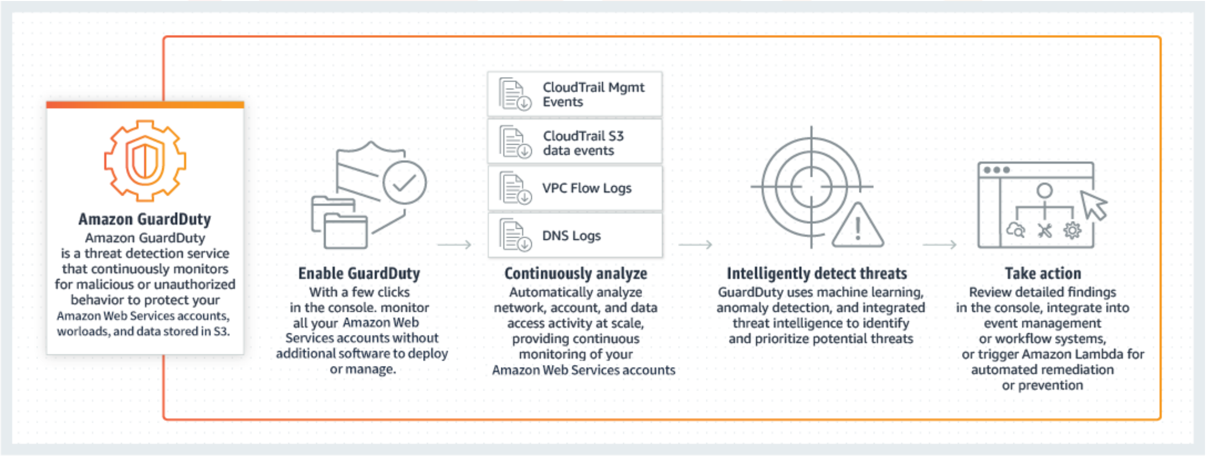
AWS Guard Duty

Protect your AWS accounts with intelligent threat detection

- Achieve organization-wide visibility into possible threats with only a few click
- Expose threats quickly with AWS threat intelligence, behavioral models, and third-party security feeds.
- Mitigate threats early by triggering automated responses.

How it works

Amazon GuardDuty is a threat detection service that continuously monitors your AWS accounts and workloads for malicious activity and delivers detailed security findings for visibility and remediation.



Amazon GuardDuty
Amazon GuardDuty is a threat detection service that continuously monitors for malicious or unauthorized behavior to protect your Amazon Web Services accounts, worloads, and data stored in S3.

Enable GuardDuty
With a few clicks in the console. monitor all your Amazon Web Services accounts without additional software to deploy or manage.

Continuously analyze
Automatically analyze network, account, and data access activity at scale, providing continuous monitoring of your Amazon Web Services accounts

Intelligently detect threats
GuardDuty uses machine learning, anomaly detection, and integrated threat intelligence to identify and prioritize potential threats

Take action
Review detailed findings in the console, integrate into event management or workflow systems, or trigger Amazon Lambda for automated remediation or prevention

Terminology

Account

A standard Amazon Web Services (AWS) account that contains your AWS resources. You can sign in to AWS with your account and enable GuardDuty.

You can also invite other accounts to enable GuardDuty and become associated with your AWS account in GuardDuty. If your invitations are accepted, your account is designated as the **administrator** GuardDuty account, and the added accounts become your **member** accounts. You can then view and manage those accounts' GuardDuty findings on their behalf.

Users of the administrator account can configure GuardDuty as well as view and manage GuardDuty findings for their own account and all of their member accounts. You can have up to 5000 member accounts in GuardDuty.

Users of member accounts can configure GuardDuty as well as view and manage GuardDuty findings in their account (either through the GuardDuty management console or GuardDuty API). Users of member accounts can't view or manage findings in other members' accounts.

An AWS account can't be a GuardDuty administrator and member account at the same time. An AWS account can accept only one membership invitation. Accepting a membership invitation is optional.

Detector

All GuardDuty findings are associated with a detector, which is an object that represents the GuardDuty service. The detector is a regional entity, and a unique detector is required in each region GuardDuty operates in. When you enable GuardDuty in a region a new detector with a unique 32 alphanumeric detector ID is generated in that region. The detector ID format looks like this:

12abc34d567e8fa901bc2d34e56789f0

You can find your detector ID for your current region in the console from the **Settings** pane, or programmatically using the ListDetectors API.

Note

In multiple account environments all findings for member accounts roll up to the administrator account's detector.
Some GuardDuty functionality is configured through the detector, such as configuring CloudWatch Events notification frequency and the enabling or disabling of optional data sources for GuardDuty to process.

Data source

The origin or location of a set of data. To detect unauthorized and unexpected activity in your AWS environment, GuardDuty analyzes and processes data from AWS CloudTrail event logs, VPC Flow Logs, and DNS logs.

Finding

A potential security issue discovered by GuardDuty.
Findings are displayed in the GuardDuty console and contain a detailed description of the security issue. You can also retrieve your generated findings by calling the GetFindings and ListFindings API operations.
You can also see your GuardDuty findings through Amazon CloudWatch events. GuardDuty sends findings to Amazon CloudWatch via HTTPS protocol. For more information.

Suppression rule

Suppression rules allow you to create very specific combinations of attributes to suppress findings. For example, you can define a rule through the GuardDuty filter to auto-archive Recon:EC2/Portscan from only those instances in a specific VPC, running a specific AMI, or with a specific EC2 tag. This rule would result in port scan findings being automatically archived from the instances that meet the criteria. However, it still allows alerting if GuardDuty detects those instances conducting other malicious activity, such as crypto-currency mining.
Suppression rules defined in the GuardDuty administrator account apply to the GuardDuty member accounts. GuardDuty member accounts can't modify suppression rules.

With suppression rules, GuardDuty still generates all findings. Suppression rules provide suppression of findings while maintaining a complete and immutable history of all activity.
Typically suppression rules are used to hide findings that you have determined as false positives for your environment, and reduce the noise from low-value findings so you can focus on larger threats.

Trusted IP list
A list of trusted IP addresses for highly secure communication with your AWS environment. GuardDuty does not generate findings based on trusted IP lists.

Threat list
A list of known malicious IP addresses. GuardDuty generates findings based on threat lists.

GuardDuty Data Sources

- AWS CloudTrail Event Logs
- AWS CloudTrail Management Events
- AWS CloudTrail S3 Data Events
- VPC Flow Logs
- DNS logs

Severity levels for GuardDuty findings

Each GuardDuty finding has an assigned severity level and value that reflects the potential risk the finding could have to your network as determined by our security engineers. The value of the severity can fall anywhere within the 0.1 to 8.9 range, with higher values indicating greater security risk. To help you determine a response to a potential security issue that is highlighted by a finding, GuardDuty breaks down this range into, High, Medium, and Low severity levels.

Note

Values 0 and 9.0 to 10.0 are currently reserved for future use.

The following are the currently defined severity levels and values for the GuardDuty findings as well as general recommendations for each:

Severity level	Value range
High	8.9 - 7.0
A High severity level indicates that the resource in question (an EC2 instance or a set of IAM user credentials) is compromised and is actively being used for unauthorized purposes. We recommend that you treat any High severity finding security issue as a priority and take immediate remediation steps to prevent further unauthorized use of your resources. For example, clean up your EC2 instance or terminate it, or rotate the IAM credentials.	
Medium	6.9 - 4.0
A Medium severity level indicates suspicious activity that deviates from normally observed behavior and, depending on your use case, may be indicative of a resource compromise. We recommend that you investigate the implicated resource at your earliest convenience. Remediation steps will vary by resource and Finding family, but in general, you should be looking to confirm that the activity is authorized and consistent with your use case. If you cannot identify the cause, or confirm the activity	

was authorized, you should consider the resource compromised and follow Remediation Steps to secure the resource. Here are some things to consider when reviewing a Medium level finding: • Check if an authorized user has installed new software that changed the behavior of a resource (for example, allowed higher than normal traffic, or enabled communication on a new port). • Check if an authorized user changed the control panel settings, for example, modified a security group setting. • Run an anti-virus scan on the implicated resource to detect unauthorized software. • Verify the permissions that are attached to the implicated IAM role, user, group, or set of credentials. These might have to be changed or rotated.	
Low	3.9 - 1.0
A low severity level indicates attempted suspicious activity that did not compromise your network, for example, a port scan or a failed intrusion attempt. There is no immediate recommended action, but it is worth making note of this information as it may indicate someone is looking for weak points in your network.	

Findings by resource type

The following pages are broken down by each resource type GuardDuty currently generates findings for. The pages contain detailed information on all finding types for that resources type.

- EC2 finding types
- IAM finding types
- S3 finding types

Findings table

The following table lists all finding types by name, resource, data source and severity. A severity listed with an asterisk (*) indicates the finding has variable severities depending the circumstances of the finding, which are described in the details for that finding. Choose the finding name to open more info about that finding.

FINDING TYPE	RESOURCE	DATA SOURCE	SEVERITY
Backdoor:EC2/C&CActivity.B	EC2	VPC Flow Logs	High
Backdoor:EC2/C&CActivity.B!DNS	EC2	DNS logs	High
Backdoor:EC2/DenialOfService.Dns	EC2	VPC Flow Logs	High
Backdoor:EC2/DenialOfService.Tcp	EC2	VPC Flow Logs	High
Backdoor:EC2/DenialOfService.Udp	EC2	VPC Flow Logs	High
Backdoor:EC2/DenialOfService.UdpOnTcpPorts	EC2	VPC Flow Logs	High
Backdoor:EC2/DenialOfService.UnusualProtocol	EC2	VPC Flow Logs	High
Backdoor:EC2/Spambot	EC2	VPC Flow Logs	Medium
Behavior:EC2/NetworkPortUnusual	EC2	VPC Flow Logs	Medium
Behavior:EC2/TrafficVolumeUnusual	EC2	VPC Flow Logs	Medium
CredentialAccess:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Medium
CryptoCurrency:EC2/BitcoinTool.B	EC2	VPC Flow Logs	High
CryptoCurrency:EC2/BitcoinTool.B!DNS	EC2	DNS logs	High
DefenseEvasion:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Medium
Discovery:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Low
Discovery:S3/MaliciousIPCaller	S3	CloudTrail S3 data event	High

Discovery:S3/MaliciousIPCaller.Custom	S3	CloudTrail S3 data event	High
Discovery:S3/TorIPCaller	S3	CloudTrail S3 data event	Medium
Exfiltration:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	High
Exfiltration:S3/MaliciousIPCaller	S3	CloudTrail S3 data event	High
Exfiltration:S3/ObjectRead.Unusual	S3	S3 CloudTrail data event	Medium*
Impact:EC2/AbusedDomainRequest.Reputation	EC2	DNS logs	Medium
Impact:EC2/BitcoinDomainRequest.Reputation	EC2	DNS logs	High
Impact:EC2/MaliciousDomainRequest.Reputation	EC2	DNS logs	High
Impact:EC2/PortSweep	EC2	VPC Flow Logs	High
Impact:EC2/SuspiciousDomainRequest.Reputation	EC2	DNS logs	Low
Impact:EC2/WinRMBruteForce	EC2	VPC Flow Logs	Low*
Impact:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	High
Impact:S3/MaliciousIPCaller	S3	CloudTrail S3 data event	High
InitialAccess:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Medium
PenTest:IAMUser/KaliLinux	IAM	CloudTrail management event	Medium
PenTest:IAMUser/ParrotLinux	IAM	CloudTrail management event	Medium
PenTest:IAMUser/PentooLinux	IAM	CloudTrail management event	Medium

