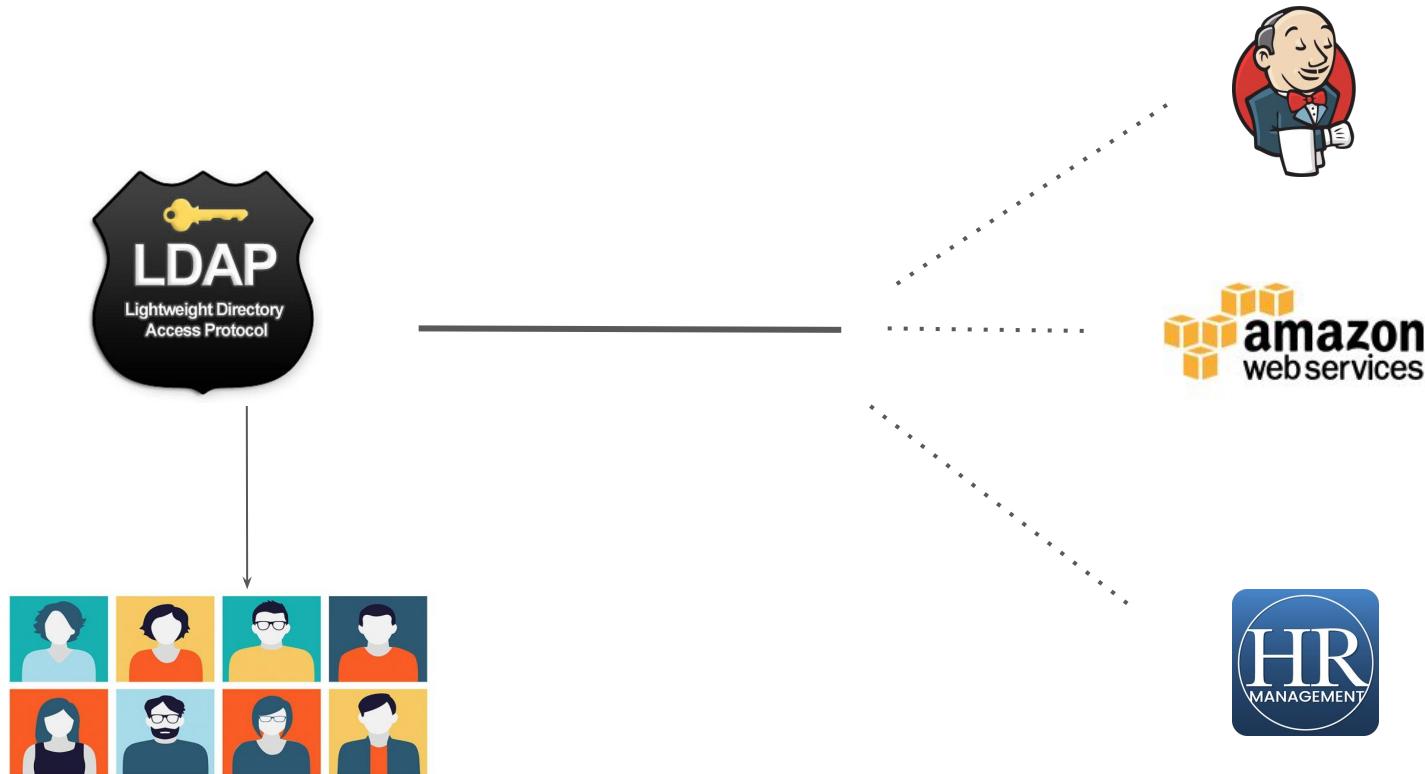
Let's assume there are 500 users within an organization. Your organization are using 3 services :-

- AWS (Infrastructure)
- Jenkins (CI / CD)
- HR Activator (Payroll)



You have been assigned role to give users access to all 3 services.

Storing Users Centrally



Active Directory Users and Computers

File Action View Help

Active Directory Users and Computers [pdc.e]

Saved Queries

enterprise.com

- Builtin
- CEO
- Computers
- Contractors
- Disabled Computers
- Disabled Users
- Districts
- Domain Controllers
- ForeignSecurityPrincipals
- Groups
- Inactive Users
- LostAndFound
- Managed Service Accounts
- Managers
- Microsoft Exchange Security Groups
- Production
- Program Data
- System
- TestOU
- Users
- Microsoft Exchange System Objects
- NTDS Quotas
- TPM Devices

Name Type Description

Ian Scur	User	
Cain Decker	User	
Elena Anderson	User	
Bill Jackson	User	Moved from: CN=Bill Jackson,OU=
Phill Jefferson	User	Moved from: CN=Phill Jefferson,OU=

Delegate Control...
Move...
Find...

New Computer
All Tasks Contact
Refresh Group
Export List... InetOrgPerson
View msExchDynamicDistributionList
Arrange Icons msImaging-PSPs
Line up Icons MSMQ Queue Alias
Properties Organizational Unit
Printer
Help User
Shared Folder

Create a new object...

The screenshot shows the Windows Active Directory Users and Computers (ADUC) management console. On the left is a navigation pane with a tree view of the directory structure under 'enterprise.com'. The main pane displays a list of users with columns for Name, Type, and Description. A context menu is open over the user list, with 'New' selected. A secondary dropdown menu shows various object types: Computer, Contact, Group, InetOrgPerson, msExchDynamicDistributionList, msImaging-PSPs, MSMQ Queue Alias, Organizational Unit, Printer, User, and Shared Folder. The 'User' option is also highlighted in this secondary menu.

Central Users

There are various solutions available which can store users centrally :-

- Microsoft Active Directory
- RedHat Identity Management / freeIPA



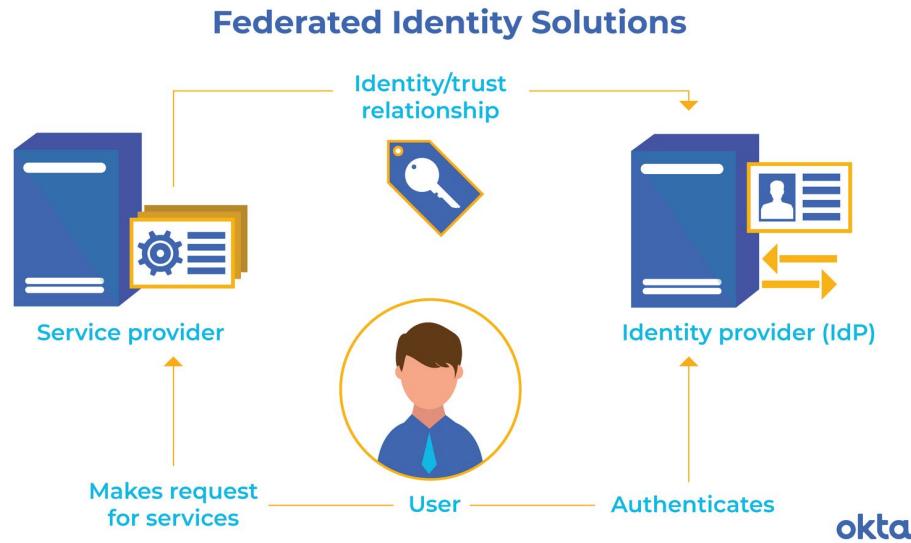
Basics of Federation - AWS Perspective

Federation allows external identities (Federated Users) to have secure access in your AWS account without having to create any IAM users.

These external identities can come from :-

- Corporate Identity Provider (AD, IPA)
- Web Identity Provider (Facebook, Google, Amazon, Cognito or OpenID)

Basic Workflow



Understanding Identity Broker

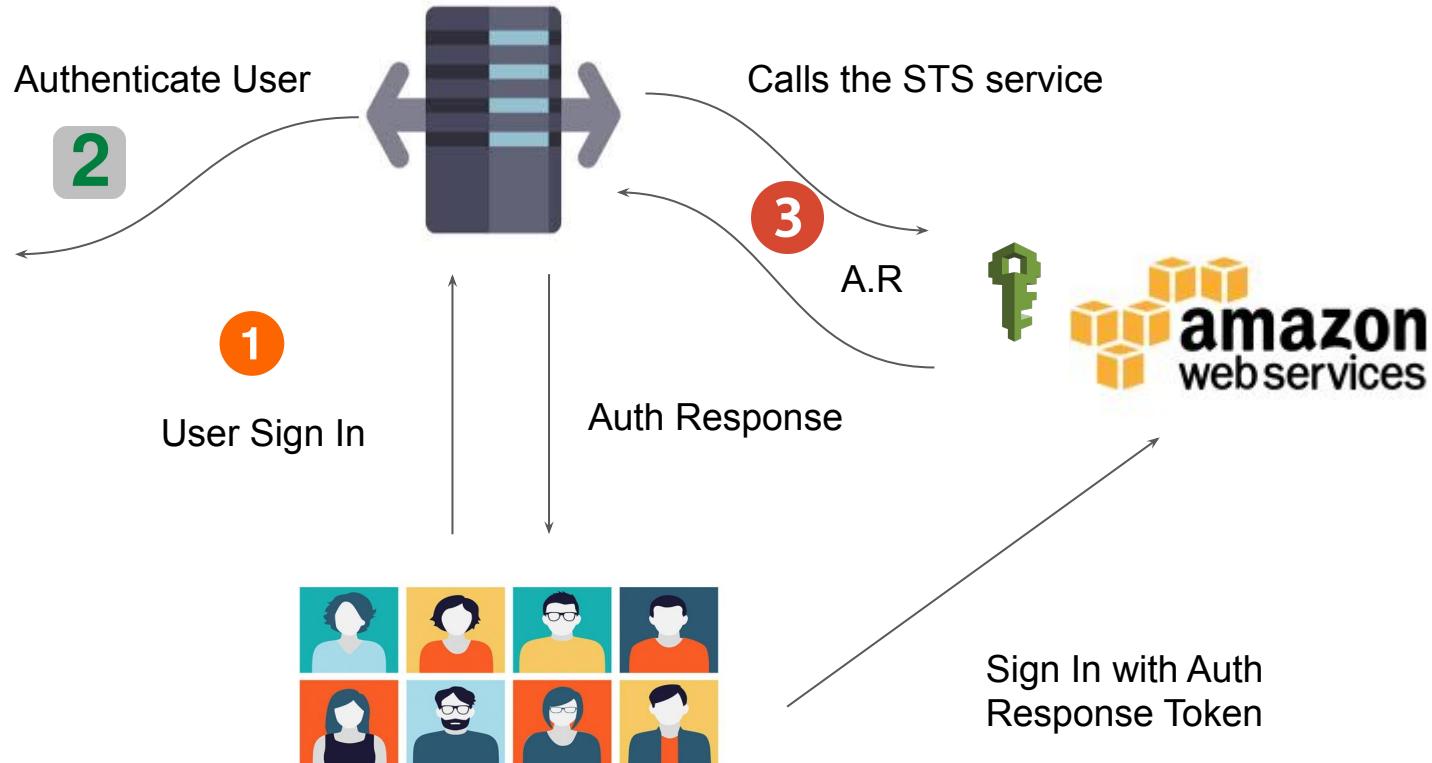
Identity Broker :-

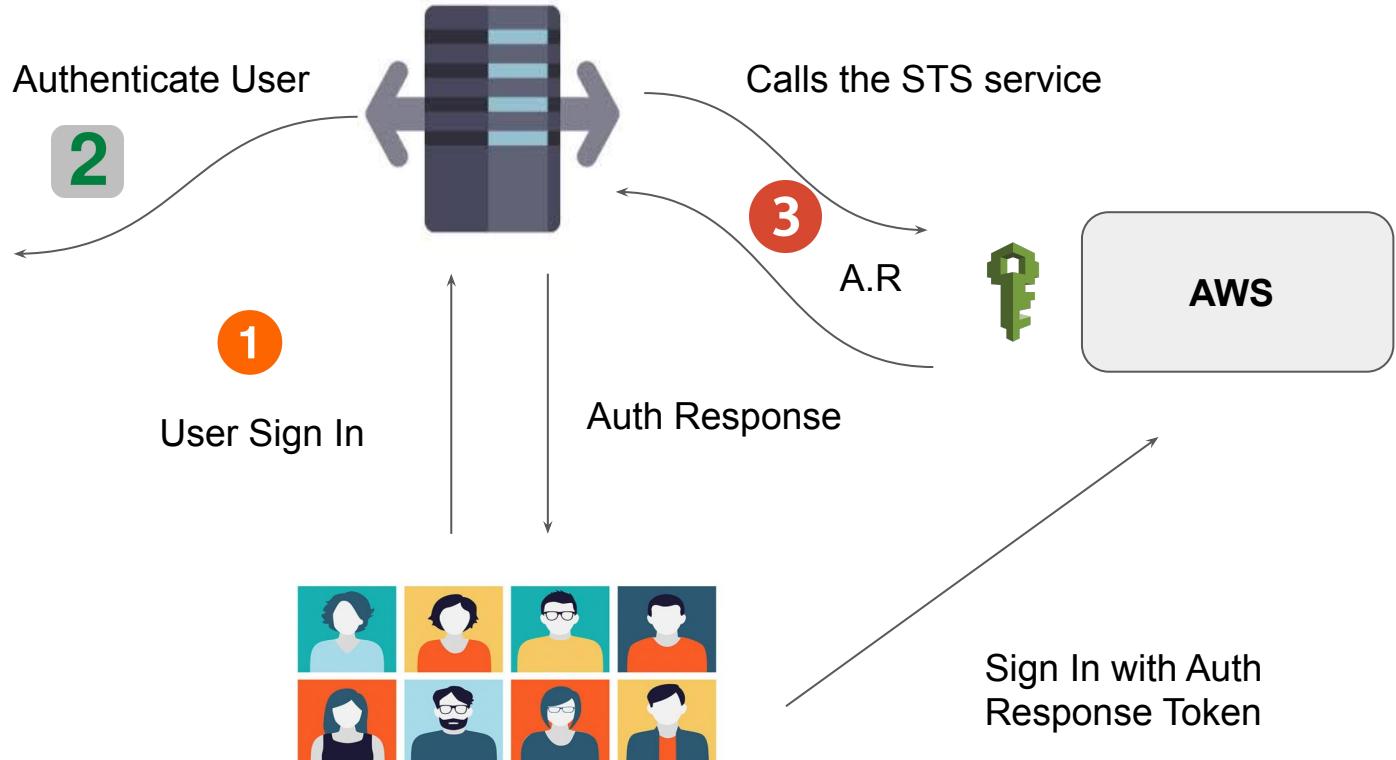
It is an intermediate service which connects multiple providers.

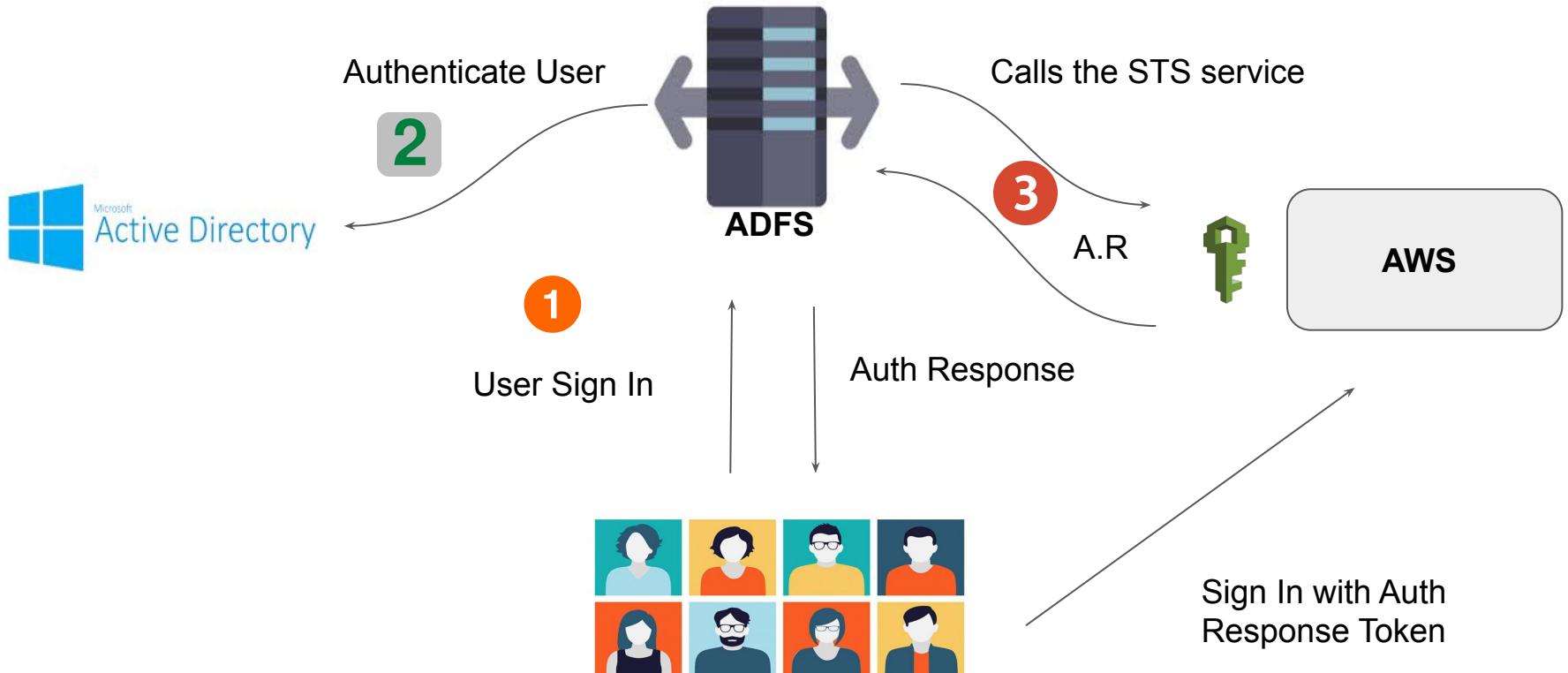




Microsoft
Active Directory







Steps to Remember

- User logs in with username & Password.
- This credentials are given to the Identity Broker.
- Identity Broker validates it against the AD.
- If credentials are valid, Broker will contact the STS token service.
- STS will share the following 4 things :-

Access Key + Secret Key + Token + Duration

- User can now use to login to AWS Console or CLI.