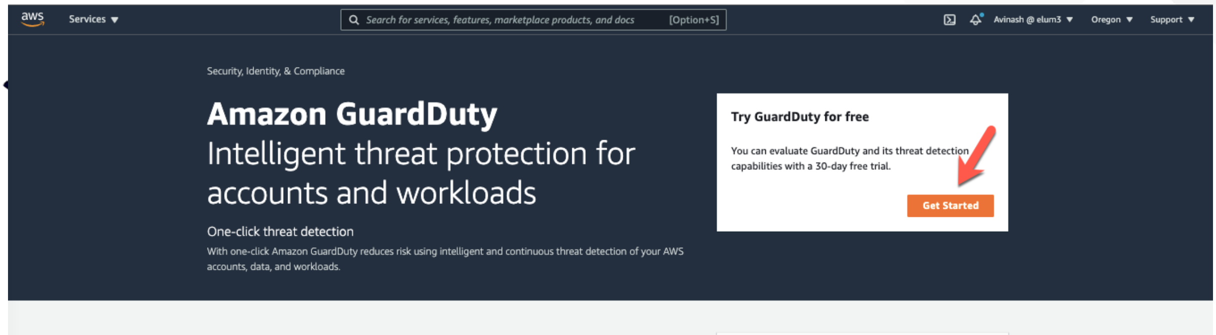
PenTest:S3/KaliLinux	S3	CloudTrail S3 data event	Medium
PenTest:S3/ParrotLinux	S3	CloudTrail S3 data event	Medium
PenTest:S3/PentooLinux	S3	CloudTrail S3 data event	Medium
Persistence:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Medium
Policy:IAMUser/RootCredentialUsage	IAM	CloudTrail management event or CloudTrail data event	Low
Policy:S3/AccountBlockPublicAccessDisabled	S3	CloudTrail management event	Low
Policy:S3/BucketAnonymousAccessGranted	S3	CloudTrail management event	High
Policy:S3/BucketBlockPublicAccessDisabled	S3	CloudTrail management event	Low
Policy:S3/BucketPublicAccessGranted	S3	CloudTrail management event	High
PrivilegeEscalation:IAMUser/AnomalousBehavior	IAM	CloudTrail management event	Medium
Recon:EC2/PortProbeEMRUnprotectedPort	EC2	VPC Flow Logs	High
Recon:EC2/PortProbeUnprotectedPort	EC2	VPC Flow Logs	Low*
Recon:EC2/Portscan	EC2	VPC Flow Logs	Medium
Recon:IAMUser/MaliciousIPCaller	IAM	CloudTrail management event	Medium
Recon:IAMUser/MaliciousIPCaller.Custom	IAM	CloudTrail management event	Medium

Recon:IAMUser/TorIPCaller	IAM	CloudTrail management event	Medium
Stealth:IAMUser/CloudTrailLoggingDisabled	IAM	CloudTrail management event	Low
Stealth:IAMUser/PasswordPolicyChange	IAM	CloudTrail management event	Low
Stealth:S3/ServerAccessLoggingDisabled	S3	CloudTrail management event	Low
Trojan:EC2/BlackholeTraffic	EC2	VPC Flow Logs	Medium
Trojan:EC2/BlackholeTraffic!DNS	EC2	DNS logs	Medium
Trojan:EC2/DGADomainRequest.B	EC2	DNS logs	High
Trojan:EC2/DGADomainRequest.C!DNS	EC2	DNS logs	High
Trojan:EC2/DNSDataExfiltration	EC2	DNS logs	High
Trojan:EC2/DriveBySourceTraffic!DNS	EC2	DNS logs	High
Trojan:EC2/DropPoint	EC2	VPC Flow Logs	Medium
Trojan:EC2/DropPoint!DNS	EC2	DNS logs	Medium
Trojan:EC2/PhishingDomainRequest!DNS	EC2	DNS logs	High
UnauthorizedAccess:EC2/MaliciousIPCaller.Custom	EC2	VPC Flow Logs	Medium
UnauthorizedAccess:EC2/MetadataDNSRebind	EC2	DNS logs	High
UnauthorizedAccess:EC2/RDPBruteForce	EC2	VPC Flow Logs	Low*
UnauthorizedAccess:EC2/SSHBruteForce	EC2	VPC Flow Logs	Low*
UnauthorizedAccess:EC2/TorClient	EC2	VPC Flow Logs	High
UnauthorizedAccess:EC2/TorRelay	EC2	VPC Flow Logs	High
UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B	IAM	CloudTrail management event	Medium

UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS	IAM	CloudTrail management event	High
UnauthorizedAccess:IAMUser/MaliciousIPCaller	IAM	CloudTrail management event	Medium
UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom	IAM	CloudTrail management event	Medium
UnauthorizedAccess:IAMUser/TorIPCaller	IAM	CloudTrail management event	Medium
UnauthorizedAccess:S3/MaliciousIPCaller.Custom	S3	CloudTrail S3 data event	High
UnauthorizedAccess:S3/TorIPCaller	S3	CloudTrail S3 data event	

Enable GuardDuty

- Navigate to Amazon GuardDuty console



aws
Services ▼
Search for services, features, marketplace products, and docs [Option+S]
Avinash @ elum3 ▼
Oregon ▼
Support ▼

Security, Identity, & Compliance

Amazon GuardDuty

Intelligent threat protection for accounts and workloads

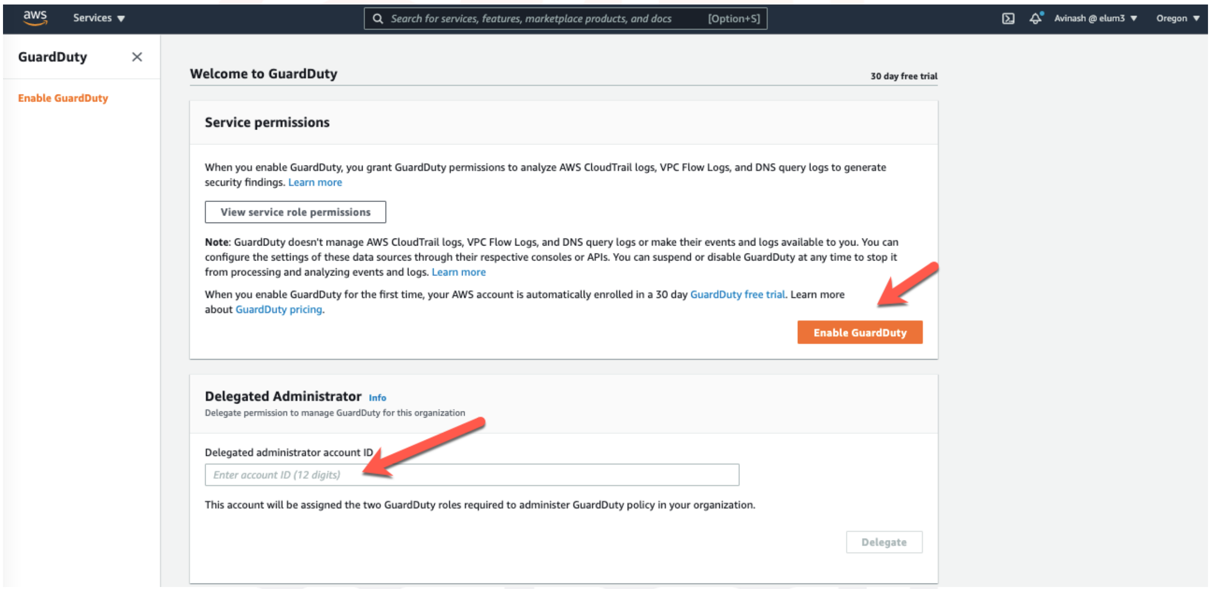
One-click threat detection

With one-click Amazon GuardDuty reduces risk using intelligent and continuous threat detection of your AWS accounts, data, and workloads.

Try GuardDuty for free

You can evaluate GuardDuty and its threat detection capabilities with a 30-day free trial.

- If you are trying to enable GuardDuty in child accounts you can delegate this to Admin account
- If you are doing this in an individual account , you are not required to delegate access



aws
Services ▼
Search for services, features, marketplace products, and docs [Option+S]
Avinash @ elum3 ▼
Oregon ▼
GuardDuty
Enable GuardDuty

Welcome to GuardDuty 30 day free trial

Service permissions

When you enable GuardDuty, you grant GuardDuty permissions to analyze AWS CloudTrail logs, VPC Flow Logs, and DNS query logs to generate security findings. Learn more

View service role permissions

Note: GuardDuty doesn't manage AWS CloudTrail logs, VPC Flow Logs, and DNS query logs or make their events and logs available to you. You can configure the settings of these data sources through their respective consoles or APIs. You can suspend or disable GuardDuty at any time to stop it from processing and analyzing events and logs. Learn more

When you enable GuardDuty for the first time, your AWS account is automatically enrolled in a 30 day GuardDuty free trial. Learn more about GuardDuty pricing.

Delegated Administrator Info

Delegate permission to manage GuardDuty for this organization

Delegated administrator account ID

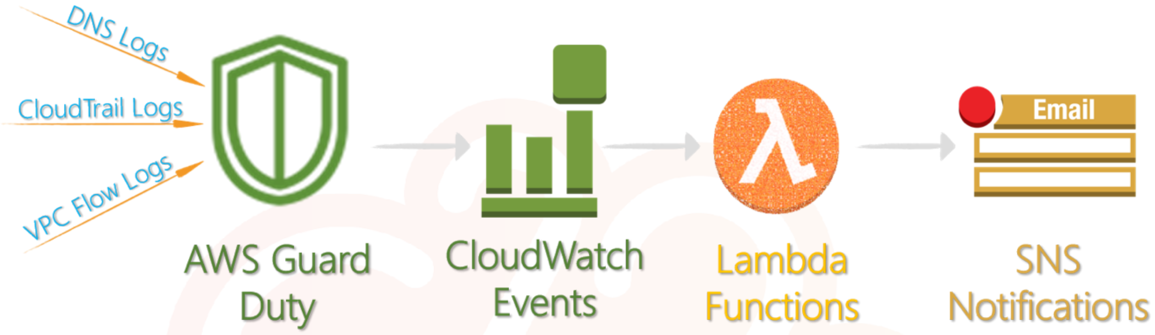
Enter account ID (12 digits)

This account will be assigned the two GuardDuty roles required to administer GuardDuty policy in your organization.

Demo:

Amazon GuardDuty Findings to SNS

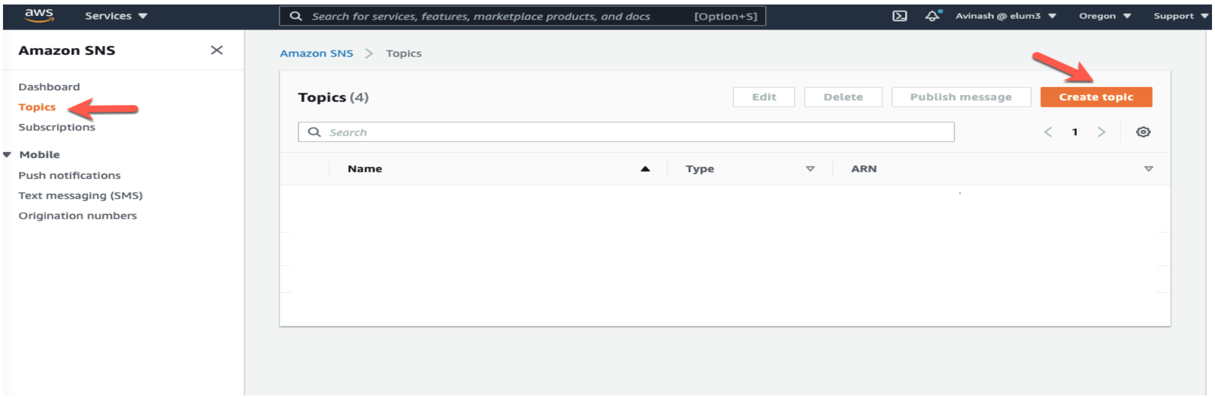
Every GuardDuty finding is assigned a finding ID. For every finding with a unique finding ID, GuardDuty aggregates all subsequent occurrences of a particular finding that take place in six-hour intervals into a single event. GuardDuty then sends a notification about these subsequent occurrences based on this event. We can use this to push the notifications into SNS topic, and getting the security teams to investigate the findings.