Notations to Remember

Identities : Users

Identity Broker :

- It is a middleware that takes the users from point A & help connect them to point B.

Identity Store :-

- Place where users are present. Eg : AD, IPA, Facebook etc.

SAML

Single Sign On

Introduction to SAML

- SAML stands for Security Assertion Markup Language.
- It is a secure XML based communication mechanism for communicating identities across organizations.
- SAML eliminates the need to maintain multiple authentication credentials, such as passwords in multiple locations.

Classic Way



Challenges with classic way

- The administrator does not have direct visibility with the underlying database of the SAAS provider.
- If there are multiple SAAS providers, it is difficult to keep track of which user has access to which SAAS application.
- When the user leaves the organization, he needs to be removed from all the entities (Jenkins, AWS, HR app)

Different Views

Administrator's View

Have to login to different providers to manage and control the permissions of an individual user across the organization.

User forgetting username and passwords, MFA :(

User's View

I have to remember passwords of all the applications in the organization.

It might be possible that even userID across apps is different, so have to remember that as well.

SAAS Provider's View

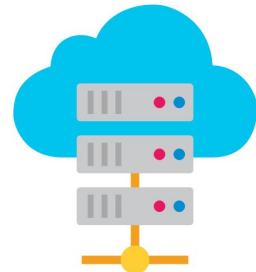
Have to maintain the user and password database of customers.

This is a big security liability.

SAML



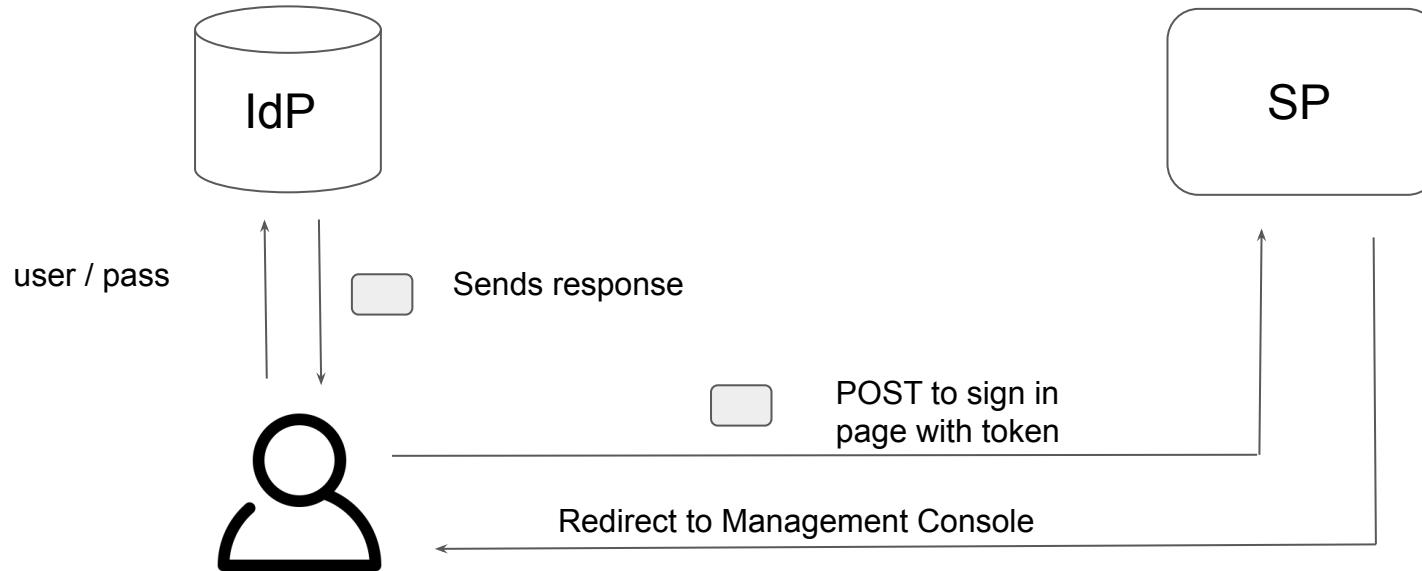
Identity Provider



Service Provider



The SAML Way



Introduction to SAML

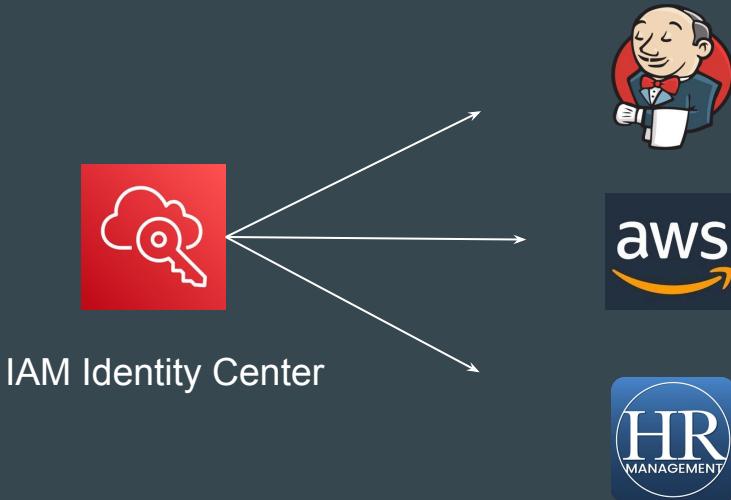
- The flow gets initiated when user opens the IdP URL and enters the username and password and selects the appropriate application.
- IdP will validate the credentials and associated permissions and then user receives SAML assertion from the IdP as part of response.
- User does a POST of that SAML assertion to the SAAS sign in page and SP will validate those assertion.
- On validation, SP will construct relevant temporary credentials, and constructs sign in URL for the console and sends to the user.

IAM Identity Center



Understanding the Basics

IAM Identity Center (successor to AWS Single Sign-On) allows centralized access to multiple AWS accounts and applications and provide users with single sign-on access to all their assigned accounts and applications from one place.



Basic Steps

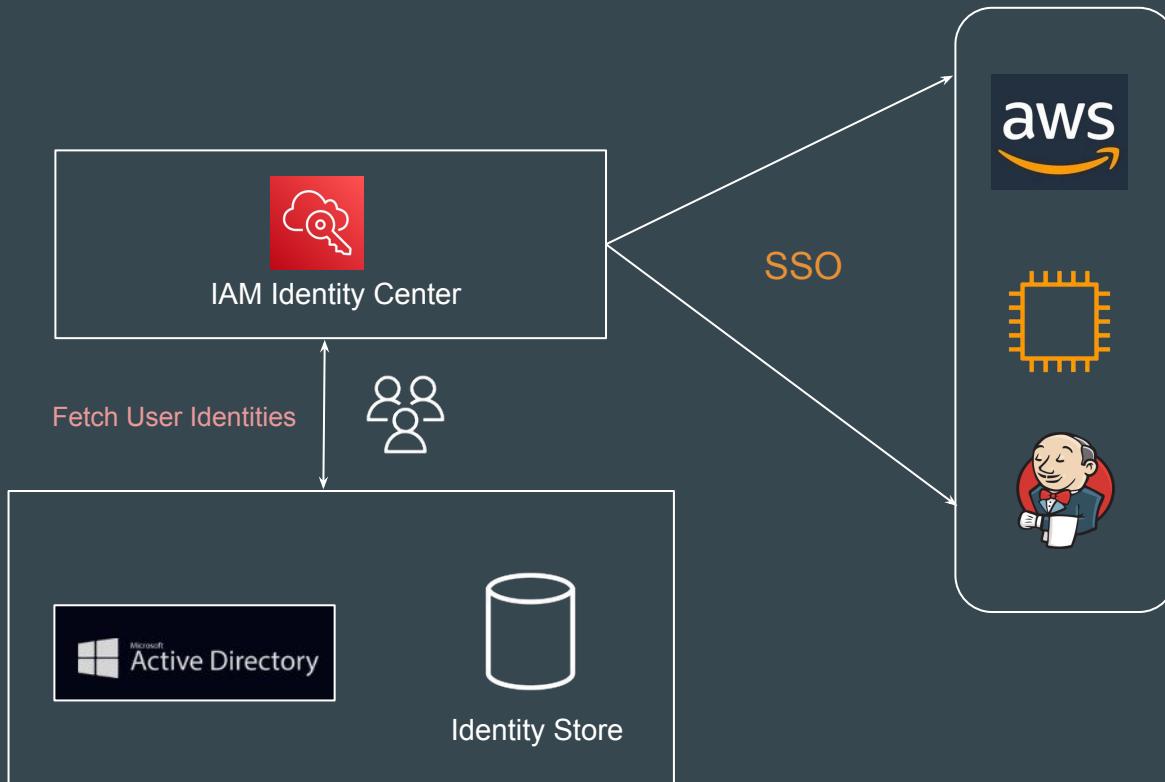


A screenshot of the AWS account selection screen. At the top right is a search bar with the placeholder 'Search'. Below it is a section titled 'AWS Account (2)'. It lists two accounts: 'Europa' (with ID #042025557788) and 'Sandbox Account' (with ID #004417287555). Each account entry has a small orange hexagonal icon and a dropdown arrow to its right.

Login to Access Portal

Connect with AWS Accounts / Apps available

Understanding the Workflow



SSO with AWS CLI

AWS CLI integrates with the SSO.

SSO users can authenticate via CLI, and they will be able to perform the CLI operations without having to add keys in their `~/.aws/credentials` file.

```
C:\Users\zealv>aws s3 ls --profile AdministratorAccess-004417287555
C:\Users\zealv>aws sso login --profile AdministratorAccess-004417287555
Attempting to automatically open the SSO authorization page in your default browser.
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
https://device.sso.us-east-1.amazonaws.com/
Then enter the code:
KQKZ-NRWR
Successfully logged into Start URL: https://d-9067a61937.awsapps.com/start
```

IAM Team After Implementing SSO



Benefits of IAM Identity Center

Your users can use their directory credentials for single sign-on access to multiple AWS accounts.

Enable single sign-on access to your AWS applications

Enable single sign-on access to Amazon EC2 Windows instances

Enable single sign-on access to cloud-based applications that support SAML

IAM Identity Center - Concepts & Considerations



Prerequisite for Identity Center

Your AWS account must be managed by AWS Organizations.

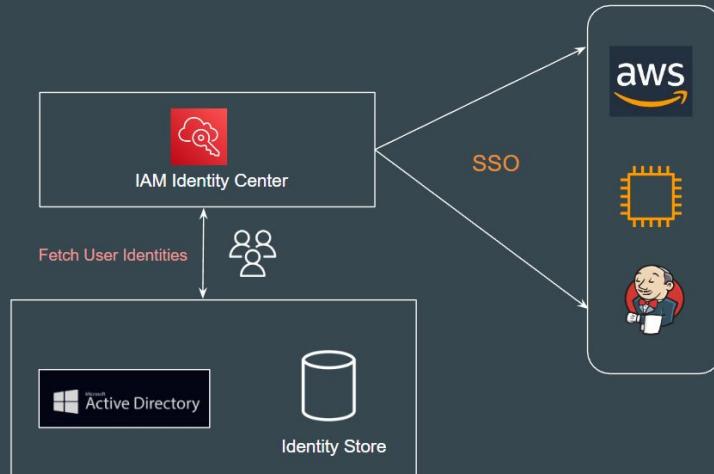
If you've already set up AWS Organizations, make sure that all features are enabled

When you enable IAM Identity Center, you will choose whether to have AWS create an organization for you.

Identity Source

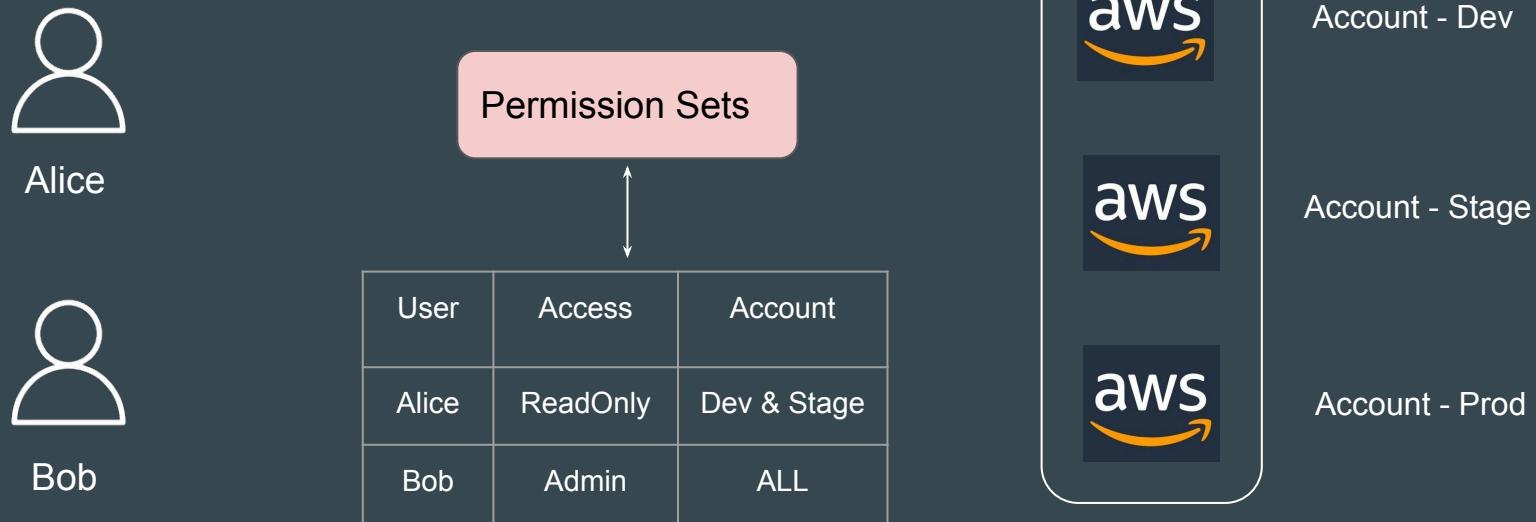
If you're already managing users and groups in Active Directory or an external IdP, it is recommended that you consider connecting this identity source when you enable IAM Identity Center and choose your identity source.

You can also create users and groups directly in IAM Identity Center.

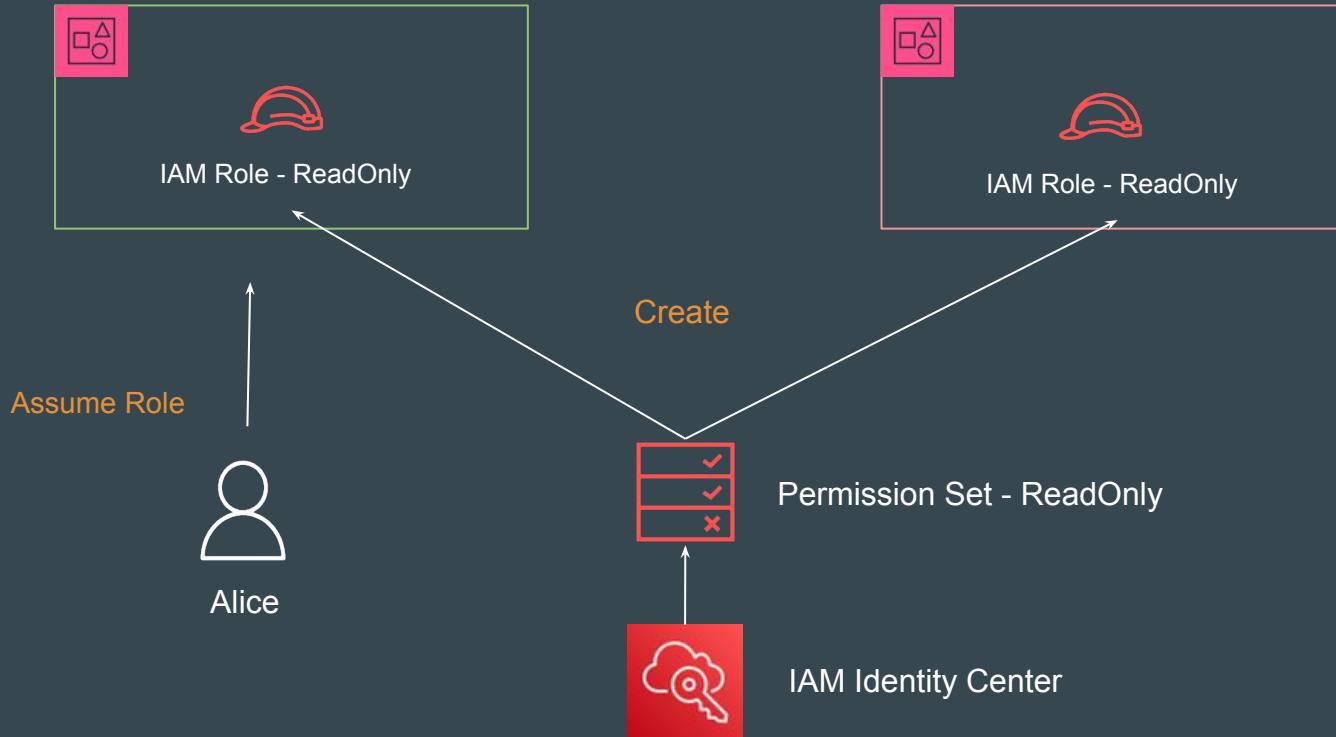


Permission Sets

Permission sets define the level of access that users in IAM Identity Center have to their assigned AWS accounts



How it Works



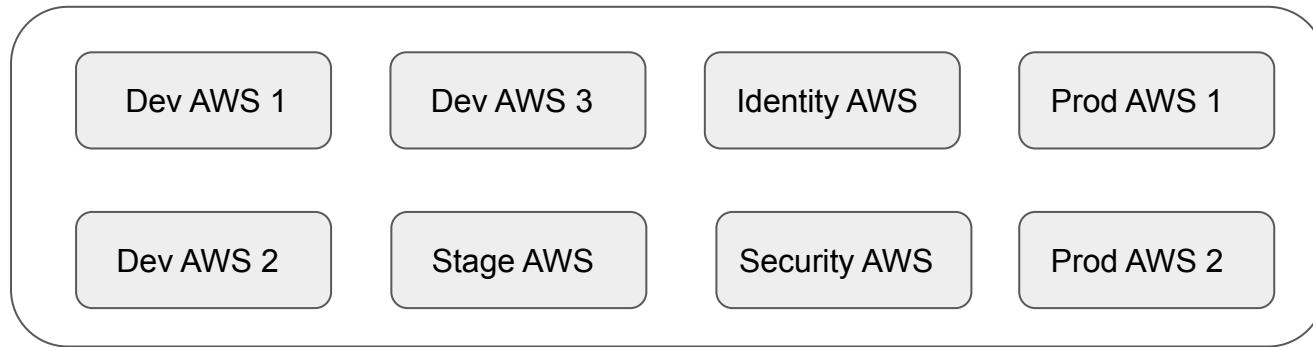
AWS Control Tower

Agility and Governance

Challenges with Multi-Account Environments

Most of the organizations follow a multi-account based architecture.

When the amount of AWS account increases, it leads to own set of challenges.

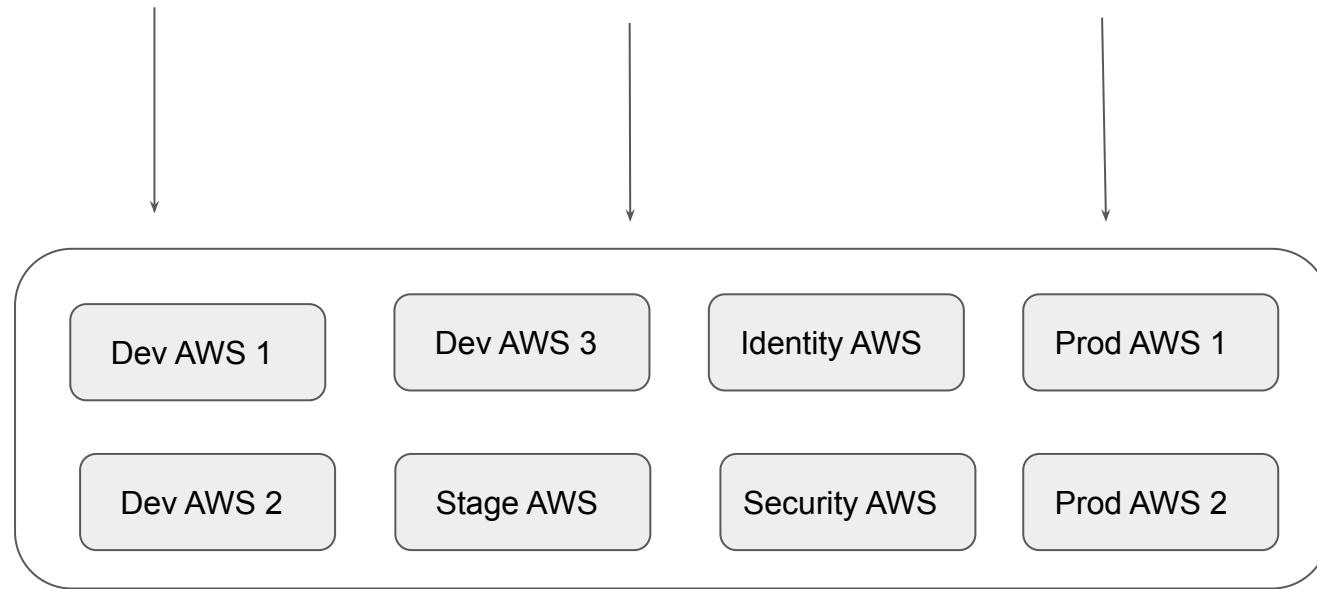


Challenge 1 - Identity Management

username1, password1

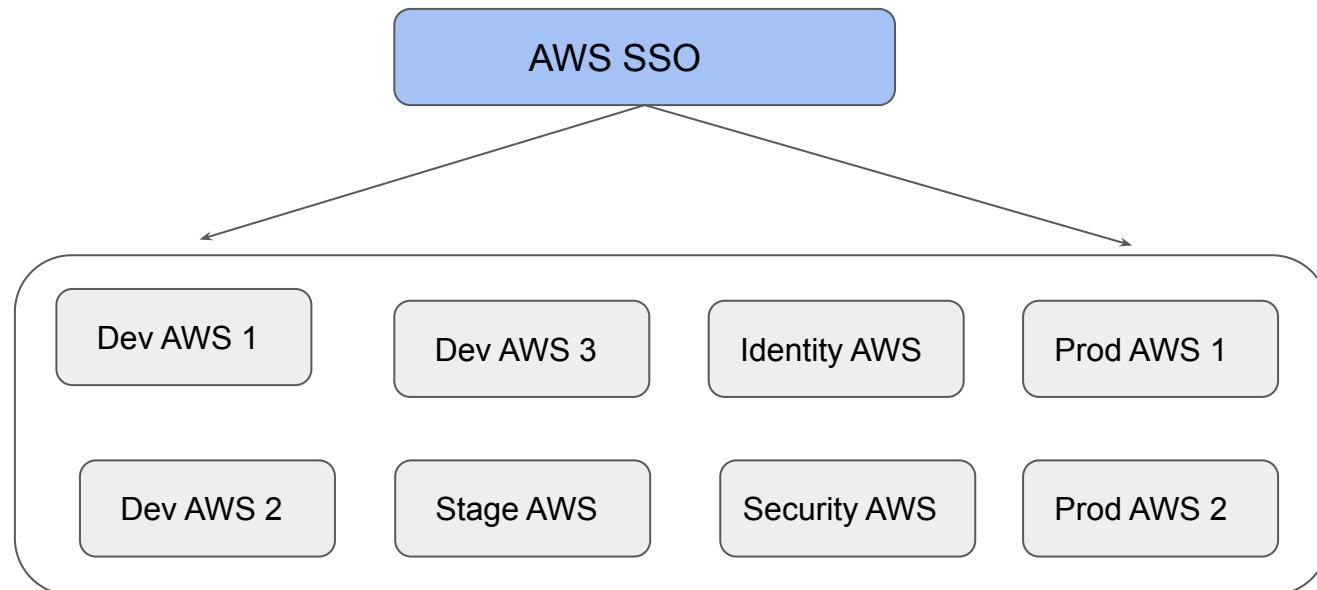
username2, password2

username3 password3



Solution 1 - Single Sign On

Single sign-on (SSO) is an authentication method that enables users to securely authenticate with multiple applications and websites by using just one set of credentials.

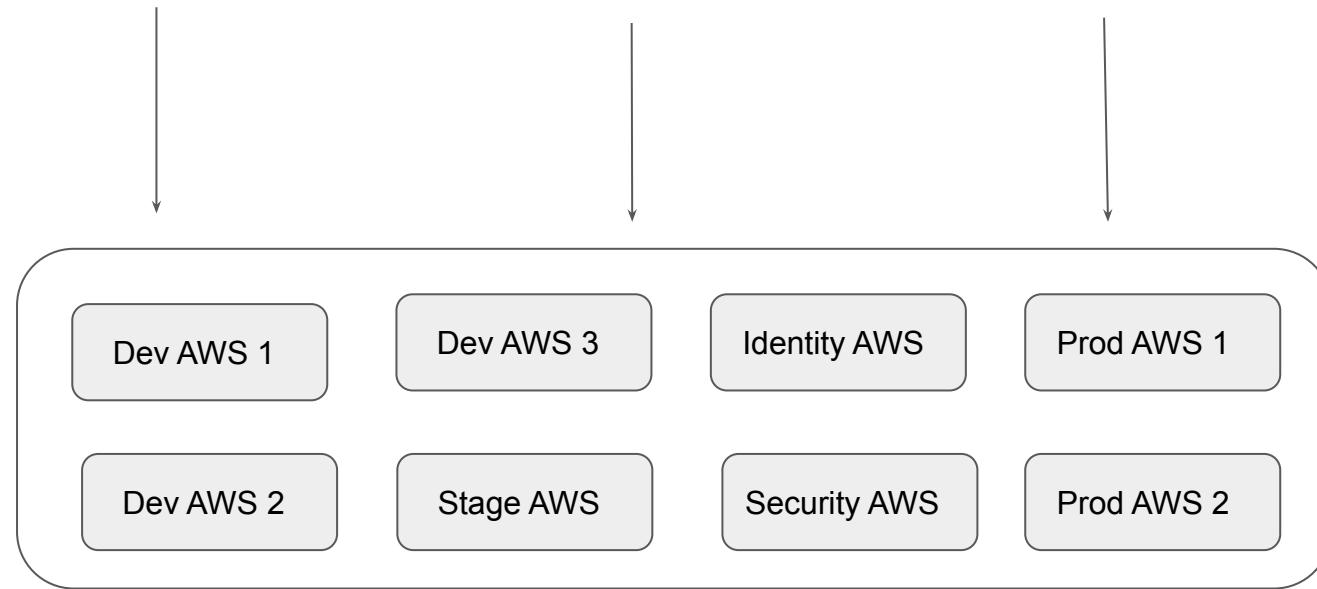


Challenge 2 - Security Hardening

Enable AWS Config

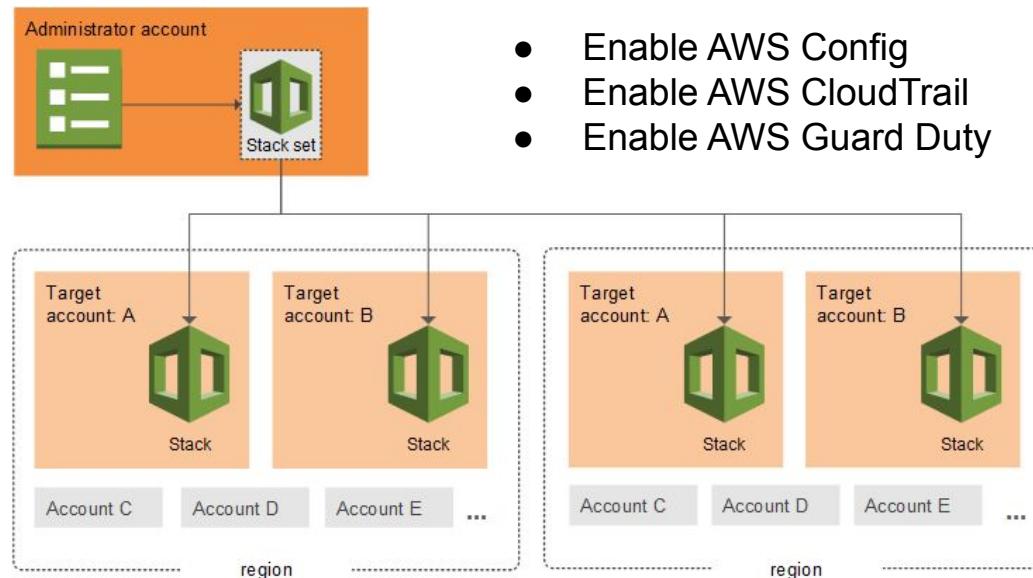
AWS Organizations & SCP

Centralized Logging



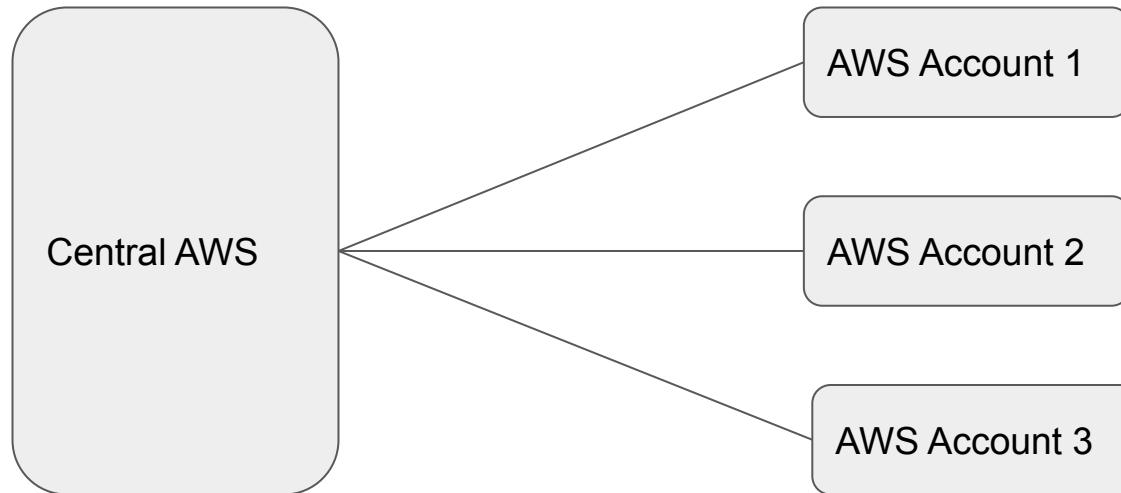
Solution 2 - Security Automation

AWS CloudFormation StackSets allows you to create, update, or delete stacks across multiple accounts and Regions with a single operation



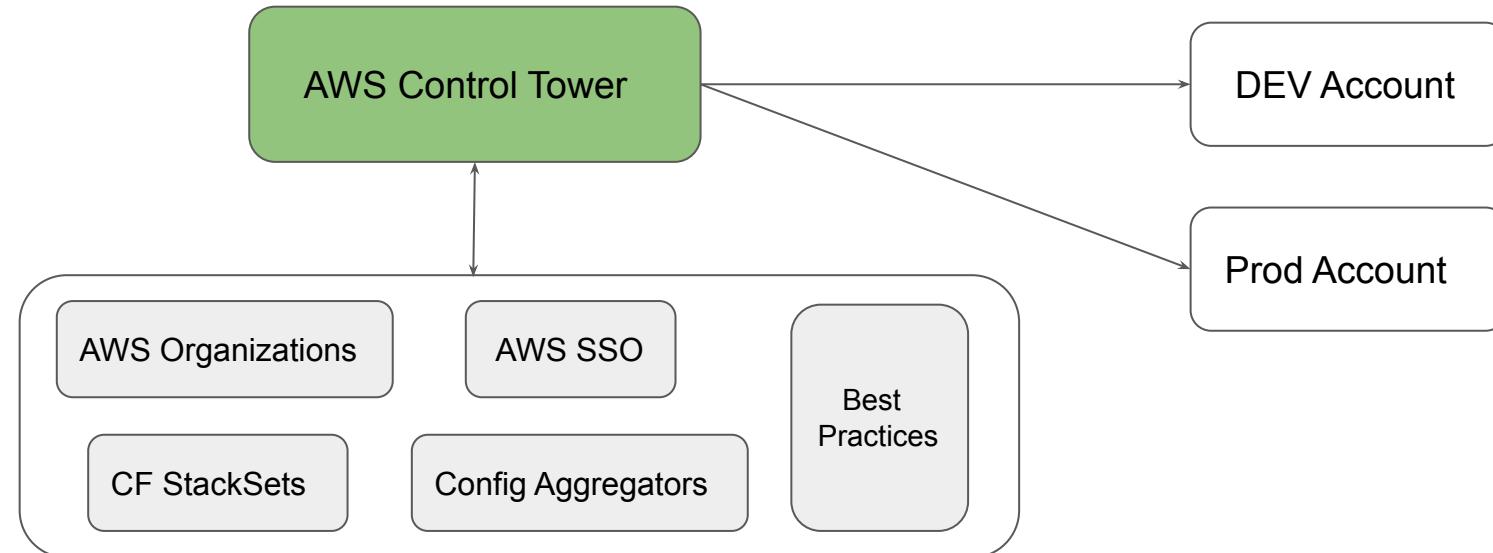
Challenge 3 - Centralized Console

We need to have a centralized console that shows details of all AWS accounts, their security compliance level, and other information



AWS Control Tower

AWS Control Tower offers a straightforward way to set up and govern an AWS multi-account environment, following the best practices.