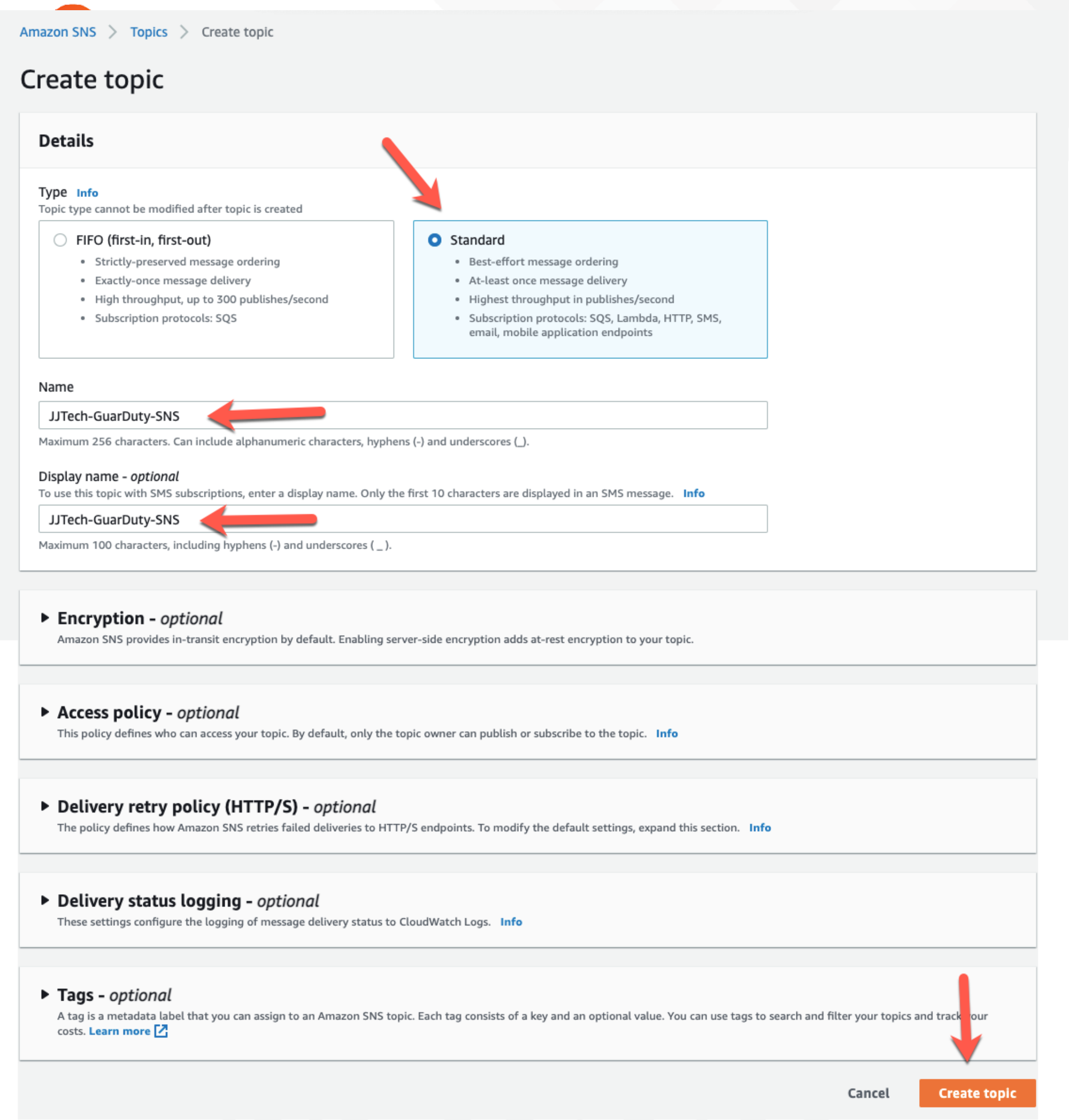


This AWS Lambda function will help you to automatically push GuardDuty findings to an SNS topic which can be used by ITSM tools for their workflows.

Step-1

- Create a SNS Topic for Lambda to publish the GuardDuty Findings.
- Navigate to AWS SNS Console





Create topic

Details

Type Info

Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)
- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS

● Standard
- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

Name

JJTech-GuarDuty-SNS

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Display name - *optional*

To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message. Info

JJTech-GuarDuty-SNS

Maximum 100 characters, including hyphens (-) and underscores (_).

▶ **Encryption** - *optional*

Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

▶ **Access policy** - *optional*

This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic. Info

▶ **Delivery retry policy (HTTP/S)** - *optional*

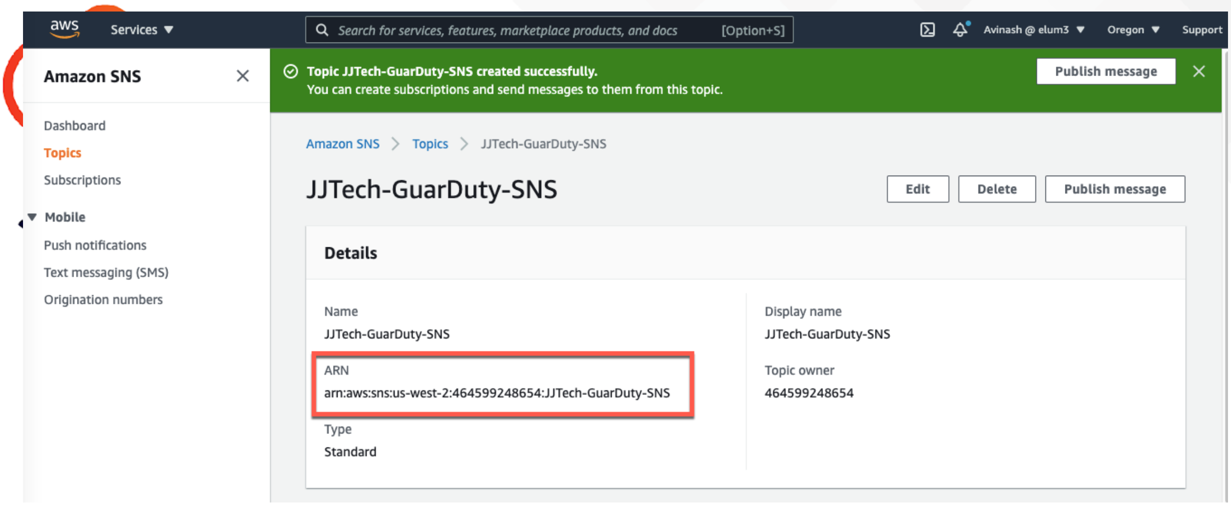
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section. Info

▶ **Delivery status logging** - *optional*

These settings configure the logging of message delivery status to CloudWatch Logs. Info

▶ **Tags** - *optional*

A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. Learn more



Topic JJTech-GuarDuty-SNS created successfully.
You can create subscriptions and send messages to them from this topic.

JJTech-GuarDuty-SNS

Details

Name
JJTech-GuarDuty-SNS

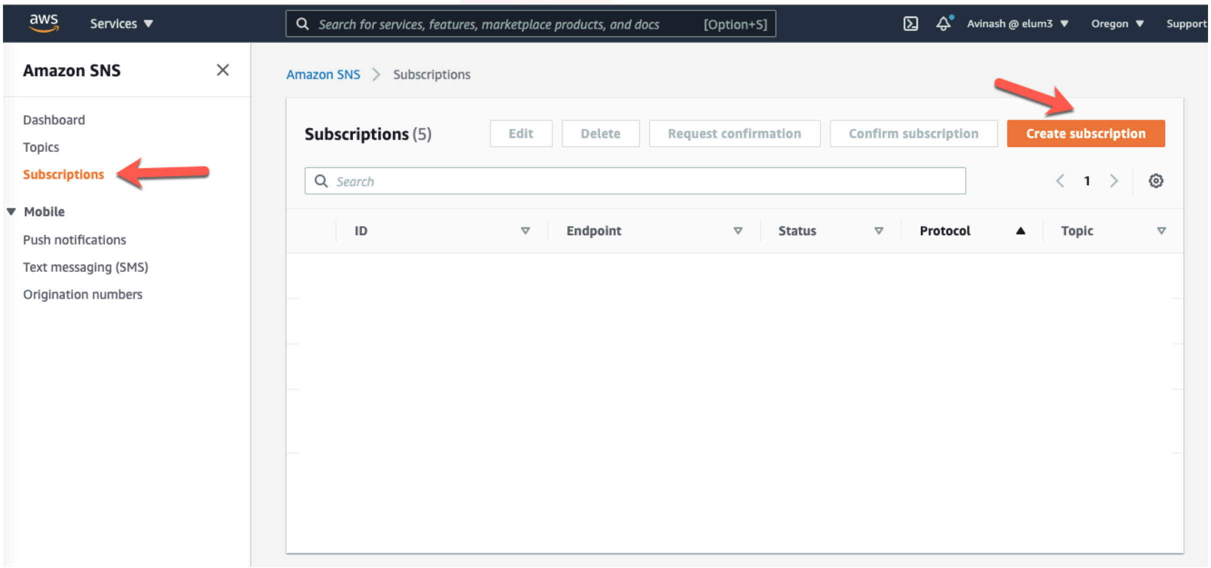
ARN
arn:aws:sns:us-west-2:464599248654:JJTech-GuarDuty-SNS

Type
Standard

Display name
JJTech-GuarDuty-SNS

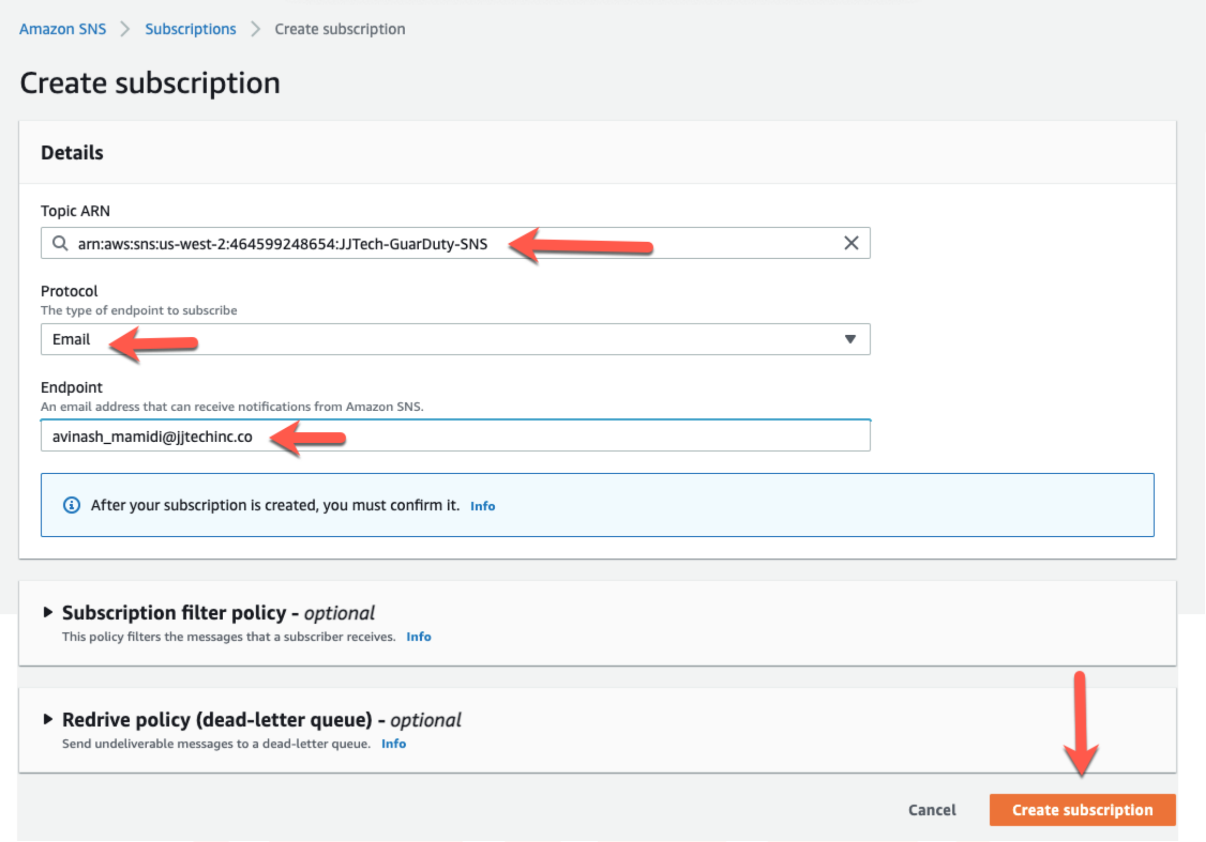
Topic owner
464599248654

- Now you can subscribe the SNS topic with your mail id.



Subscriptions (5)

ID	Endpoint	Status	Protocol	Topic



Amazon SNS > Subscriptions > Create subscription

Create subscription

Details

Topic ARN: arn:aws:sns:us-west-2:464599248654:JJTech-GuardDuty-SNS

Protocol
The type of endpoint to subscribe
Email

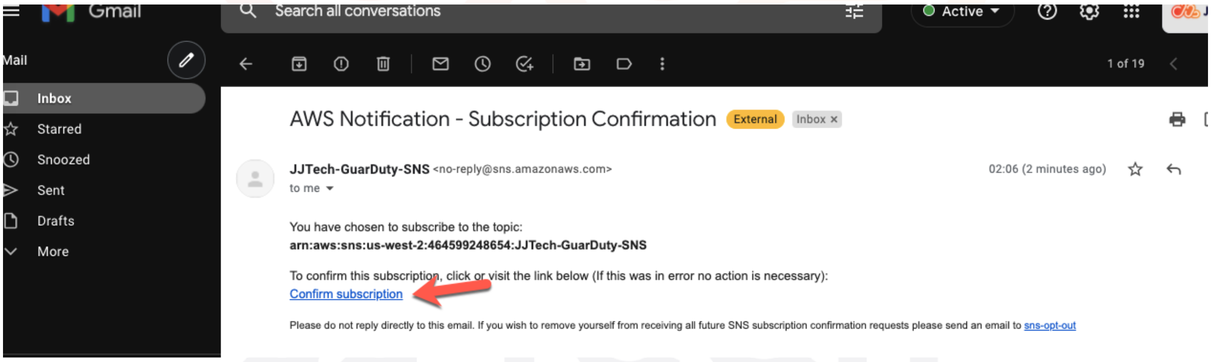
Endpoint
An email address that can receive notifications from Amazon SNS.
avinash_mamidi@jjtechinc.co

After your subscription is created, you must confirm it. Info

Subscription filter policy - *optional*
This policy filters the messages that a subscriber receives. Info

Redrive policy (dead-letter queue) - *optional*
Send undeliverable messages to a dead-letter queue. Info

- You will receive a mail to confirm the subscription to your mail



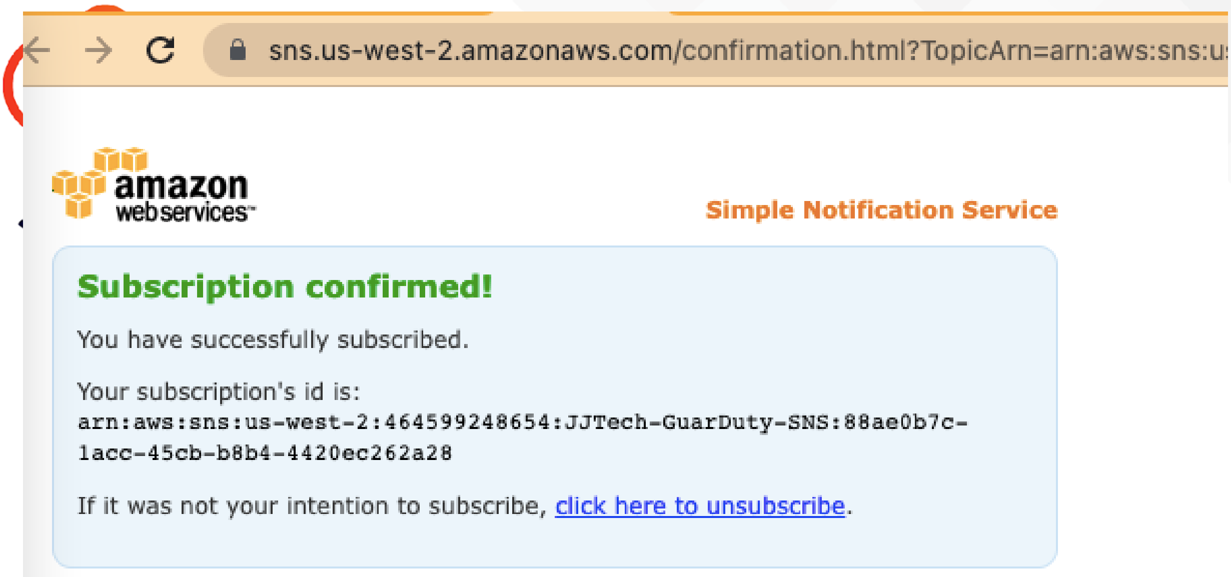
AWS Notification - Subscription Confirmation

JJTech-GuardDuty-SNS <no-reply@sns.amazonaws.com>
to me
02:06 (2 minutes ago)

You have chosen to subscribe to the topic:
arn:aws:sns:us-west-2:464599248654:JJTech-GuardDuty-SNS

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out



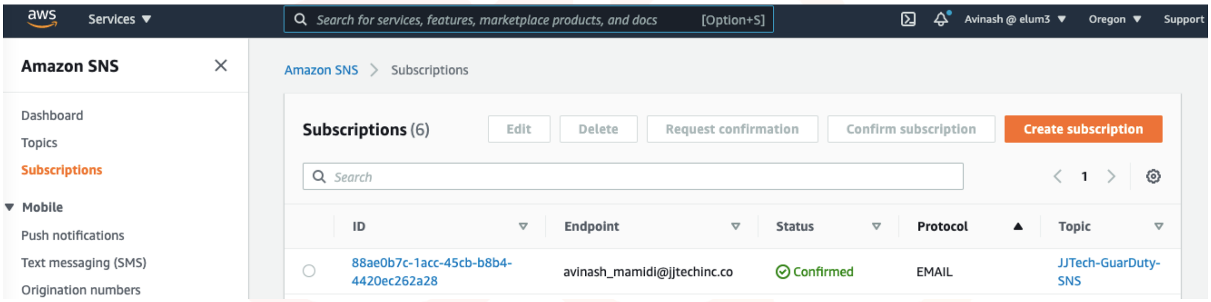
sns.us-west-2.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:u

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:
`arn:aws:sns:us-west-2:464599248654:JJTech-GuarDuty-SNS:88ae0b7c-1acc-45cb-b8b4-4420ec262a28`

If it was not your intention to subscribe, click here to unsubscribe.

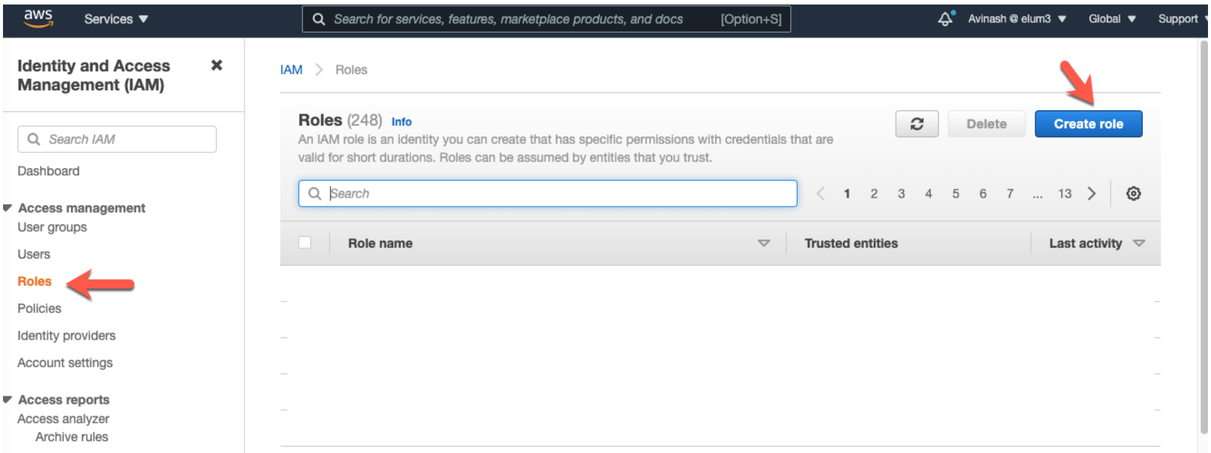


Subscriptions (6)

	ID	Endpoint	Status	Protocol	Topic
○	88ae0b7c-1acc-45cb-b8b4-4420ec262a28	avinash_mamidi@jjtechinc.co	✓ Confirmed	EMAIL	JJTech-GuarDuty-SNS

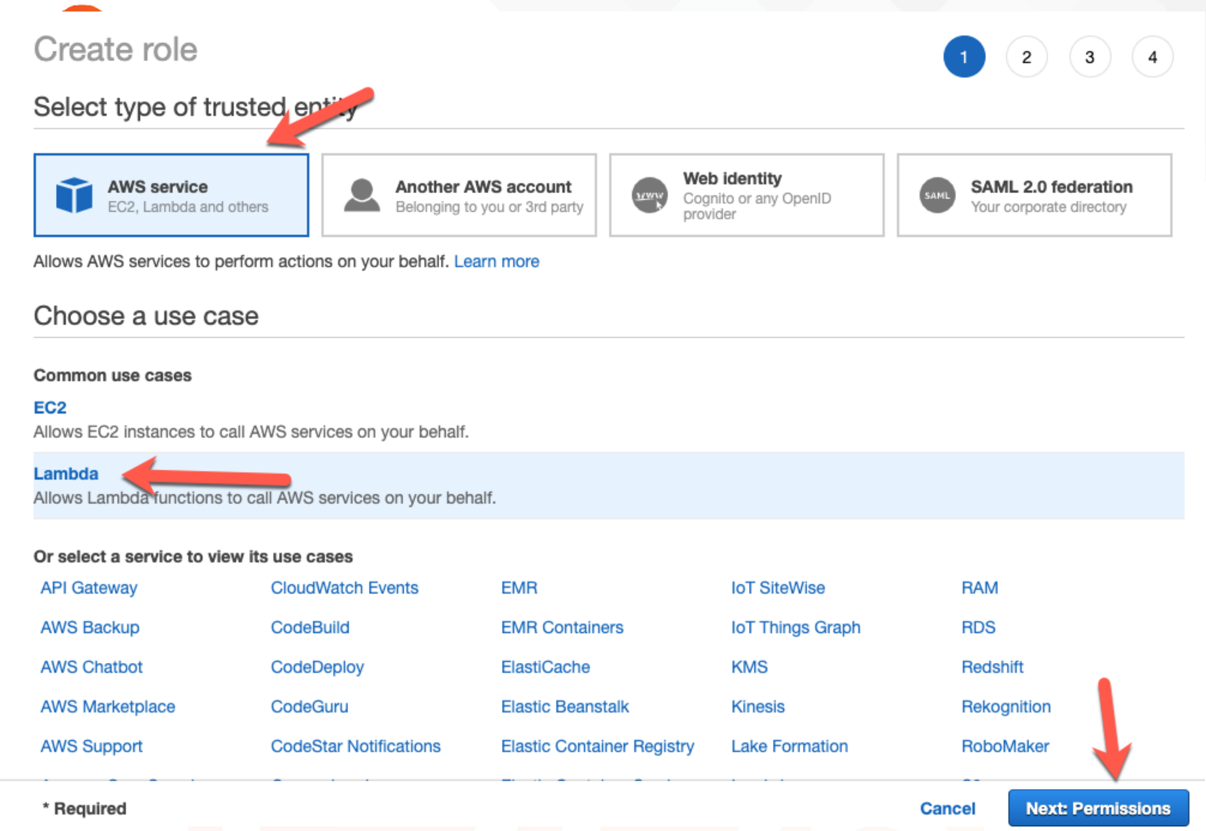
Step-2 : Create a IAM role for the Lambda function

- Navigate to IAM Console



Roles (248) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.