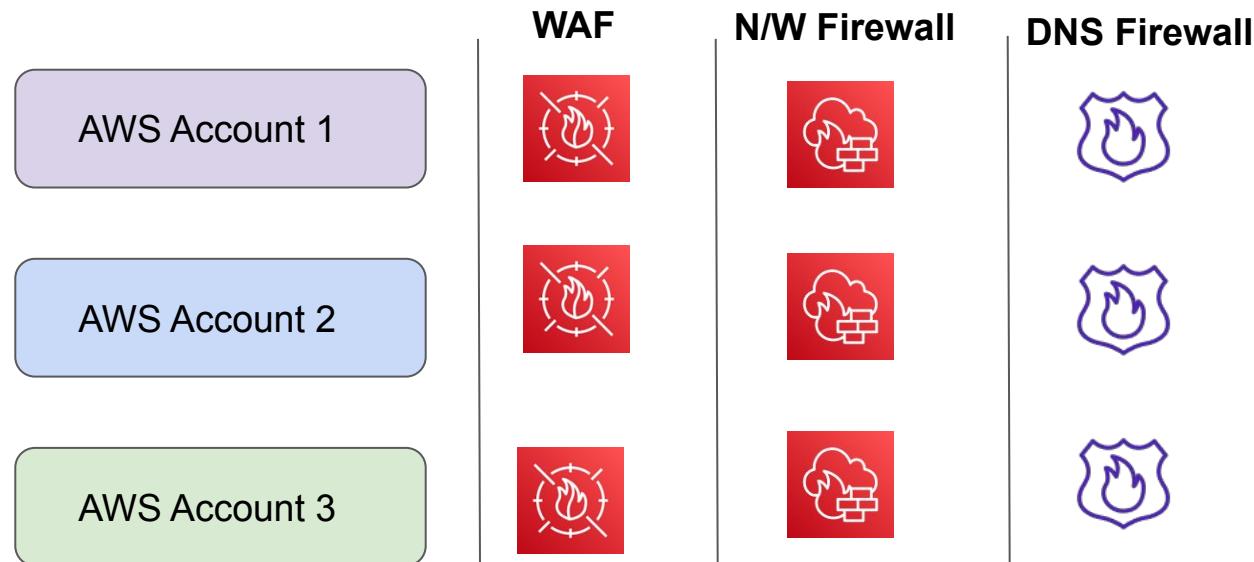
Firewall Manager

Centrally Manage Rules

Understanding the Challenge

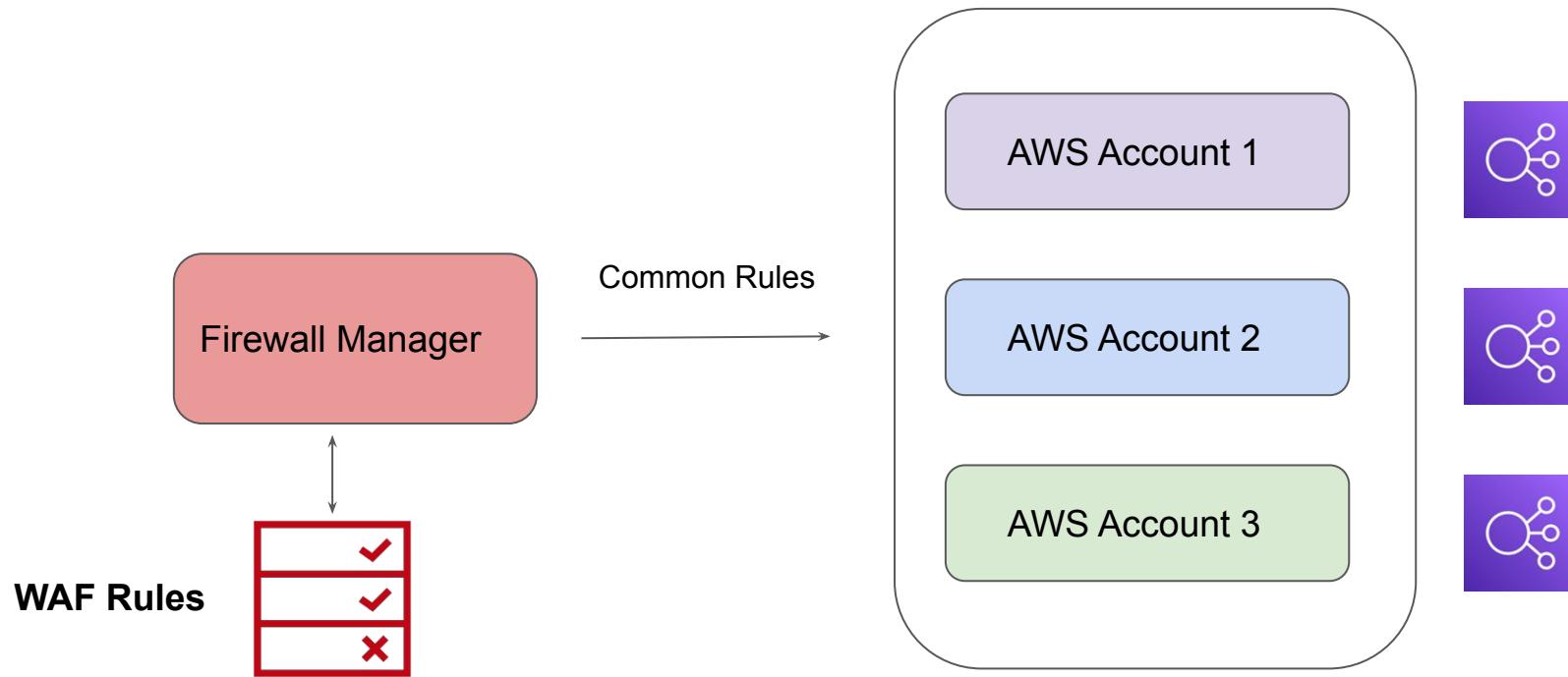
Most of the organizations are opting for Multi-Account based strategy for separation of environments (dev, stage, prod)

Security Team needs to create, maintain and update security services across all of the accounts.



Understanding the Basics

AWS Firewall Manager is a security management service which allows you to centrally configure and manage firewall rules across your accounts and applications in AWS Organizations



Supported Service

Firewall Manager supports wide variety of services, including:

- AWS WAF
- VPC Security Groups
- AWS Network Firewall
- Route53 DNS Firewall
- AWS Shield Advanced
- Palo Alto Cloud Next-generation firewalls

Important Prerequisite: AWS Organizations + AWS Config.

Benefits of Firewall Manager

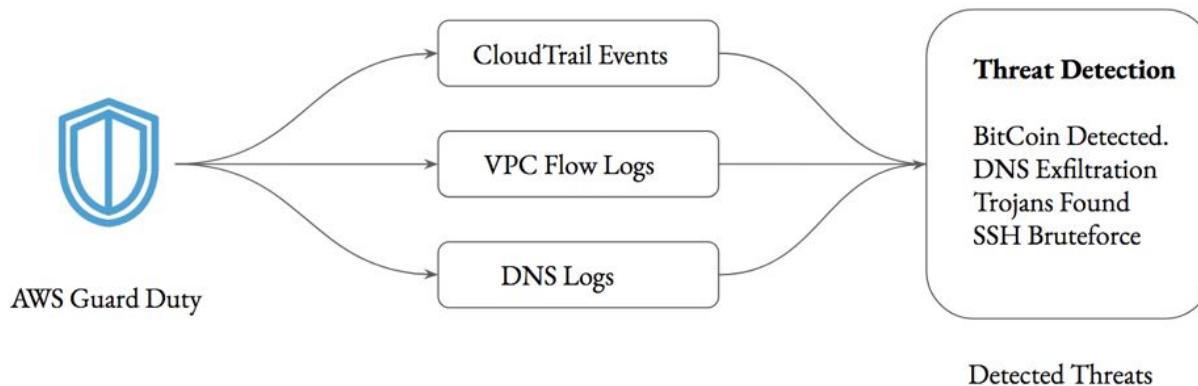
1. Simplify management of firewall rules across your accounts
2. Ensure compliance of existing and new applications
3. Easily deploy managed rules across accounts
4. Centrally deploy protections for your VPCs

AWS GuardDuty

Let's start Rolling !

Understanding GuardDuty

AWS Guard Duty is a threat intelligence service by AWS which monitors for malicious behavior to help customers protect their AWS workloads.



GuardDuty Findings

Findings 

Showing 67 of 67

9

47

11

Actions		Finding type	Resource	Last seen	Account	Co...
<input type="checkbox"/>	 [SAMPLE] UnauthorizedAccess:EC2/RDPBruteForce	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] Trojan:EC2/PhishingDomainRequest!DNS	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] CryptoCurrency:EC2/BitcoinTool.B!DNS	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] Trojan:EC2/BlackholeTraffic!DNS	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] UnauthorizedAccess:EC2/SSHBruteForce	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] UnauthorizedAccess:EC2/TorIPCaller	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] UnauthorizedAccess:IAMUser/MaliciousIPCaller	AccessKey: GeneratedFindingAccess		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] Recon:EC2/Portscan	Instance: i-99999999		2 months ago	101586075...	1
<input type="checkbox"/>	 [SAMPLE] UnauthorizedAccess:EC2/MaliciousIPCaller.C...	Instance: i-99999999		2 months ago	101586075...	1

Important Pointers

Guard Duty will only monitor the Route53 for DNS Logs.

Lot of organizations makes use of Active Directory DNS. The logs from these servers will not be monitored.

Relax and Have a Meme Before Proceeding

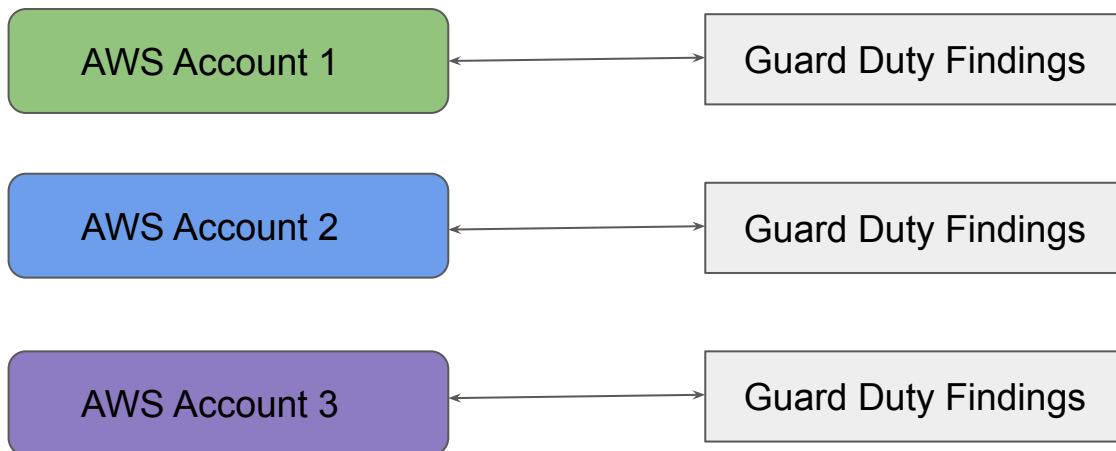


Managing GuardDuty Findings Centrally

Multiple AWS Accounts

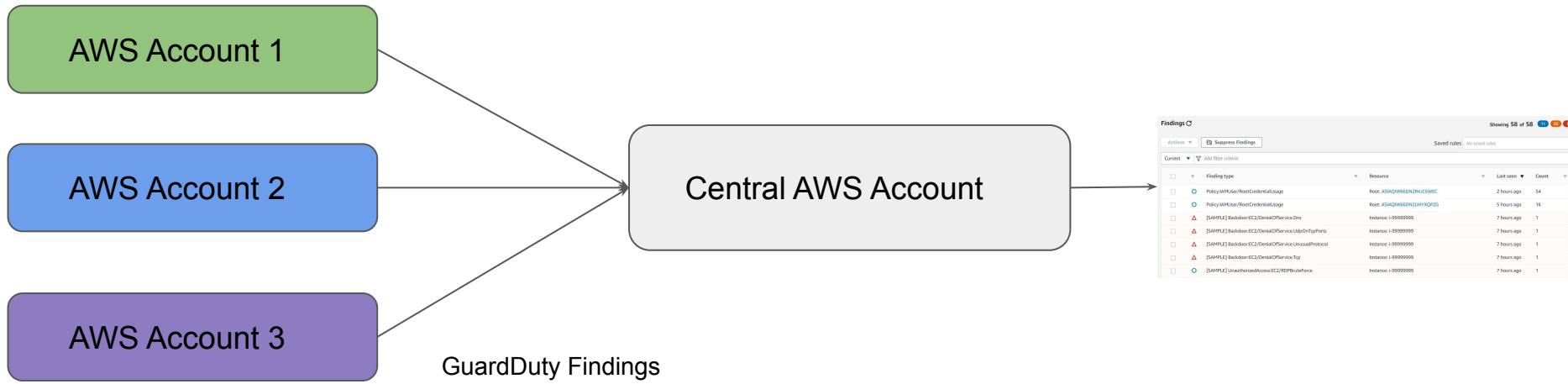
Understanding the Challenge

- Enabling GuardDuty across all AWS account is recommended.
- Checking findings across individual account is troublesome.



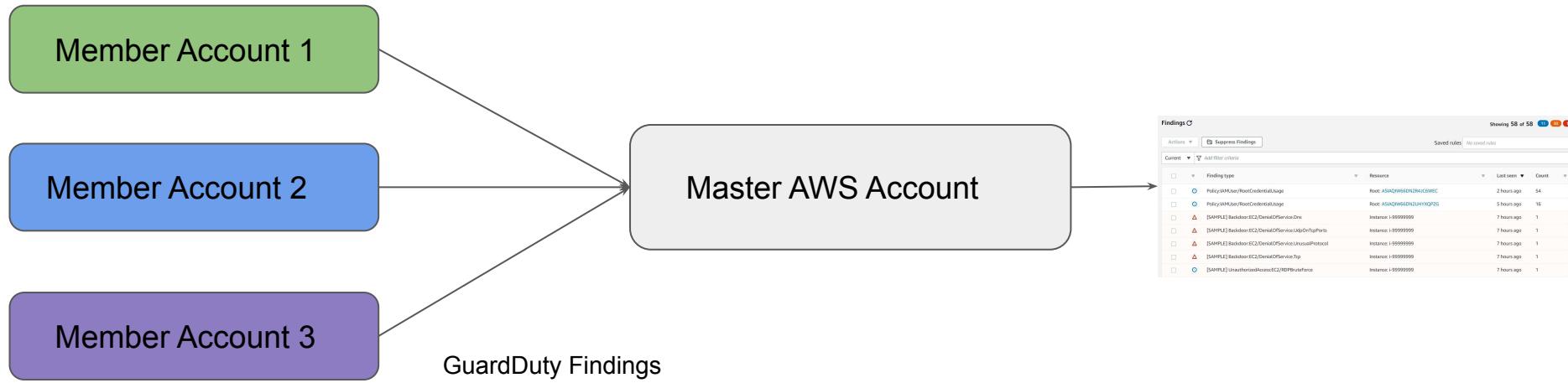
Central Architecture

In this architecture, the Guard Duty findings from all the AWS Accounts will be sent to the Central AWS account.



Right Terminology

In this architecture, the Guard Duty findings from all the AWS Accounts will be sent to the Central AWS account.



Amazon Macie

Machine Learning based Security

Core Feature of Macie

S3 might contain sensitive information like PII data, database backups, SSL private keys and various others.

Amazon Macie **makes use of machine learning** to identify sensitive data stored in AWS.

Policy findings		C
Most recent policy findings		
High	Policy:IAMUser/S3BucketReplicatedExternally	1 minute ago
High	Policy:IAMUser/S3BlockPublicAccessDisabled	1 minute ago
High	Policy:IAMUser/S3BucketSharedExternally	1 minute ago
Medium	Policy:IAMUser/S3BucketEncryptionDisabled	1 minute ago
High	Policy:IAMUser/S3BucketPublic	1 minute ago

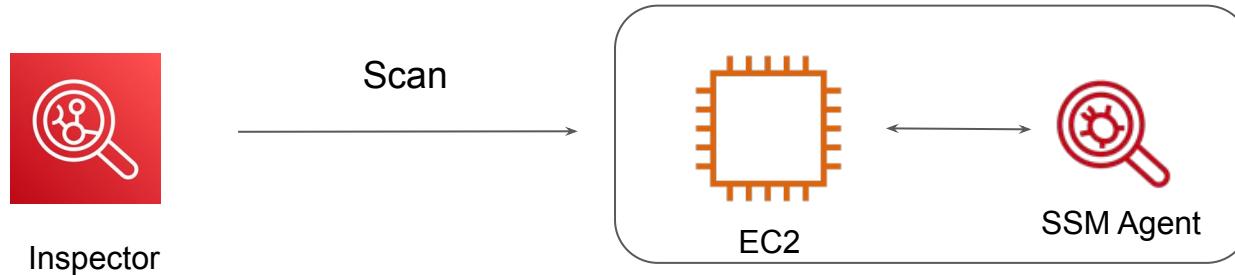
AWS Inspector

Vulnerability Scanner

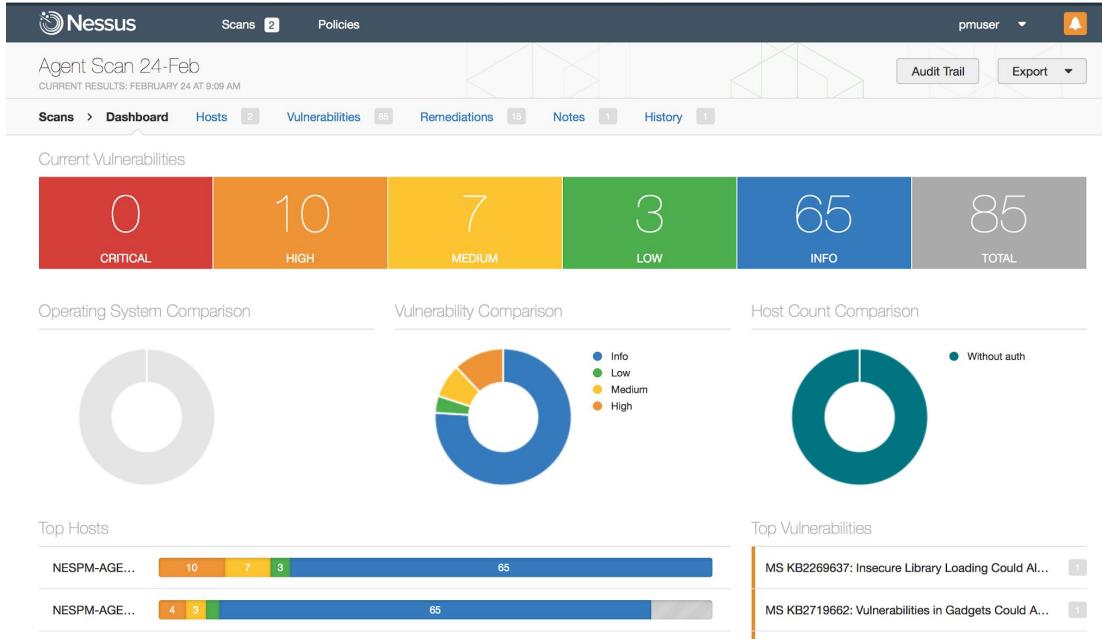
Basics of AWS Inspector

AWS Inspector is similar to a vulnerability scanner which will scan the system for specific vulnerabilities and provide the results.