Select type of trusted entity

Allows AWS services to perform actions on your behalf. Learn more

Choose a use case

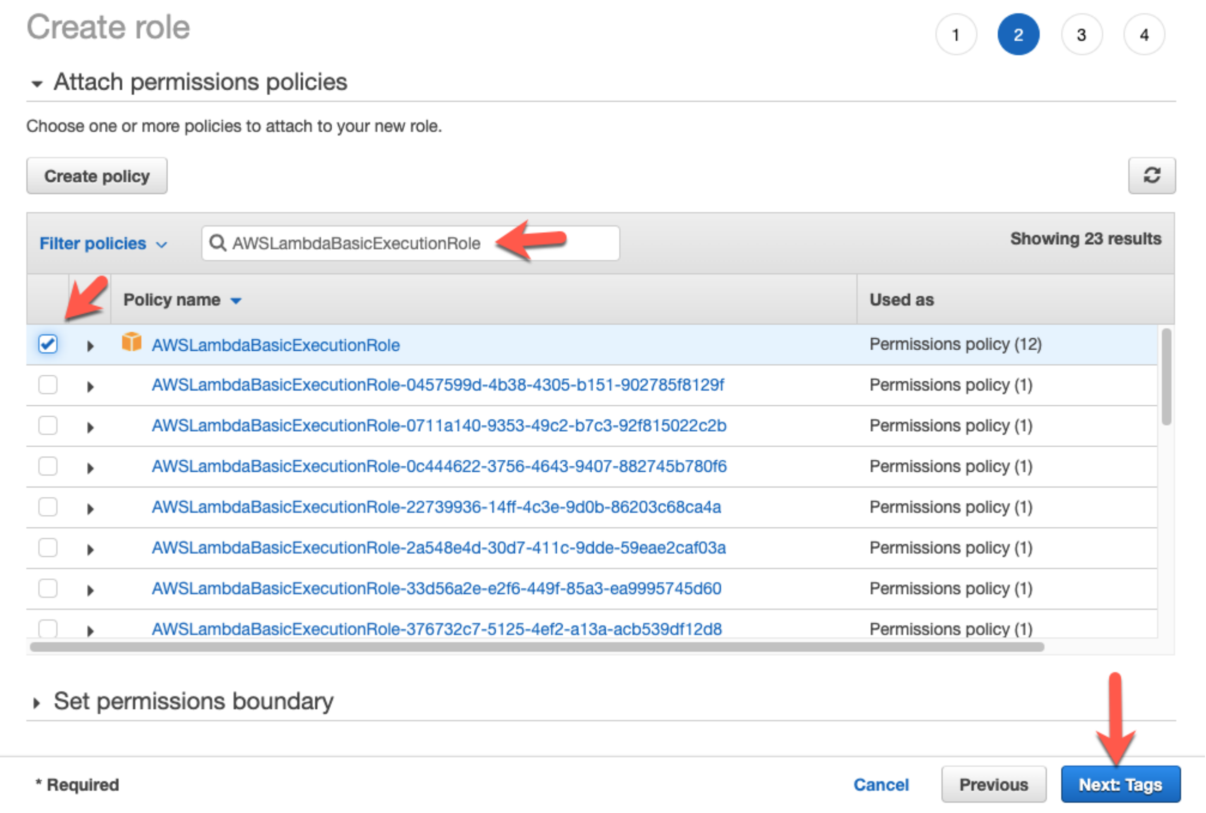
Common use cases

EC2
Allows EC2 instances to call AWS services on your behalf.

Lambda
Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

API Gateway	CloudWatch Events	EMR	IoT SiteWise	RAM
AWS Backup	CodeBuild	EMR Containers	IoT Things Graph	RDS
AWS Chatbot	CodeDeploy	ElastiCache	KMS	Redshift
AWS Marketplace	CodeGuru	Elastic Beanstalk	Kinesis	Rekognition
AWS Support	CodeStar Notifications	Elastic Container Registry	Lake Formation	RoboMaker



Attach permissions policies

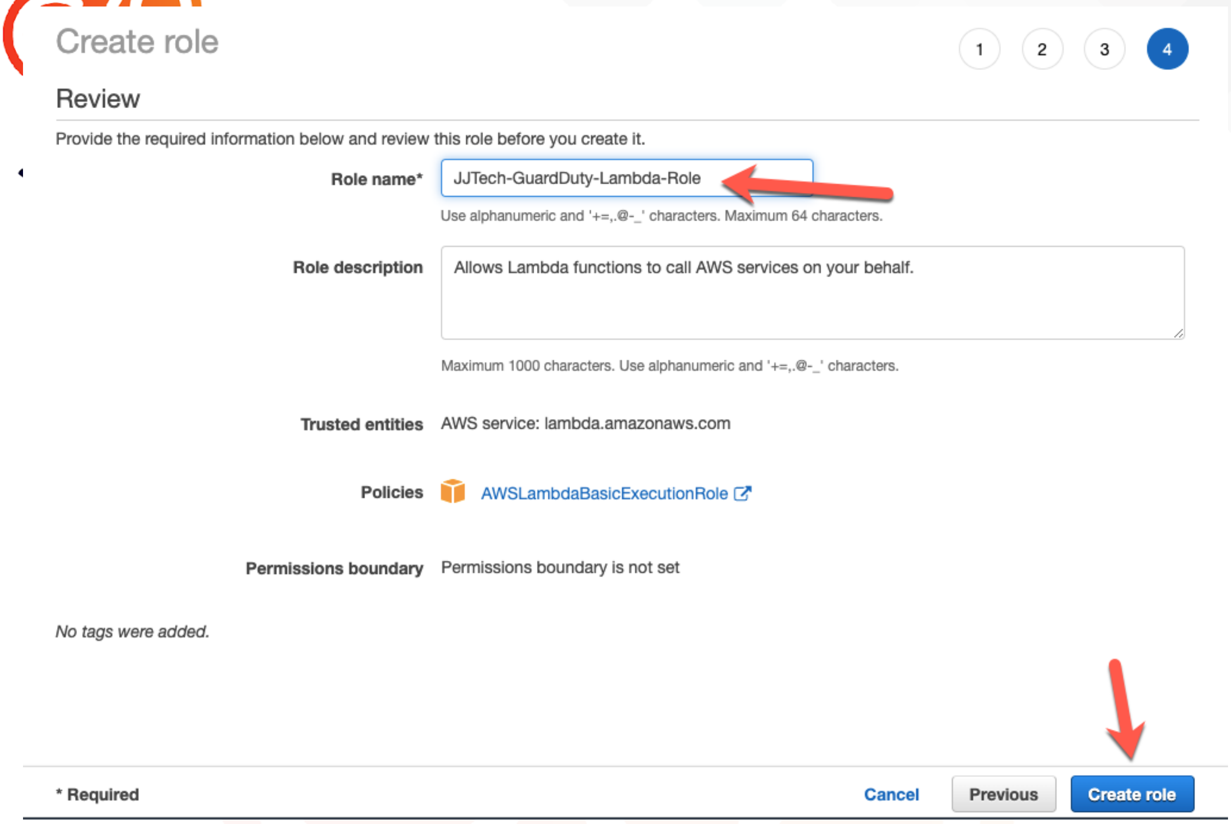
Choose one or more policies to attach to your new role.

AWSLambdaBasicExecutionRole

Showing 23 results

	Policy name	Used as
☑	AWSLambdaBasicExecutionRole	Permissions policy (12)
□	AWSLambdaBasicExecutionRole-0457599d-4b38-4305-b151-902785f8129f	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-0711a140-9353-49c2-b7c3-92f815022c2b	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-0c444622-3756-4643-9407-882745b780f6	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-22739936-14ff-4c3e-9d0b-86203c68ca4a	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-2a548e4d-30d7-411c-9dde-59eae2caf03a	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-33d56a2e-e2f6-449f-85a3-ea9995745d60	Permissions policy (1)
□	AWSLambdaBasicExecutionRole-376732c7-5125-4ef2-a13a-acb539df12d8	Permissions policy (1)

Set permissions boundary



Create role

Review

Provide the required information below and review this role before you create it.

Role name* JJTech-GuardDuty-Lambda-Role

Use alphanumeric and '+=,.@-_' characters. Maximum 64 characters.

Role description Allows Lambda functions to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

Trusted entities AWS service: lambda.amazonaws.com

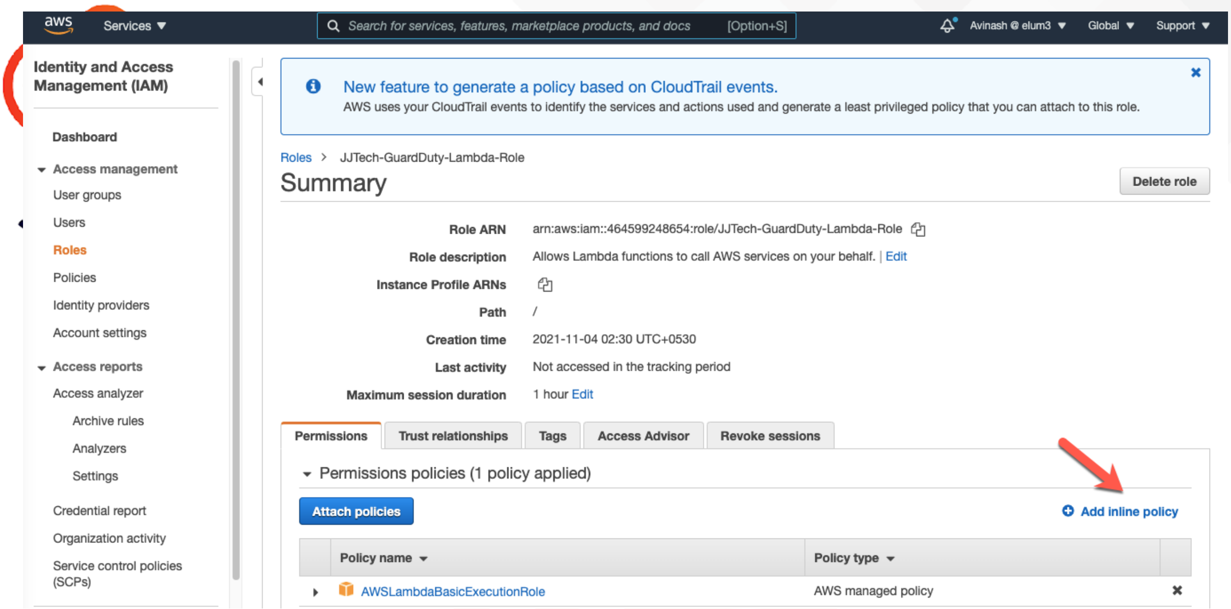
Policies AWSLambdaBasicExecutionRole

Permissions boundary Permissions boundary is not set

No tags were added.

* Required

- Navigate to the role you created



aws
Services ▼
Search for services, features, marketplace products, and docs [Option+S]
Avinash @ elum3 ▼
Global ▼
Support ▼

Identity and Access Management (IAM)

New feature to generate a policy based on CloudTrail events.
AWS uses your CloudTrail events to identify the services and actions used and generate a least privileged policy that you can attach to this role.