It relies on the agent installed on the server to scan the server.

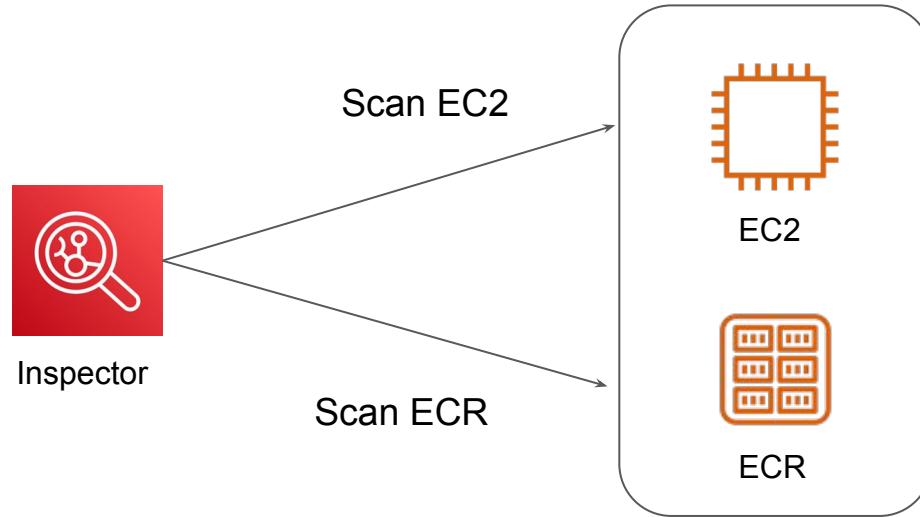


Similar to Nessus



Supported Scans

Amazon Inspector gives you the flexibility to enable either EC2 scanning or ECR container image scanning, or both.



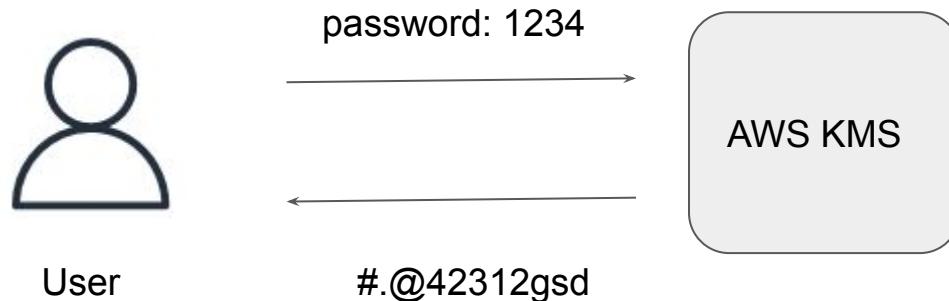
AWS KMS

Do things the right way

Basics of KMS

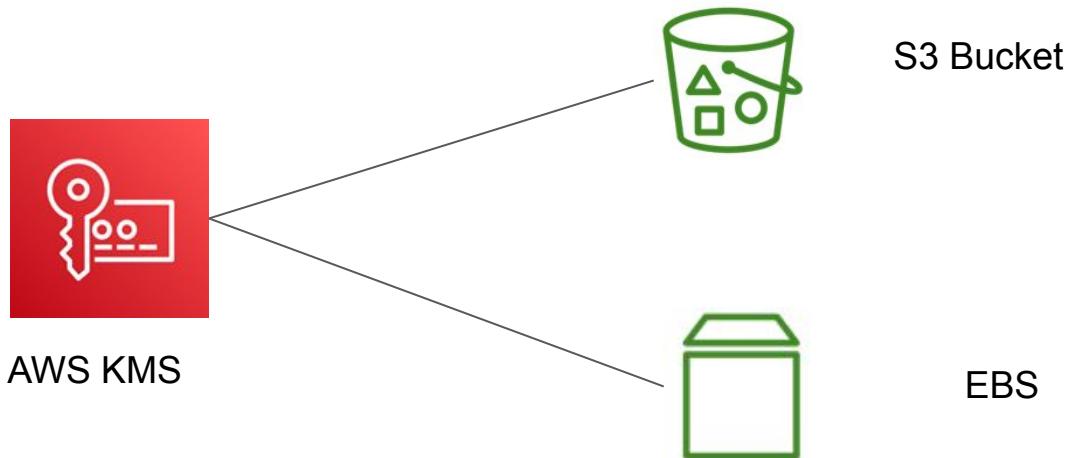
AWS KMS stands for AWS Key Management Service.

This service provides capability to encrypt and decrypt the data.



Integration of KMS

AWS KMS also integrates with various AWS services like S3, DynamoDB, EBS and others.



KMS Practical

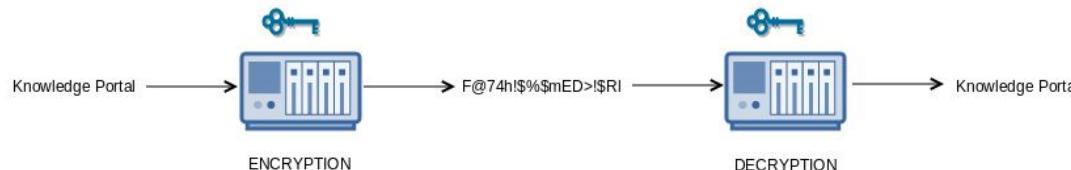
Time to Defend Easily

Revising Cryptography Concepts

Plaintext can refer to anything which humans can understand and/or relate to. This may be as simple as English sentences or even Python code.

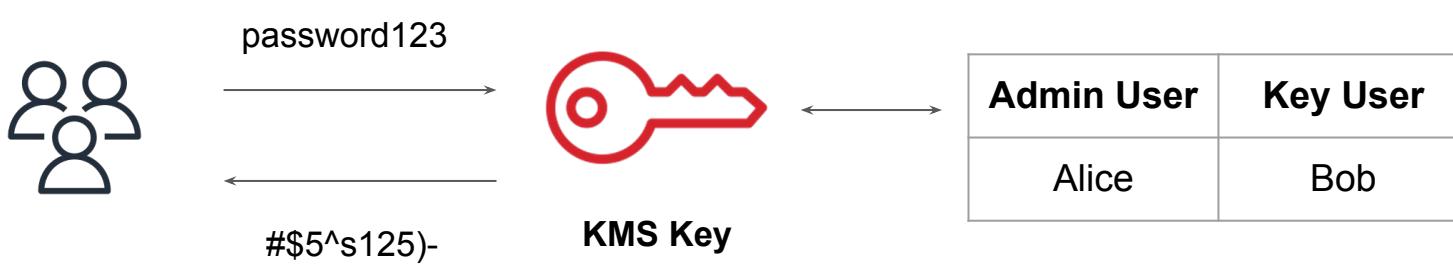
Ciphertext, or encrypted text, is a series of randomized letters and numbers which humans cannot make any sense of.

An encryption algorithm is step by step approach that tells on how the PT will be converted to the CipherText.



KMS Practical Workflow

1. Create a Customer Managed Key (CMK)
2. Define the Administrative User & Key User.
3. Encrypt and Decrypt data with the CMK.



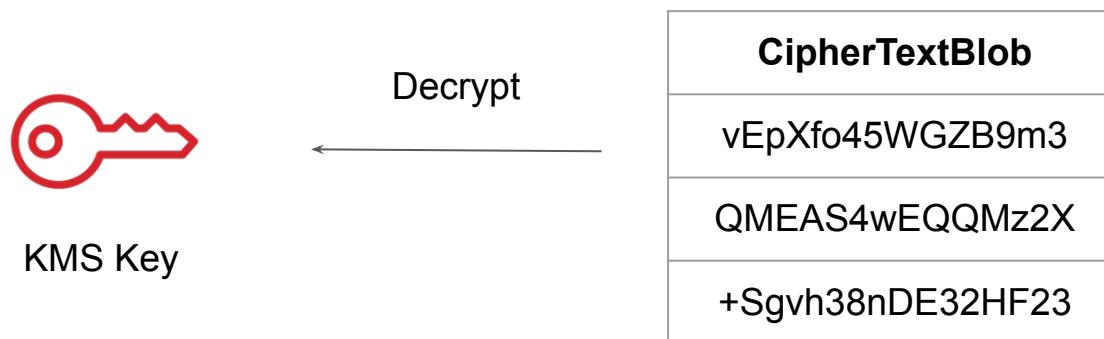
Schedule Key Deletion

Delete the KMS Key

Deleting Key in KMS

Deleting KMS key is destructive and potentially dangerous and an irreversible process.

After a KMS key is deleted, you can no longer decrypt the data that was encrypted under that KMS key, which means that data becomes unrecoverable.



Important Note

You should delete a KMS key only when you are sure that you don't need to use it anymore.

If you are not sure, consider disabling the KMS key instead of deleting it.

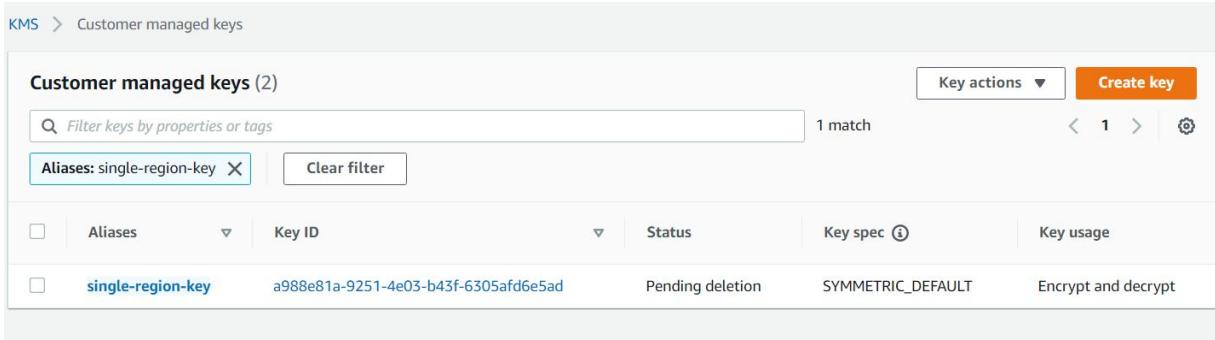
If you disable a KMS key, it cannot be used to encrypt or decrypt data until you re-enable it.

You can re-enable a disabled KMS key if you need to use it again later

Waiting Period for Key Deletion

Because it is destructive and potentially dangerous to delete a KMS key, AWS KMS requires you to set a waiting period of 7 – 30 days. The default waiting period is 30 days.

During the waiting period, A KMS key pending deletion cannot be used in any cryptographic operations.



The screenshot shows the AWS KMS console interface. The top navigation bar has 'KMS' selected. Below it, the title 'Customer managed keys (2)' is displayed. A search bar contains the placeholder 'Filter keys by properties or tags'. To the right of the search bar are 'Key actions' and a 'Create key' button. Below the search bar, there is a filter section with 'Aliases: single-region-key' and a 'Clear filter' button. The main table lists two keys:

Aliases	Key ID	Status	Key spec	Key usage
<input type="checkbox"/> single-region-key	a988e81a-9251-4e03-b43f-6305af6e5ad	Pending deletion	SYMMETRIC_DEFAULT	Encrypt and decrypt

IAM Access Analyzer



Understanding the Basics

AWS IAM Access Analyzer provides the following capabilities:

- IAM Access Analyzer helps identify resources in your organization and accounts that are shared with an external entity.
- IAM Access Analyzer validates IAM policies against policy grammar and best practices.
- IAM Access Analyzer generates IAM policies based on access activity in your AWS CloudTrail logs.

Capability 1 - Identify Shared Resource

IAM Access Analyzer helps you identify the resources in your organization and accounts, such as Amazon S3 buckets or IAM roles, **shared with an external entity**.



Supported Resource Types

IAM Access Analyzer analyzes the following resource types:

- Amazon Simple Storage Service buckets
- AWS Identity and Access Management roles
- AWS Key Management Service keys
- AWS Lambda functions and layers
- Amazon Simple Queue Service queues
- AWS Secrets Manager secrets
- Amazon Simple Notification Service topics
- Amazon Elastic Block Store volume snapshots
- Amazon Relational Database Service DB snapshots
- Amazon Relational Database Service DB cluster snapshots
- Amazon Elastic Container Registry repositories
- Amazon Elastic File System file systems

Points to Note

For each instance of a resource shared outside of your account, IAM Access Analyzer generates a finding

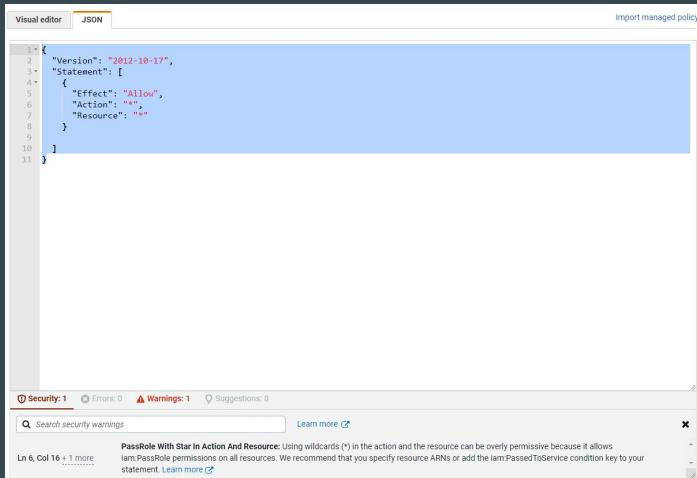
You can review findings to determine if the access is intended and safe or if the access is unintended and a security risk

Active findings						
Account ID 042025557788						
<input type="text"/> Filter active findings						
<input type="checkbox"/>	Finding ID	Resource	External principal	Condition	Shared through	Access level
<input type="checkbox"/>	95a5821b-bb83-4dd...	EC2 Snapshot snapshot/snap-02e015523dca9a4de	AWS Account 004417287555	-	-	Write, Read, List
<input type="checkbox"/>	17834d48-adda-407...	IAM Role Cross-Account-Role	AWS Account 004417287555	-	-	Write
<input type="checkbox"/>	8bf8920b-36ef-4a65...	S3 Bucket cross-account-demo-s3-bucket	All Principals	Source IP 101.0.63.213/32	Bucket policy	Read

Capability 2 - Validating IAM Policy

IAM Access Analyzer validates your policy against IAM policy grammar and best practices.

You can view policy validation check findings that include security warnings, errors, general warnings, and suggestions for your policy.



IAM Access Analyzer



Understanding the Basics

AWS IAM Access Analyzer provides the following capabilities:

- IAM Access Analyzer helps identify resources in your organization and accounts that are shared with an external entity.
- IAM Access Analyzer validates IAM policies against policy grammar and best practices.
- IAM Access Analyzer generates IAM policies based on access activity in your AWS CloudTrail logs.

Capability 1 - Identify Shared Resource

IAM Access Analyzer helps you identify the resources in your organization and accounts, such as Amazon S3 buckets or IAM roles, **shared with an external entity**.



Supported Resource Types

IAM Access Analyzer analyzes the following resource types:

- Amazon Simple Storage Service buckets
- AWS Identity and Access Management roles
- AWS Key Management Service keys
- AWS Lambda functions and layers
- Amazon Simple Queue Service queues
- AWS Secrets Manager secrets
- Amazon Simple Notification Service topics
- Amazon Elastic Block Store volume snapshots
- Amazon Relational Database Service DB snapshots
- Amazon Relational Database Service DB cluster snapshots
- Amazon Elastic Container Registry repositories
- Amazon Elastic File System file systems

Points to Note

For each instance of a resource shared outside of your account, IAM Access Analyzer generates a finding

You can review findings to determine if the access is intended and safe or if the access is unintended and a security risk

Active findings						
Account ID 042025557788						
<input type="text"/> Filter active findings						
<input type="checkbox"/>	Finding ID	Resource	External principal	Condition	Shared through	Access level
<input type="checkbox"/>	95a5821b-bb83-4dd...	EC2 Snapshot snapshot/snap-02e015523dca9a4de	AWS Account 004417287555	-	-	Write, Read, List
<input type="checkbox"/>	17834d48-adda-407...	IAM Role Cross-Account-Role	AWS Account 004417287555	-	-	Write
<input type="checkbox"/>	8bf8920b-36ef-4a65...	S3 Bucket cross-account-demo-s3-bucket	All Principals	Source IP 101.0.63.213/32	Bucket policy	Read

Capability 3 - Generate IAM Policy

IAM Access Analyzer analyzes your AWS CloudTrail logs to identify actions and services that have been used by an IAM entity (user or role) within your specified date range.

It then generates an IAM policy that is based on that access activity.

Generate policy for demo-user
Generate a policy based on the CloudTrail activity for this user.

Time period and permissions to analyze CloudTrail events

Select time period

Last 1 day(s)

Specific dates
Choose a range of up to 90 days.

CloudTrail access

CloudTrail trail to be analyzed
Specify the CloudTrail trail that logs events for this account

US East (N. Virginia)

To analyze this role's access activity, IAM uses the service role below on your behalf to access the specified trail.

Create and use a new service role

Use an existing service role
There are no suitable roles existing.

[View permission details](#)

[Cancel](#)

Capability 3 - Generate IAM Policy

IAM Access Analyzer analyzes your AWS CloudTrail logs to identify actions and services that have been used by an IAM entity (user or role) within your specified date range.

It then generates an IAM policy that is based on that access activity.

Generate policy for demo-user
Generate a policy based on the CloudTrail activity for this user.

Time period and permissions to analyze CloudTrail events

Select time period

Last 1 day(s)

Specific dates
Choose a range of up to 90 days.