Roles > JJTech-GuardDuty-Lambda-Role

Summary

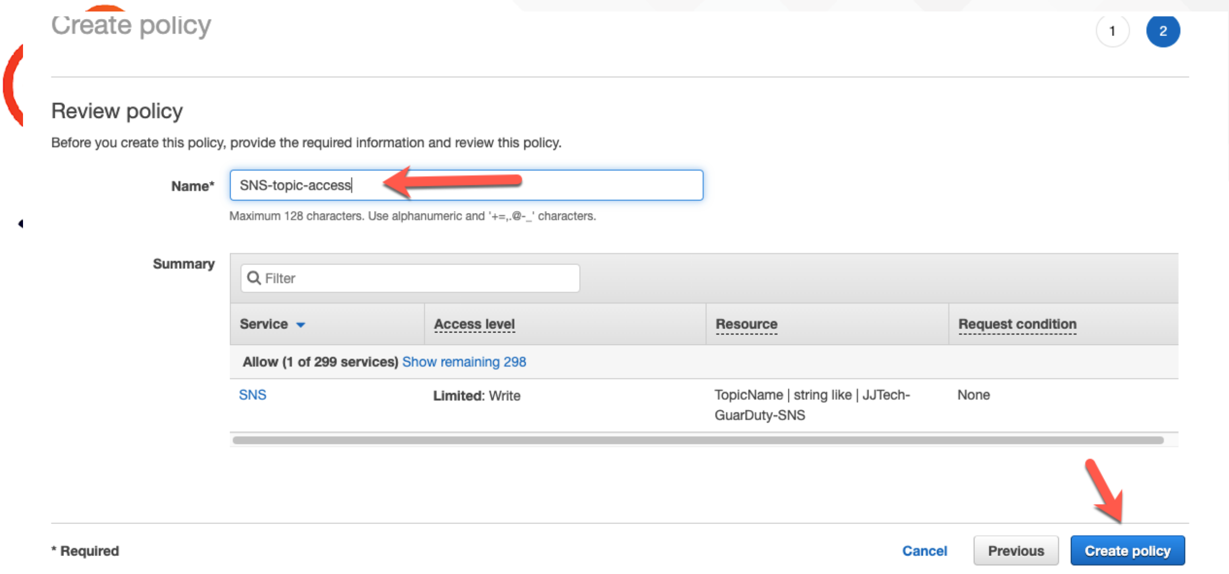
Role ARN	arn:aws:iam::464599248654:role/JJTech-GuardDuty-Lambda-Role
Role description	Allows Lambda functions to call AWS services on your behalf. | Edit
Instance Profile ARNs	
Path	/
Creation time	2021-11-04 02:30 UTC+0530
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

Permissions policies (1 policy applied)

Policy name	Policy type
AWSLambdaBasicExecutionRole	AWS managed policy

- Update the SNS Topic ARN which was created in Step -

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sns:Publish",
      "Resource": "<ARN-OF-YOUR-SNS-TOPIC>"
    }
  ]
}
```



Create policy

Review policy

Before you create this policy, provide the required information and review this policy.

Name* SNS-topic-access

Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

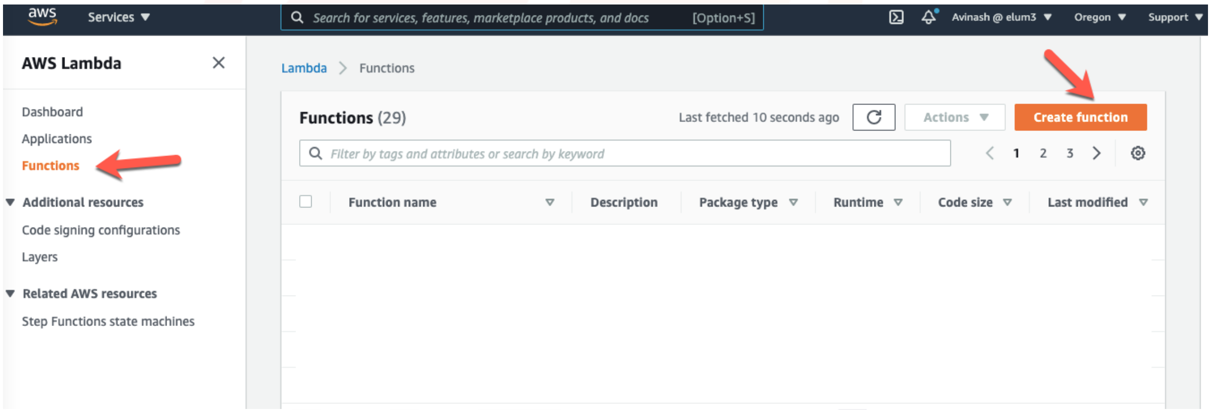
Summary

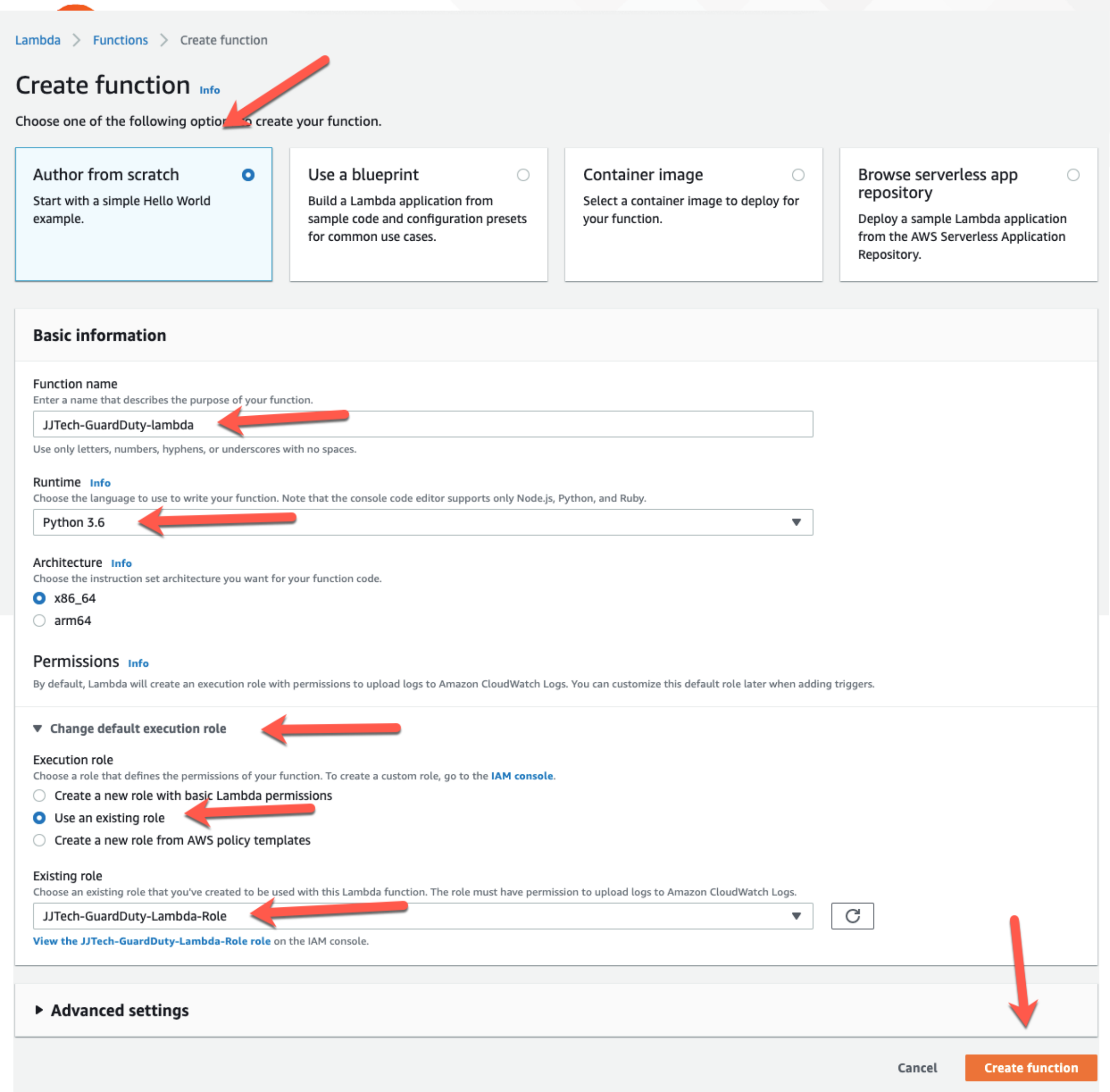
Service	Access level	Resource	Request condition
Allow (1 of 299 services) Show remaining 298			
SNS	Limited: Write	TopicName | string like | JJTech-GuarDuty-SNS	None

* Required

Step 3: Create Lambda Function

- Navigate to Lambda console





Lambda > Functions > Create function

Create function Info

Choose one of the following options to create your function.

● Author from scratch
Start with a simple Hello World example.

○ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

○ Container image
Select a container image to deploy for your function.

○ Browse serverless app repository
Deploy a sample Lambda application from the AWS Serverless Application Repository.

Basic information

Function name
Enter a name that describes the purpose of your function.
JJTech-GuardDuty-lambda
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.6

Architecture Info
Choose the instruction set architecture you want for your function code.
● x86_64
○ arm64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ Change default execution role

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
○ Create a new role with basic Lambda permissions
● Use an existing role
○ Create a new role from AWS policy templates

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
JJTech-GuardDuty-Lambda-Role
View the JJTech-GuardDuty-Lambda-Role role on the IAM console.

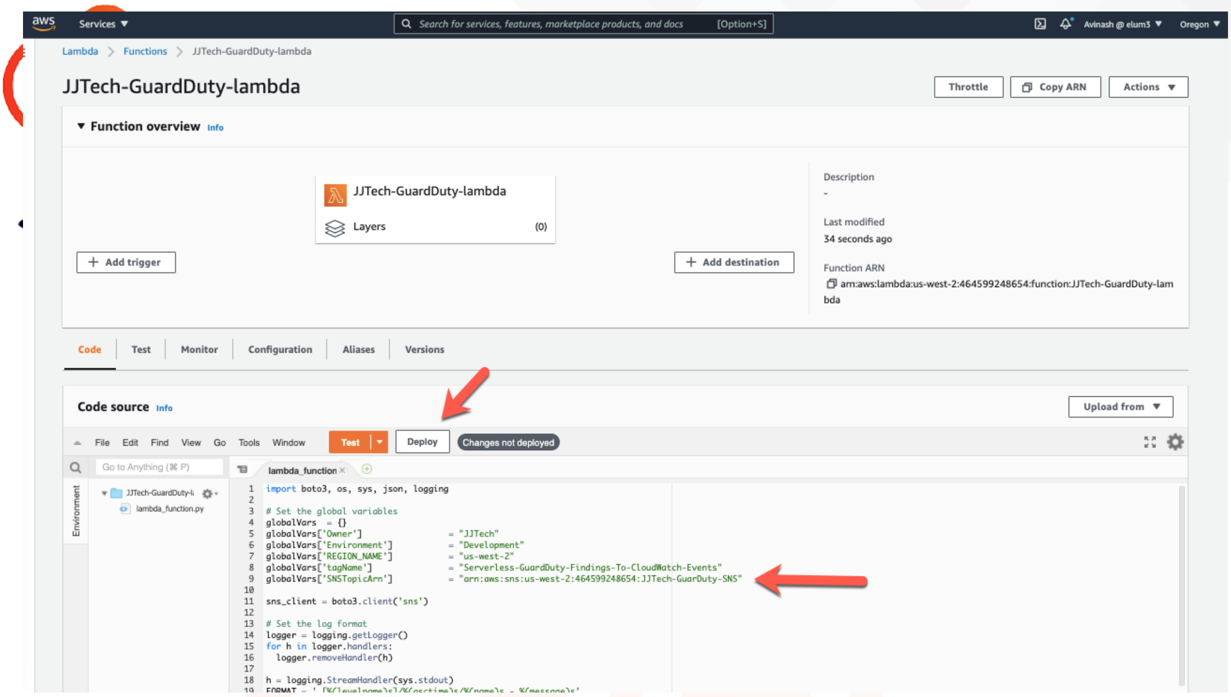
▶ **Advanced settings**

Cancel Create function

- Get the code from below link

https://raw.githubusercontent.com/avinashmamidi/Serverless-GuardDuty-Findings-to-SNS/master/Serverless-GuardDuty-Findings-To-SNS.py

- Change line 6 with your SNS topic ARN you created at Step 1
- Then Deploy the new code to the lambda function



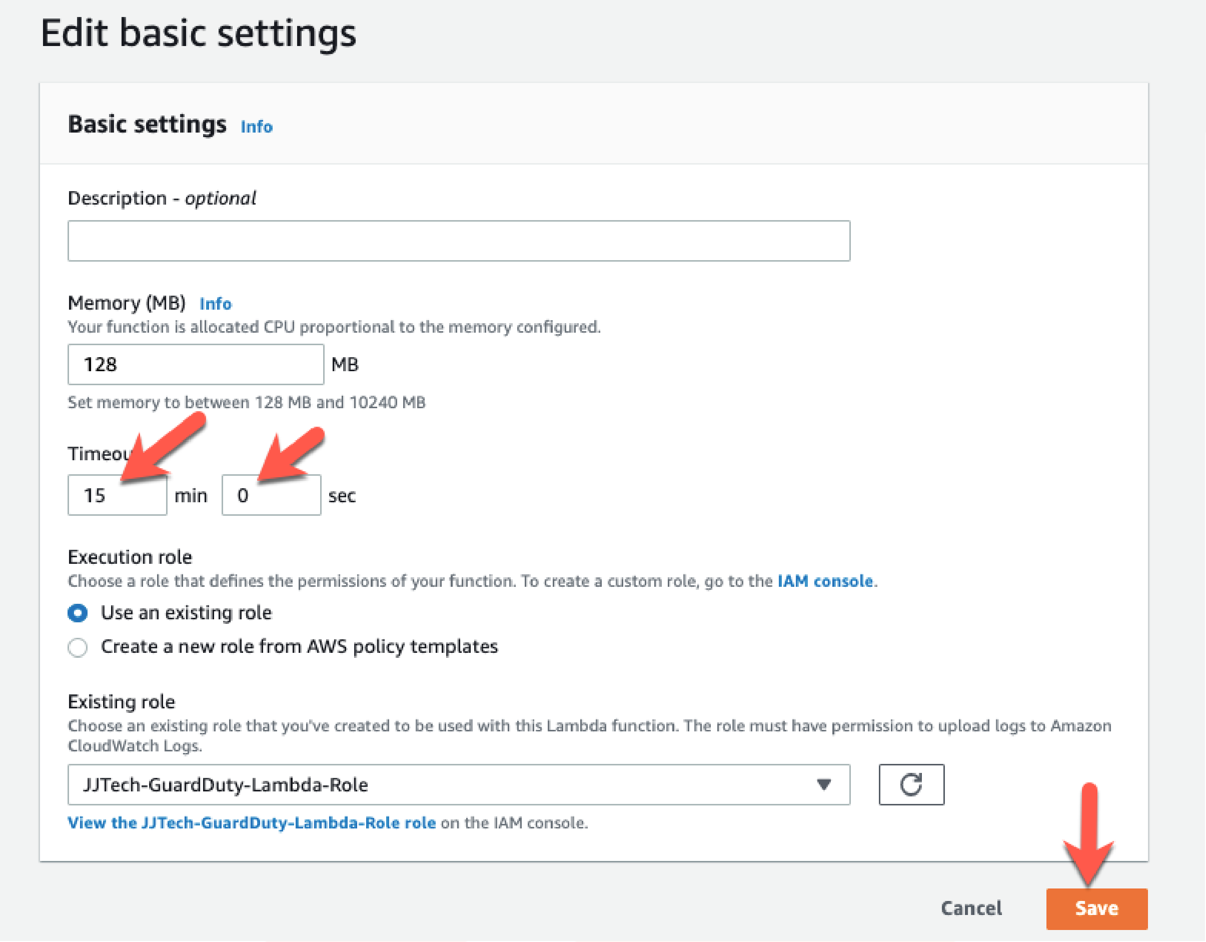
Description
-

Last modified
34 seconds ago

Function ARN
arn:aws:lambda:us-west-2:464599248654:function:JJTech-GuardDuty-lambda

```
import boto3, os, sys, json, logging

# Set the global variables
globalVars  = {}
globalVars['Owner']                 = "JJTech"
globalVars['Environment']           = "Development"
globalVars['REGION_NAME']           = "us-west-2"
globalVars['tagName']               = "Serverless-GuardDuty-Findings-To-CloudWatch-Events"
globalVars['SNSTopicArn']           = "arn:aws:sns:us-west-2:464599248654:JJTech-GuarDuty-SNS"

sns_client = boto3.client('sns')

# Set the log format
logger = logging.getLogger()
for h in logger.handlers:
  logger.removeHandler(h)

h = logging.StreamHandler(sys.stdout)
FORMAT = ' [%(levelname)s]/%(asctime)s/%(name)s - %(message)s'
```

- Now we have to increase the lambda timeout



Edit basic settings

Basic settings Info

Description - *optional*

Memory (MB) Info
Your function is allocated CPU proportional to the memory configured.
128 MB
Set memory to between 128 MB and 10240 MB

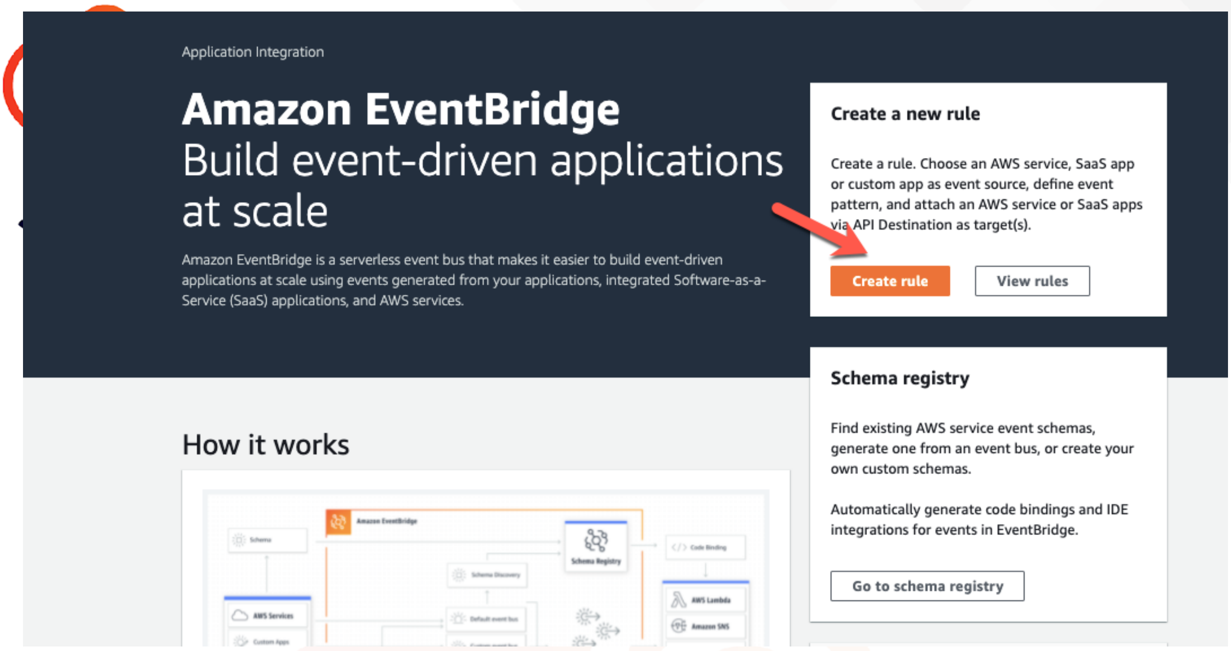
Timeout
15 min 0 sec

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
● Use an existing role
○ Create a new role from AWS policy templates

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
JJTech-GuardDuty-Lambda-Role
View the JJTech-GuardDuty-Lambda-Role role on the IAM console.

STEP 4: Create Cloudwatch rule to trigger the lambda

- Create Cloudwatch rule to trigger lambda whenever New GuardDuty finding is available. So that we will get notified with the details to our mail
- Navigate to EventBridge console



Application Integration

Amazon EventBridge
Build event-driven applications at scale

Amazon EventBridge is a serverless event bus that makes it easier to build event-driven applications at scale using events generated from your applications, integrated Software-as-a-Service (SaaS) applications, and AWS services.

Create a new rule

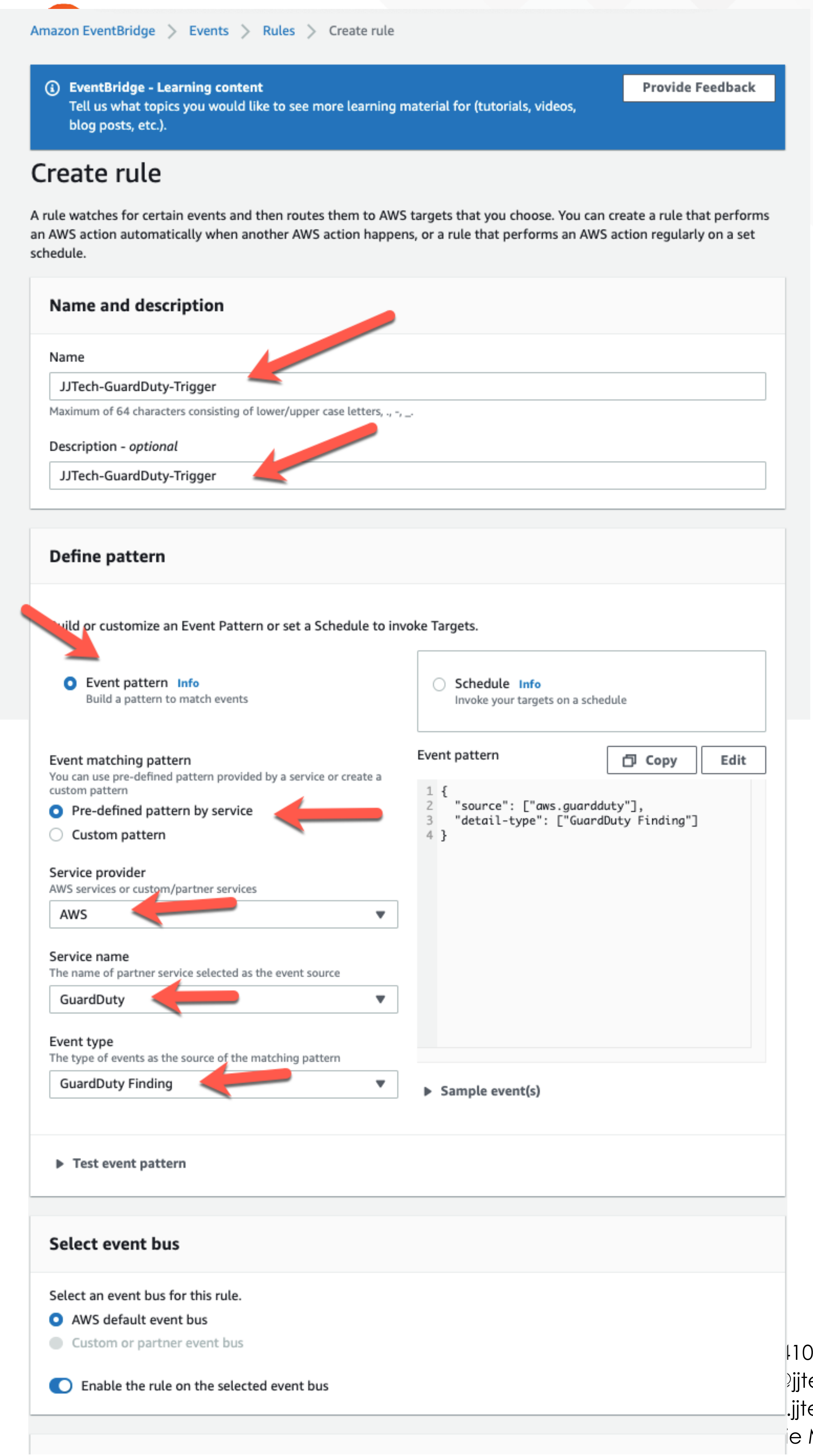
Create a rule. Choose an AWS service, SaaS app or custom app as event source, define event pattern, and attach an AWS service or SaaS apps via API Destination as target(s).

Schema registry

Find existing AWS service event schemas, generate one from an event bus, or create your own custom schemas.

Automatically generate code bindings and IDE integrations for events in EventBridge.

How it works



Amazon EventBridge > Events > Rules > Create rule

EventBridge - Learning content
Tell us what topics you would like to see more learning material for (tutorials, videos, blog posts, etc.).

Provide Feedback

Create rule

A rule watches for certain events and then routes them to AWS targets that you choose. You can create a rule that performs an AWS action automatically when another AWS action happens, or a rule that performs an AWS action regularly on a set schedule.

Name and description

Name

JJTech-GuardDuty-Trigger

Maximum of 64 characters consisting of lower/upper case letters, ., -, _.

Description - *optional*

JJTech-GuardDuty-Trigger

Define pattern

uild or customize an Event Pattern or set a Schedule to invoke Targets.

● Event pattern Info
Build a pattern to match events

○ Schedule Info
Invoke your targets on a schedule

Event matching pattern
You can use pre-defined pattern provided by a service or create a custom pattern

● Pre-defined pattern by service
○ Custom pattern

Service provider
AWS services or custom/partner services

AWS ▼

Service name
The name of partner service selected as the event source

GuardDuty ▼

Event type
The type of events as the source of the matching pattern

GuardDuty Finding ▼

Event pattern   Copy   Edit