CloudTrail access

CloudTrail trail to be analyzed
Specify the CloudTrail trail that logs events for this account

US East (N. Virginia)

To analyze this role's access activity, IAM uses the service role below on your behalf to access the specified trail.

Create and use a new service role

Use an existing service role
There are no suitable roles existing.

[View permission details](#)

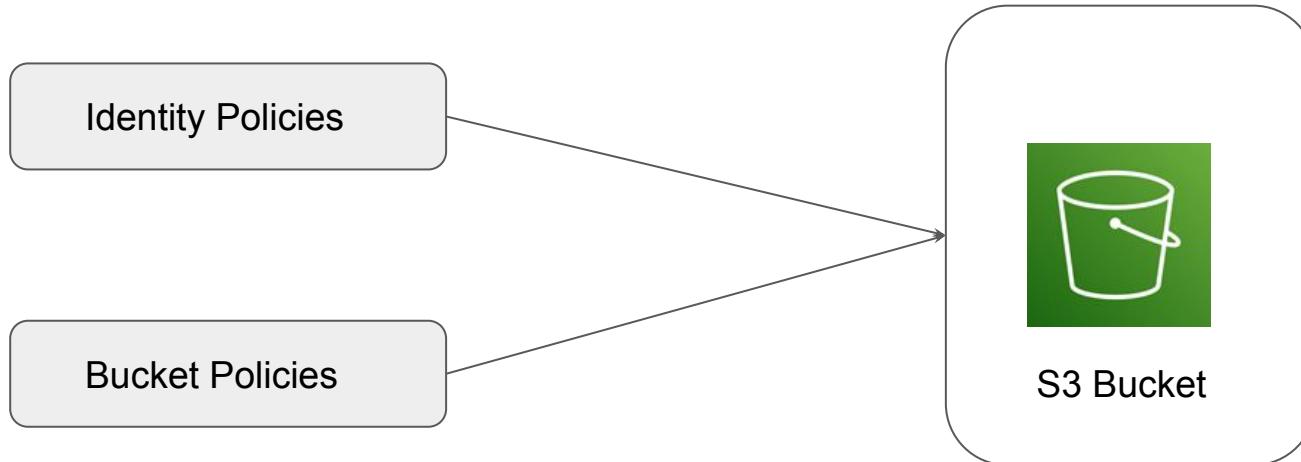
[Cancel](#)

S3 Bucket Policy

Bucket Policies

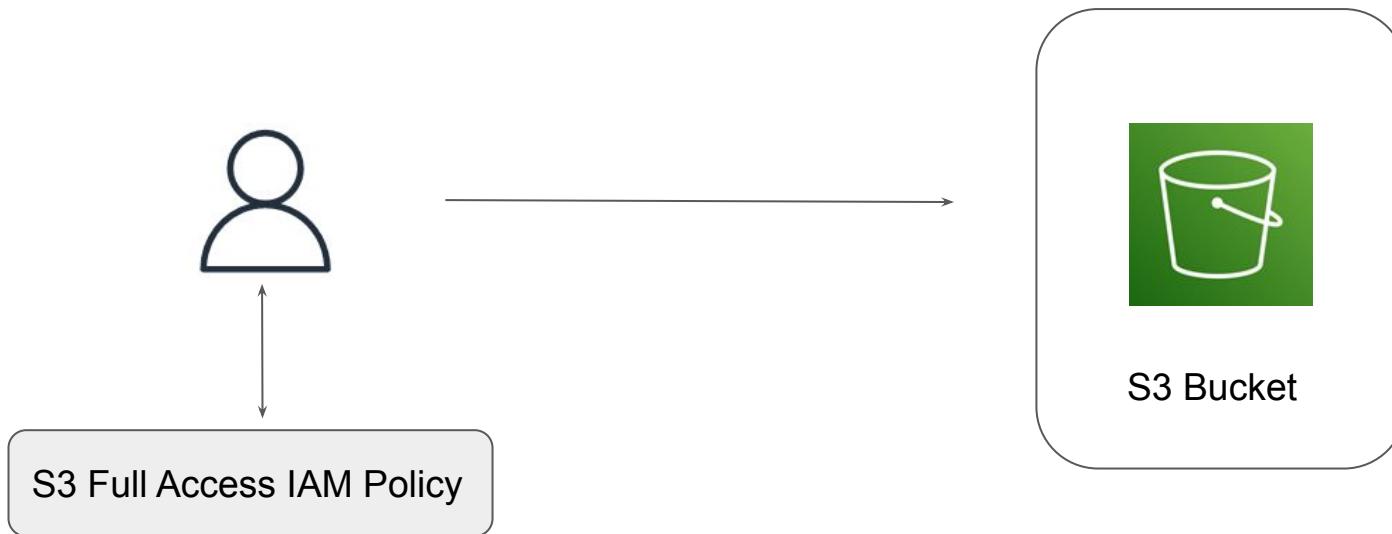
Granting Permission for S3 Resource

There are two primary ways in which a permission to a S3 resource is granted.



Use-Case 1: IAM User Needs Access to S3 Bucket

IAM User Named Bob needs Full Access to S3 Bucket.



Wider Scope of S3 Bucket

Files within the S3 bucket can have scope beyond the IAM entity.

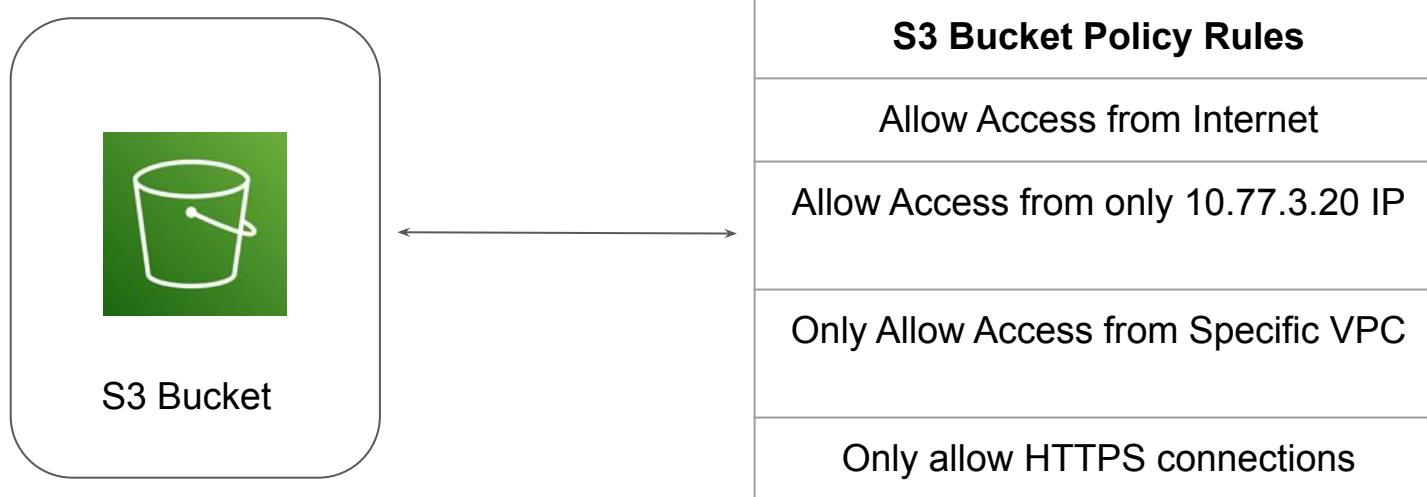
Organization can host entire websites in S3 Bucket.

S3 Buckets can even be used to host central files for download.



S3 Bucket Policy

A bucket policy is a resource-based AWS IAM policy associated with the S3 Bucket to control access permissions for the bucket and the objects in it .



Bucket Policy 1 - Public Access

The following example policy grants the s3:GetObject permission to any public anonymous users.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "PublicRead",  
            "Effect": "Allow",  
            "Principal": "*",  
            "Action": ["s3:GetObject"],  
            "Resource": ["arn:aws:s3:::demo-bucket/*"]  
        }  
    ]  
}
```

Bucket Policy 2 - Only HTTPS

Only the HTTPS requests should be allowed. All HTTP requests should be blocked.

```
{  
    "Id": "ExamplePolicy",  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowSSLRequests",  
            "Action": "s3:GetObject",  
            "Effect": "Allow",  
            "Resource": [  
                "arn:aws:s3:::demo-bucket/*"  
            ],  
            "Condition": {  
                "Bool": {  
                    "aws:SecureTransport": "true"  
                }  
            },  
            "Principal": "*"  
        }  
    ]  
}
```

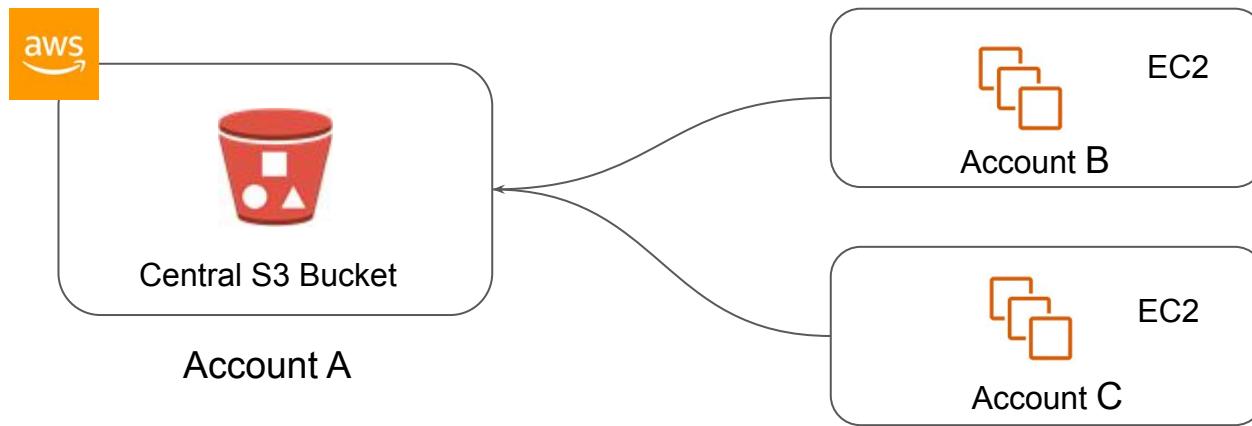
Cross Account S3 Access

Bucket Policies

Cross Account S3 Access

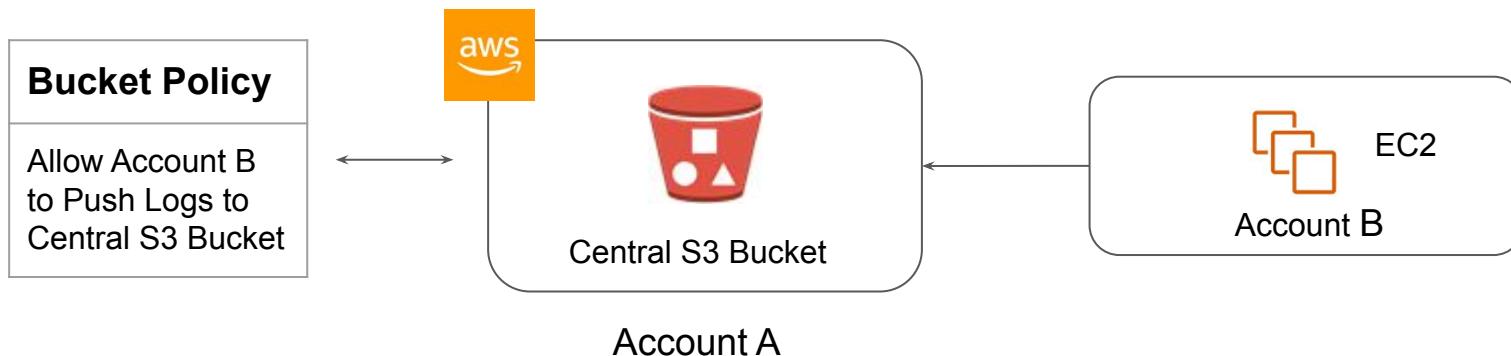
There are many requirements where logs across all AWS accounts need to be stored in a central account.

These logs can include, CloudTrail, CloudWatch, Application Logs, and others.



Creating Bucket Policy

The recommended approach is to add a Bucket Policy in the Central S3 bucket and allow the Account B to push the logs.



Bucket Policy Example - Central S3 Account

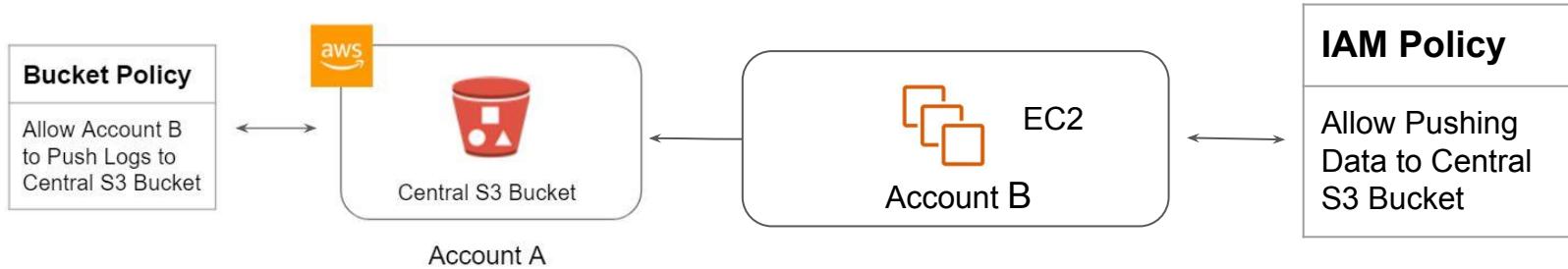
```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "arn:aws:iam::453314488441:root"  
            },  
            "Action": [  
                "s3:GetObject",  
                "s3:PutObject",  
                "s3:PutObjectAcl"  
            ],  
            "Resource": [  
                "arn:aws:s3::::central-s3-bucket/*"  
            ]  
        }  
    ]  
}
```

←———— Account B ARN

←———— Central S3 Bucket

Part 2- Permission on Account B Side

The resource in the Account B also needs to have permission to push the logs to Central Account S3 Bucket.



IAM Policy - Account B

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3:GetObject",  
                "s3:PutObject",  
                "s3:PutObjectAcl"  
            ],  
            "Resource": "arn:aws:s3:::central-s3-bucket/*"  
        }  
    ]  
}
```

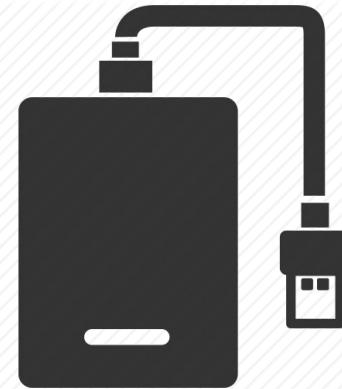


Central S3 Bucket

S3 Encryption

S3 is Back

What's the Need ?

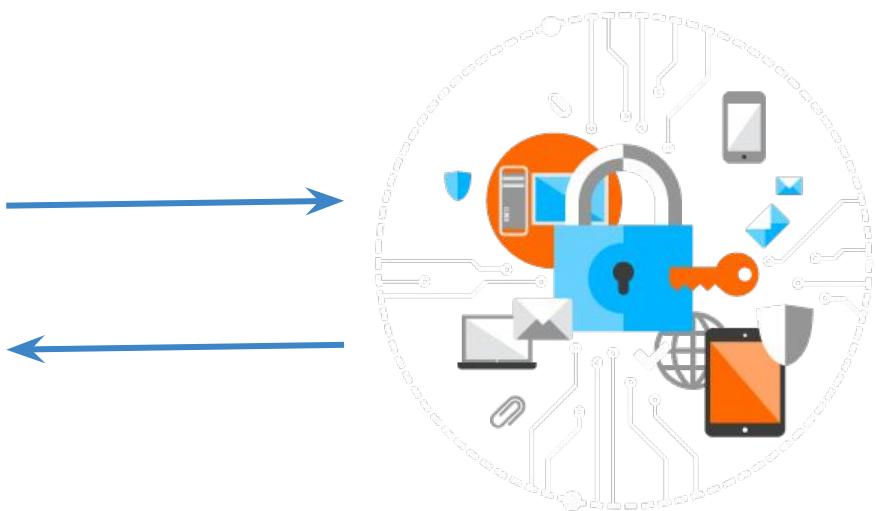


Let's be Proactive

Western Digital external HDs with hardware-based encryption

Aspiring to be a CISSP in 2017? Download the free planning kit!

WD introduced its new **My Book Essential** and **My Book for Mac** desktop external hard drives equipped with the new WD SmartWare software and hardware-based encryption.



S3 also needs Encryption

AWS S3 offers multiple approaches to encrypt the data being stored in S3.

i) Server Side Encryption

- Request Amazon S3 to encrypt your object before saving it on disks in its data centers and then decrypt it when you download the objects.

ii) Client Side Encryption

- Encrypt data client-side and upload the encrypted data to Amazon S3. In this case, you manage the encryption process, the encryption keys, and related tools.

Server Side Encryption

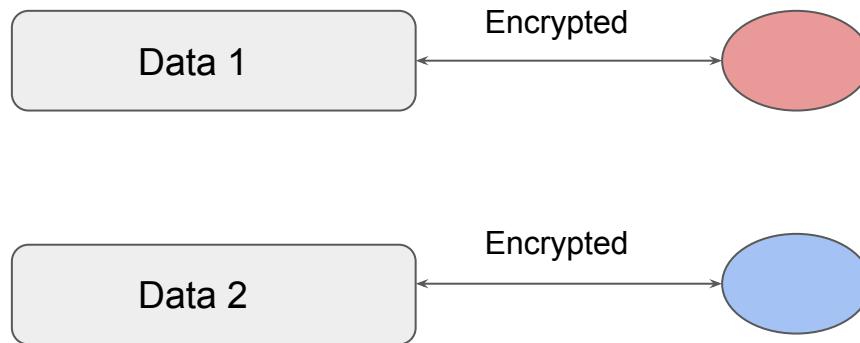
Within Server-Side encryption, there are three options that can be used depending on the use-case.

- Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)
- Server-Side Encryption with Customer Master Keys (CMKs) Stored in AWS Key Management Service (SSE-KMS)
- Server-Side Encryption with Customer-Provided Keys (SSE-C)

SSE with Amazon S3-Managed Keys (SSE-S3)

i) Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)

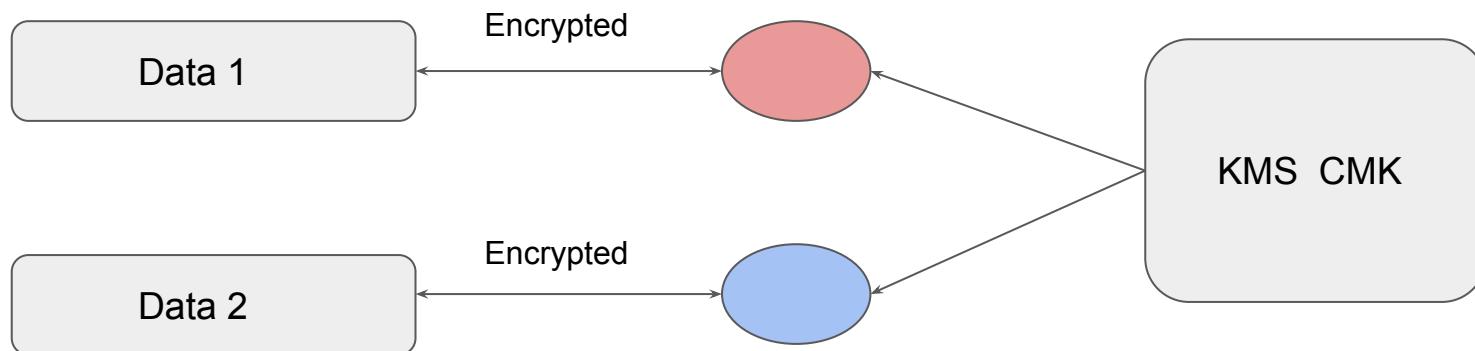
- In this approach, each object is encrypted with a unique key.
- Uses one of the strongest block ciphers to encrypt the data, AES 256.



SSE with CMK (SSE-KMS)

ii) Server-Side Encryption with CMKs Stored in AWS Key Management Service (SSE-KMS)

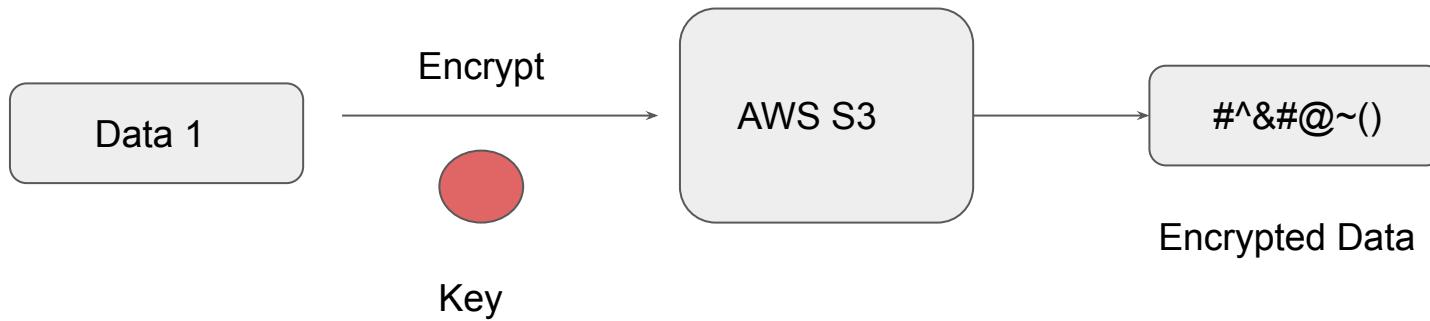
Encrypting data with own CMK allows customers to create, rotate, disable customer managed CMK's. We can also define access controls and enable auditing.



SSE with Customer-Provided Keys (SSE-C)

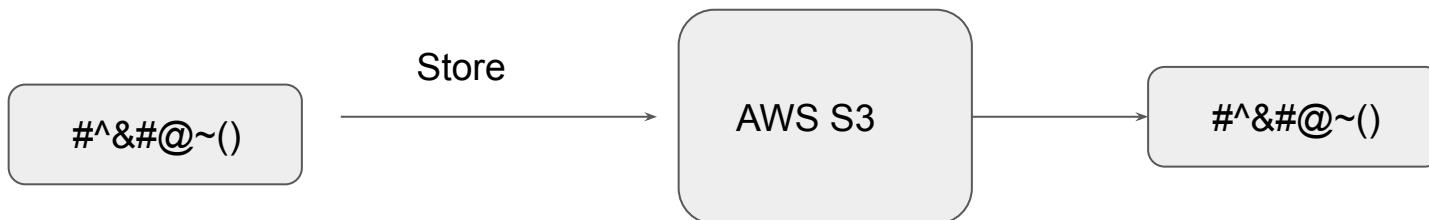
Allows customers to set their own encryption keys.

Encryption key needs to be provided as part of the request and S3 will manage both the encryption as well as the decryption options.



Client Side Encryption

Client-side encryption is the act of encrypting data before sending it to Amazon S3.



Canned ACL

Setting Right Bucket Permissions

Understanding S3 Access ACL

Every bucket and it's objects have an ACL associated with them.

When a request is received, AWS S3 will check against the attached ACL to either allow or block access to that specific object.



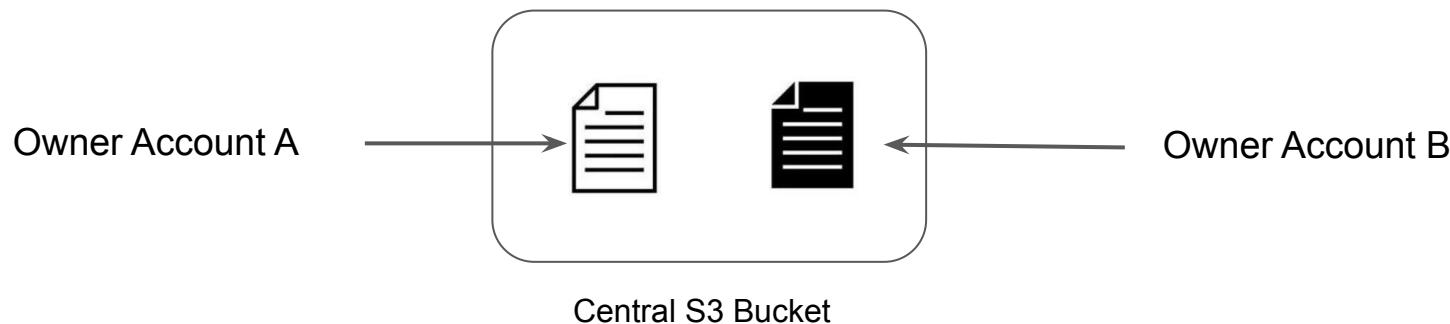
account-a.txt



account-b.txt

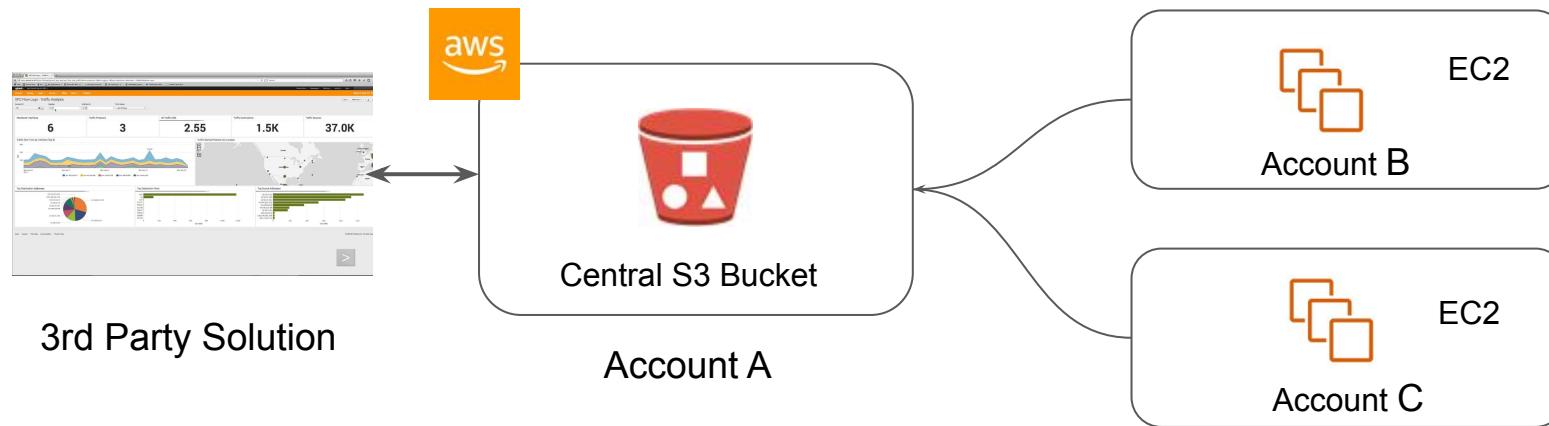
The Tricky Part

When we create a bucket or an object, AWS S3 by default will grant the resource owner full control over the resource.



Ideal Architecture

In most of the architectures, 3rd Party Log Monitoring / SIEM solutions connect to the Central S3 bucket to fetch all of the data.



Canned ACL

AWS S3 supports set of pre-defined grants, known as Canned ACL's.

Each canned ACL has predefined set of permission associated with them.

These canned ACL can be specified in the request using **x-amz-acl** header.

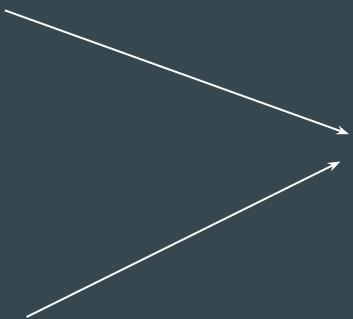
ACL Name	Description
Private	Owner gets FULL_CONTROL. No one else will have access rights (default)
Public-read	Owner has FULL_CONTROL. All others will have public read permission.
Bucket-owner-read	Owner of the object has FULL_CONTROL. Bucket owner will get read permissions.
Bucket-owner-full-control	Both the object owner and the bucket owner get FULL_CONTROL over the object.

S3 - Server Access Logging



Understanding the Basics

S3 Server access logging provides **detailed records for the requests** that are made to a bucket.



```
af410967ff22a9483659f38c3f21bf97449bc2b3ab49be917f5862f1073b439e demo-cloudfront-oai-bucket [17/Jun/2023:07:09:37 +0000] 101.0.62.184  
af410967ff22a9483659f38c3f21bf97449bc2b3ab49be917f5862f1073b439e 6KRW6D008KGMAZT7 REST.GET.VERSIONING - "GET /demo-cloudfront-oai-buckets  
- - "S3Console/0.4, aws-internal/3 aws-sdk-java/1.12.477 Linux/5.4.242-163.349.amzn2int.x86_64 OpenJDK_64-Bit_Server_VM/25.372-b08 java  
cfg/retry-mode/standard" - fK+YLoaPKoGzC0HBnBMhEGqmGgc+r/GR1h2DKpc/8Zb4syjXU+5qX1+QimRE71aFHbtVYwBod0k= SigV4 ECDHE-RSA-AES128-GCM-SHA2  
TLSv1.2 - -
```



Use-Cases for Server Access Logging

You can use server access logs for the following purposes:

- Performing security and access audits
- Learning about your customer base
- Understanding your Amazon S3 bill

Points to Note

CloudTrail **does not deliver logs for requests that fail authentication** (in which the provided credentials are not valid). However, it does include logs for requests in which authorization fails (AccessDenied) and requests that are made by anonymous users.

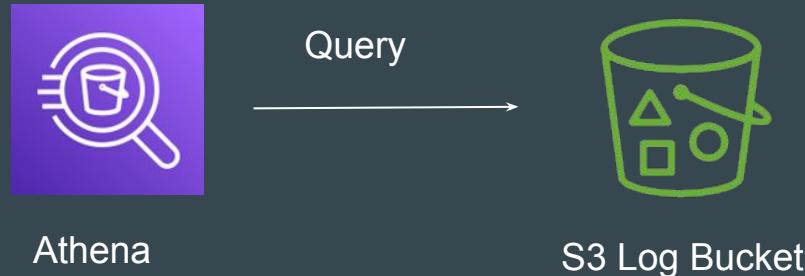
CloudTrail vs S3 Access Logs

Log properties	AWS CloudTrail	Amazon S3 server logs
Can be forwarded to other systems (Amazon CloudWatch Logs, Amazon CloudWatch Events)	Yes	No
Deliver logs to more than one destination (for example, send the same logs to two different buckets)	Yes	No
Turn on logs for a subset of objects (prefix)	Yes	No
Cross-account log delivery (target and source bucket owned by different accounts)	Yes	No
Integrity validation of log file by using digital signature or hashing	Yes	No
Default or choice of encryption for log files	Yes	No
Object operations (by using Amazon S3 APIs)	Yes	Yes
Bucket operations (by using Amazon S3 APIs)	Yes	Yes
Searchable UI for logs	Yes	No
Fields for Object Lock parameters, Amazon S3 Select properties for log records	Yes	No
Fields for Object Size, Total Time, Turn-Around Time, and HTTP Referer for log records		Yes
Lifecycle transitions, expirations, restores		Yes
Logging of keys in a batch delete operation		Yes

Authentication failures ¹		Yes
Accounts where logs get delivered	Bucket owner ² , and requester	Bucket owner only
Performance and Cost	AWS CloudTrail	Amazon S3 Server Logs
Price	Management events (first delivery) are free; data events incur a fee, in addition to storage of logs	No other cost in addition to storage of logs
Speed of log delivery	Data events every 5 minutes; management events every 15 minutes	Within a few hours
Log format	JSON	Log file with space-separated, newline-delimited records

Analyzing S3 Access Logs with Athena

You can easily analyze the S3 Access Logs using Athena to gain detailed insights.



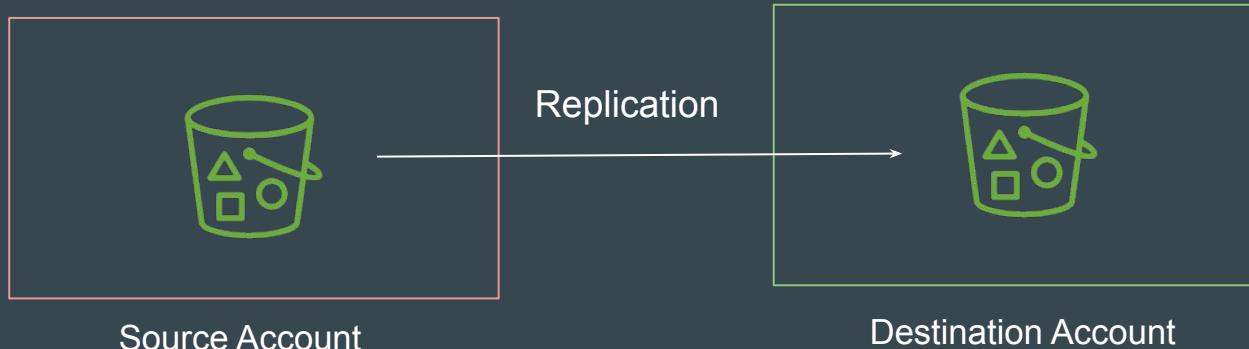
S3 - Cross Account Replication



Understanding the Basics

Replicating Data across different S3 Buckets in same account is a straightforward process.

However for requirements were Source and Destination Bucket are in different account, there are additional configurations that are needed.



End to End WorkFlow Steps

1. IAM Role in the Source Account is required with trust relationship with S3.
2. S3 Bucket Policy in Destination Account to Allow Replicate related operations from Source Account.
3. Setting up Replication Rule with appropriate IAM Role.

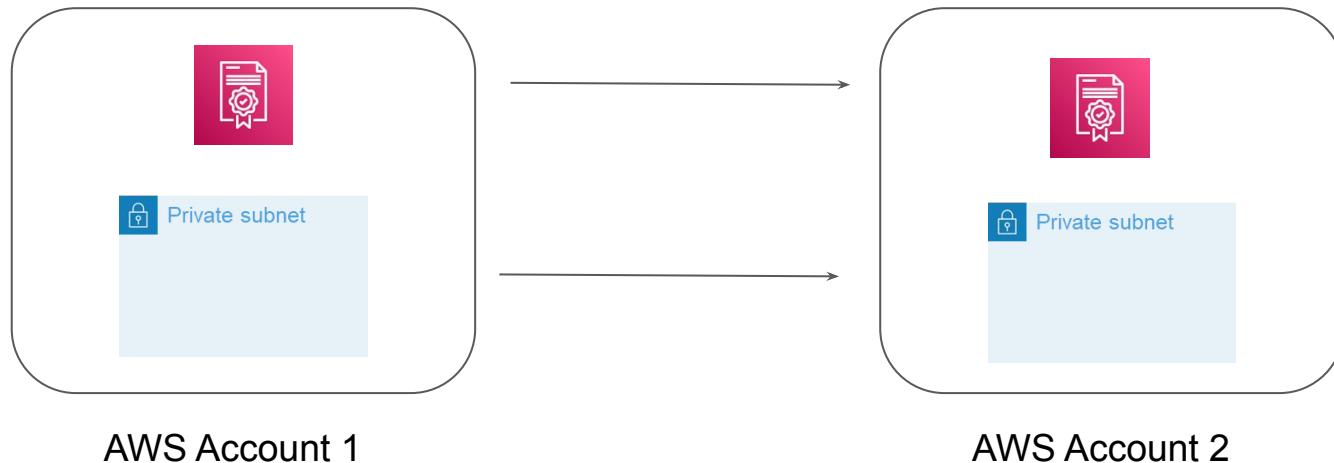


AWS Resource Access Manager

Let's Share Resources

Overview of Resource Access Manager

AWS Resource Access Manager (AWS RAM) helps you securely share the AWS resources that you create in one AWS account with other AWS accounts.



AWS Certificate Manager

Certificates Again :)

Earlier Approach

I have a website and I need to use HTTPS. There are two ways, self-signed certificate and the CA signed certificate.



Self Signed Certificate



CA Signed Certificate

Generating Certificates

To generate a certificate for your domain, you will have to go to a Certificate Authority and after required level of validation, you would be issued a certificate.



User

Generate certificate for kplabs.in



Validated for 1 year.

cert

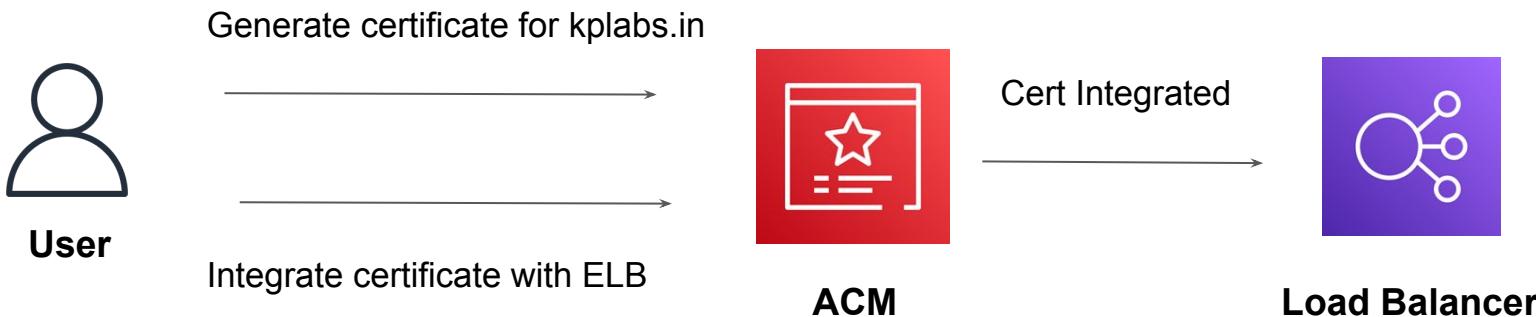
private key



Certificate Authority

AWS Certificate Manager

AWS Certificate Manager (ACM) handles the complexity of creating, storing, and renewing public and private SSL/TLS X.509 certificates and keys that protect your AWS websites and applications.



Virtual Private Network

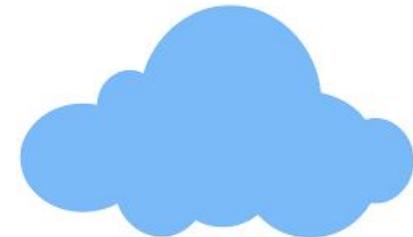
Let's Route

VPN

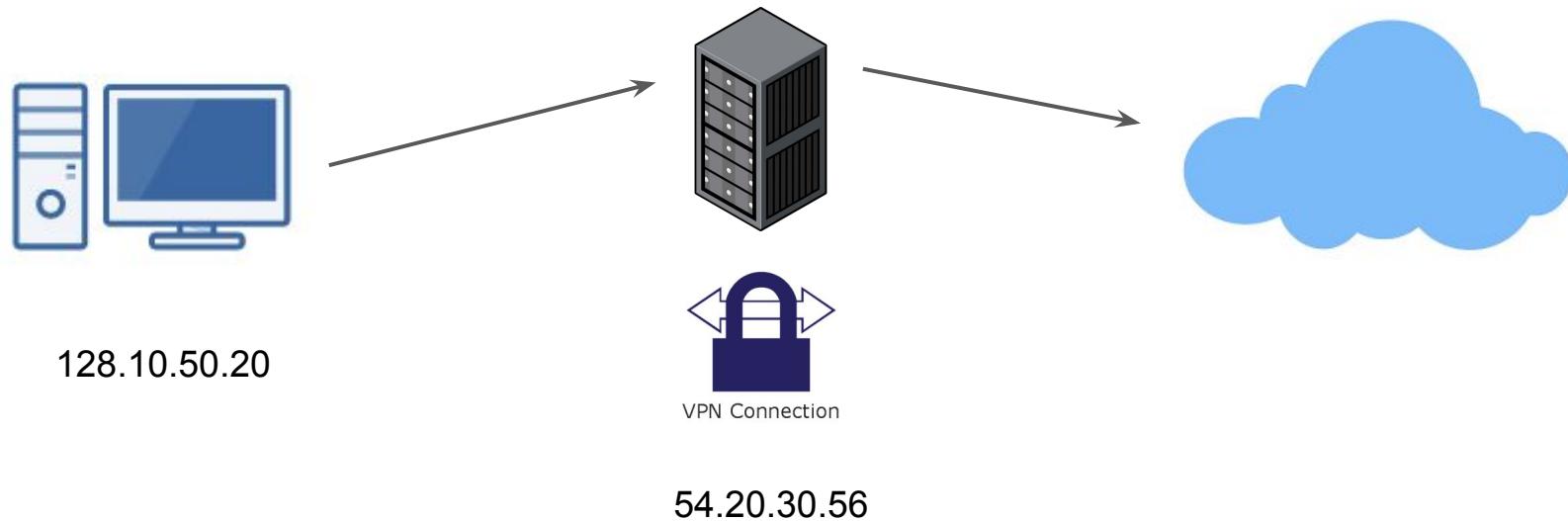
- VPN enables you to route traffic from yourself towards destination through itself.
- Something similar to Proxy.



128.10.50.20

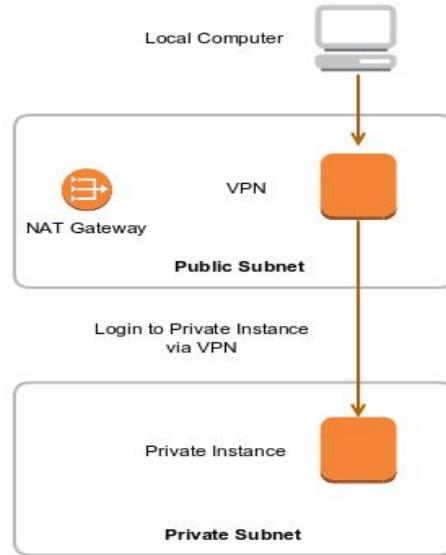


Routing via VPN Server



VPN use in Corporate Network

- In Corporate environments, VPN is used to connect to instances in Private Subnet.
- VPN Server resides in the Public Subnet and you route your traffic via VPN server to instances in Public Subnet.

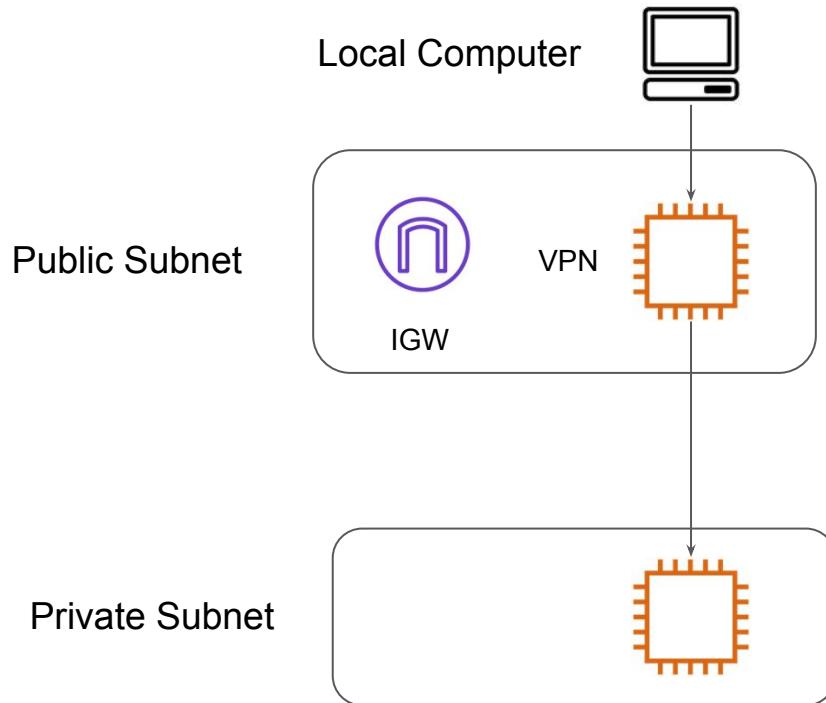


AWS Client VPN

Creating our First VPN in AWS

EC2 Based VPN Architecture

In this approach, you install VPN softwares like OpenVPN in the EC2 instance and use it to route traffic to private subnets.

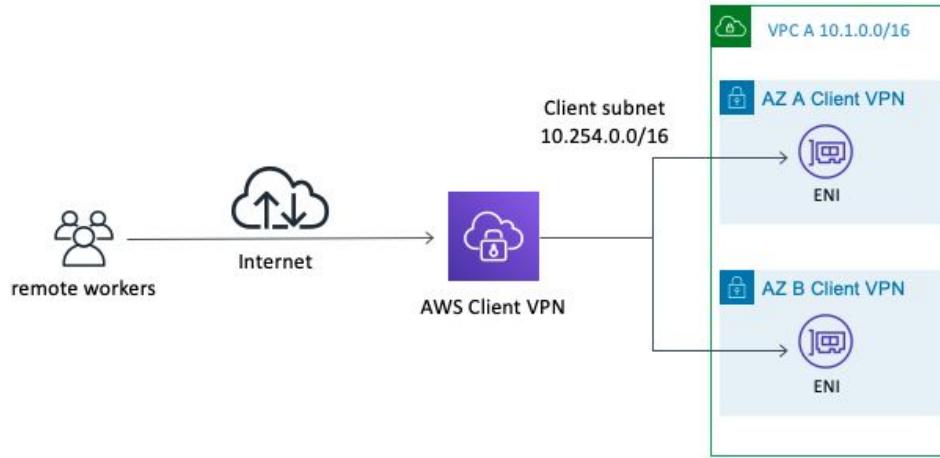


Challenges with EC2 VPN Based Architectures

1. High-Availability (What if VPN EC2 goes down)
2. Patch Management.
3. Upgrade of VPN Software
4. Performance Optimization
5. VPN Server Configuration

AWS Client VPN

AWS Client VPN is a managed client-based VPN service that enables you to securely access your AWS resources and resources in your on-premises network.



EC2 Image Builder



Setting Up the Base

One of my responsibilities was to provide the latest “Hardened AMI” ID to developers from which they can launch their EC2 instances for testing.



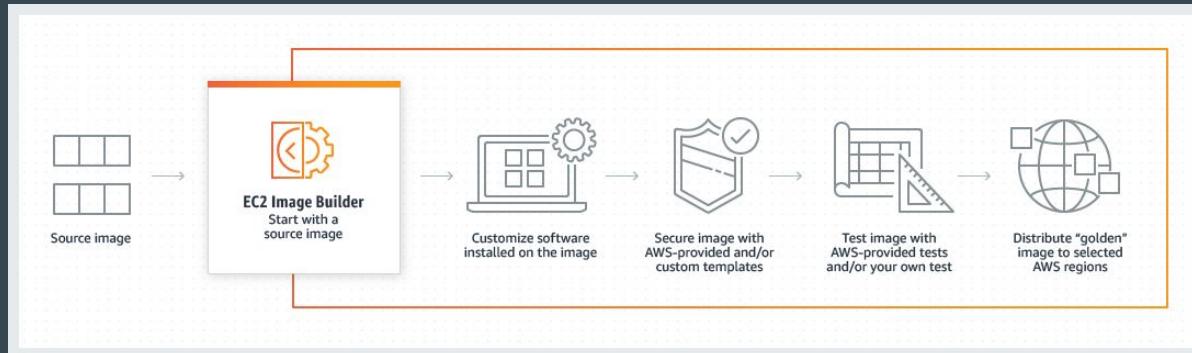
Understanding the Challenge

1. Entire process is manual.
2. What happens if Security Guy is on leave?

EC2 Image Builder

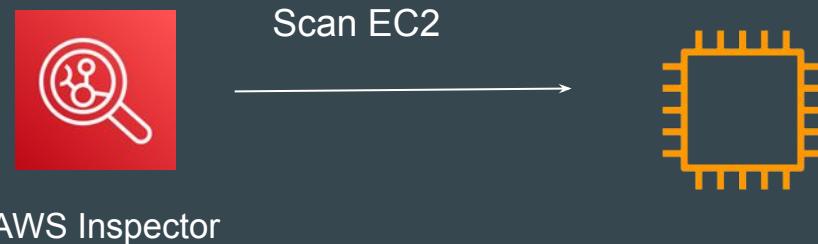
Keeping Virtual Machine and container images up-to-date can be time consuming, resource intensive, and error-prone.

EC2 Image Builder **simplifies** the building, testing, and deployment of Virtual Machine and container images for use on AWS or on-premises.



Benefits - Integration with Other Services

Benefit of EC2 Image builder is that it integrates well with other AWS services like AWS Inspector for vulnerability scanning related use-cases.



Benefits - Readymade Build Component

AWS provides several ready to use build components to install and configure various software and configurations in the base AMI.

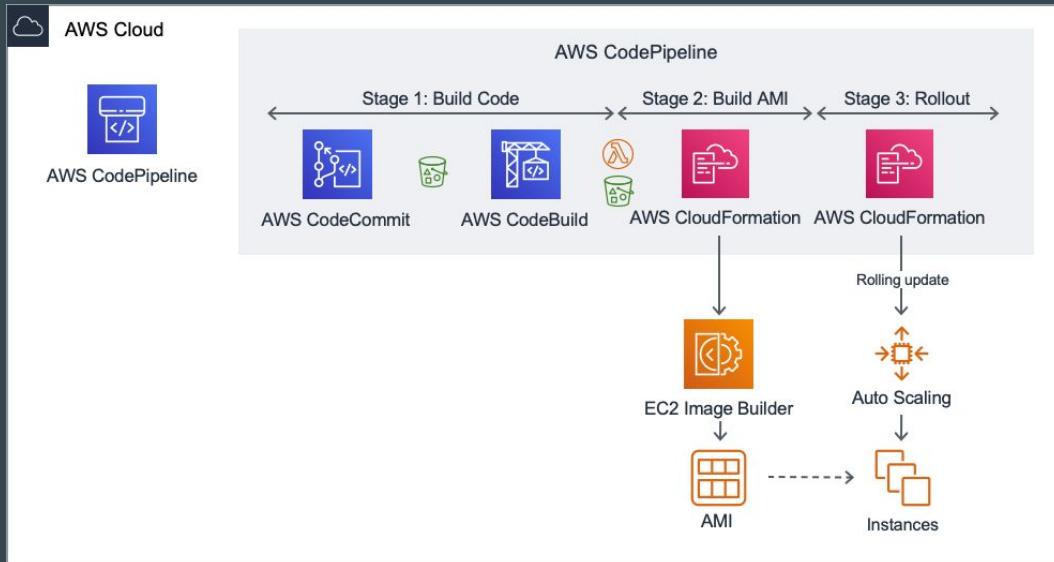
Build components - Amazon Linux (33)			
	Name	Description	
<input type="checkbox"/>	dotnet-core-sdk-linux	Installs Microsoft .NET Core SDK version 3.1 and its dependencies from the Microsoft package repository. Supports AMD64 ONLY. ARM64 is not supported. For more information, see the .NET Core 3.1 download page at https://dotnet.microsoft.com/download/dotnet-core/3.1 . Owner ARN arn:aws:imagebuilder:ap-southeast-1:aws:component/dotnet-core-sdk-linux/x.x.x	Edit
<input type="checkbox"/>	dotnet-runtime-linux	Installs the Microsoft .NET Runtime version 6.0.16. For more information, see the .NET 6.0 download page at https://dotnet.microsoft.com/download/dotnet/6.0 . Owner ARN arn:aws:imagebuilder:ap-southeast-1:aws:component/dotnet-runtime-linux/x.x.x	Edit
<input type="checkbox"/>	dotnet-sdk-linux	Installs the Microsoft .NET SDK version 6.0.408. The installation includes version 6.0.16 of the ASP.NET Core Runtime, the .NET Runtime, and the Desktop Runtime. For more information, see the .NET 6.0 download page at https://dotnet.microsoft.com/download/dotnet/6.0 . Owner ARN arn:aws:imagebuilder:ap-southeast-1:aws:component/dotnet-sdk-linux/x.x.x	Edit
<input type="checkbox"/>	go-linux	Installs Go 1.15.2 for Linux. Owner ARN arn:aws:imagebuilder:ap-southeast-1:aws:component/go-linux/x.x.x	Edit

EC2 Image Builder - Deployment Options



Integrating with Pipeline

You can integrate EC2 Image Builder with your CodePipeline for use-cases related to Immutable infrastructure.



Storing AMI ID In SSM Parameter Store

You can store latest AMI ARNs, IDs in the SSM Parameter Store using combination of SNS and AWS Lambda service..

In CloudFormation stack, you can obtain the recent AMI details from SSM Parameter store to launch EC2 instances accordingly.



SNS and Lambda Integration Workflow

1. When a Lambda function subscribes to an SNS topic, it is invoked with the payload of the published messages.
2. The Lambda function receives the message payload as an input parameter. The Lambda function first checks the message payload to see if the image status is available.
3. If the image state is available, it retrieves the AMI ID from the message payload and updates the SSM parameter.

Best of Luck for the Exams

Positive Possum believes you can do
the thing