```
{
  "source": ["aws.guardduty"],
  "detail-type": ["GuardDuty Finding"]
}
```

▶ Sample event(s)

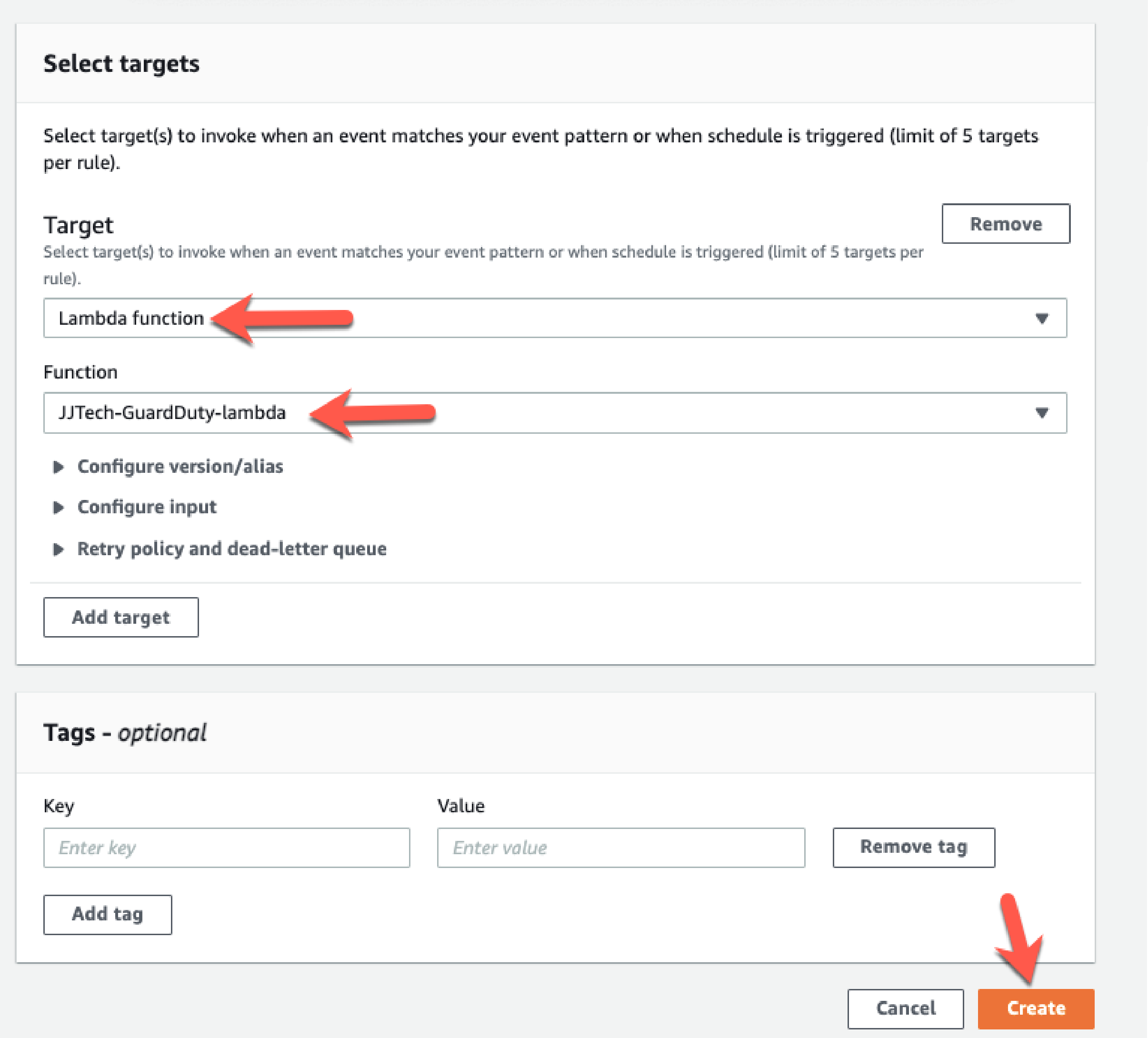
▶ Test event pattern

Select event bus

Select an event bus for this rule.

● AWS default event bus
○ Custom or partner event bus

Enable the rule on the selected event bus



Select targets

Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule).

Target

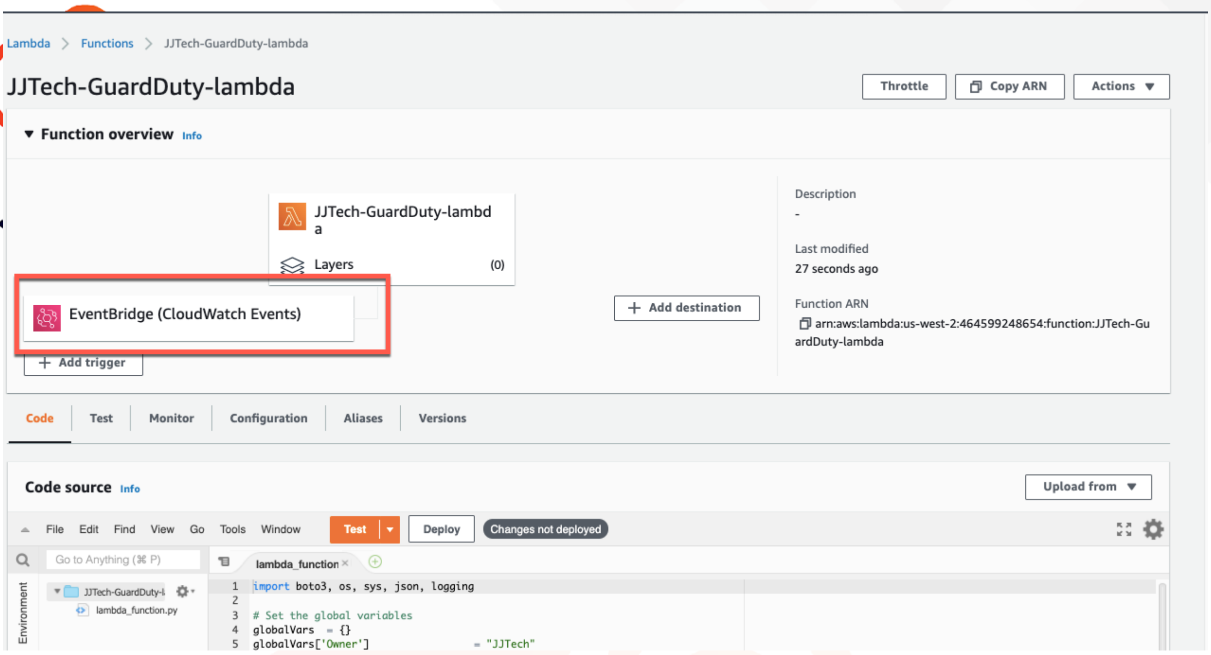
Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule).

Lambda function

Function

JJTech-GuardDuty-lambda

- Now if you navigate to Lambda console you should see below trigger to be configured



JJTech-GuardDuty-lambda

▼ Function overview

JJTech-GuardDuty-lambda

EventBridge (CloudWatch Events)

Description
-

Last modified
27 seconds ago

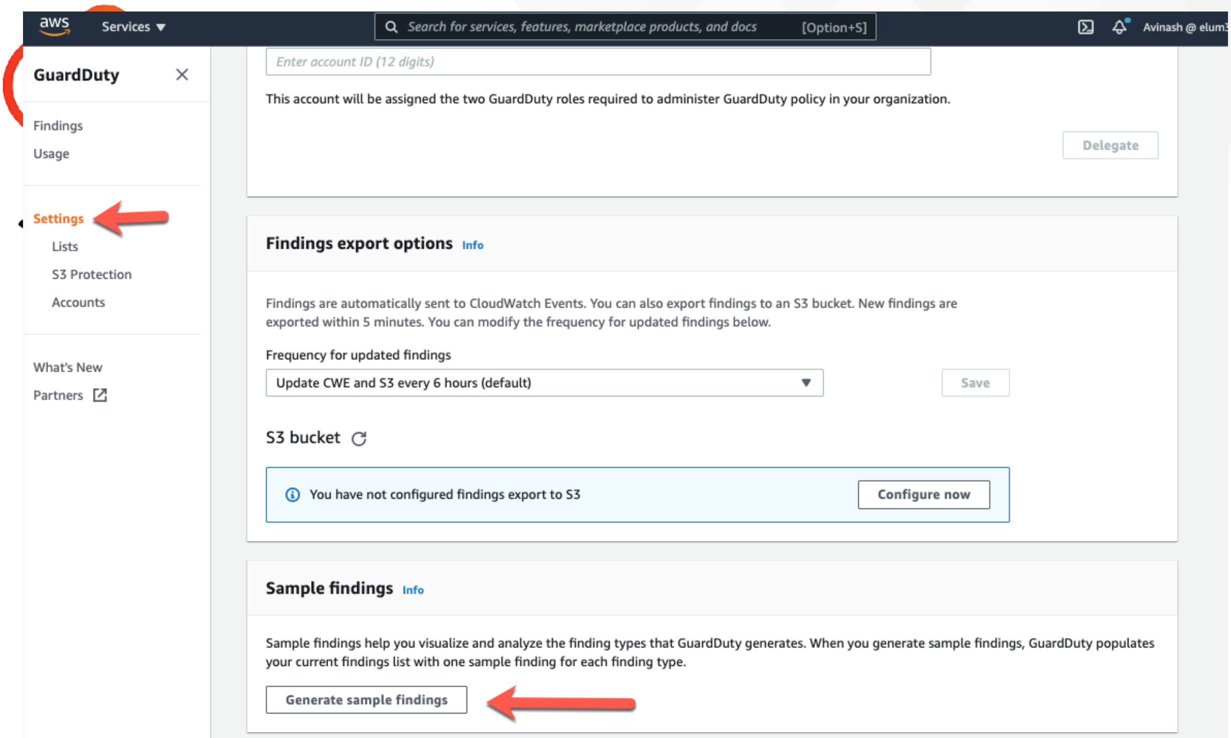
Function ARN
arn:aws:lambda:us-west-2:464599248654:function:JJTech-GuardDuty-lambda

Code source

```
import boto3, os, sys, json, logging

# Set the global variables
globalVars  = {}
globalVars['Owner']                 = "JJTech"
```

Step 5: Test the solution

- Navigate to Amazon GuardDuty console
- Let’s create some sample findings in Guardduty



Enter account ID (12 digits)

This account will be assigned the two GuardDuty roles required to administer GuardDuty policy in your organization.

Delegate

Findings export options Info

Findings are automatically sent to CloudWatch Events. You can also export findings to an S3 bucket. New findings are exported within 5 minutes. You can modify the frequency for updated findings below.

Frequency for updated findings: Update CWE and S3 every 6 hours (default) ▼ Save

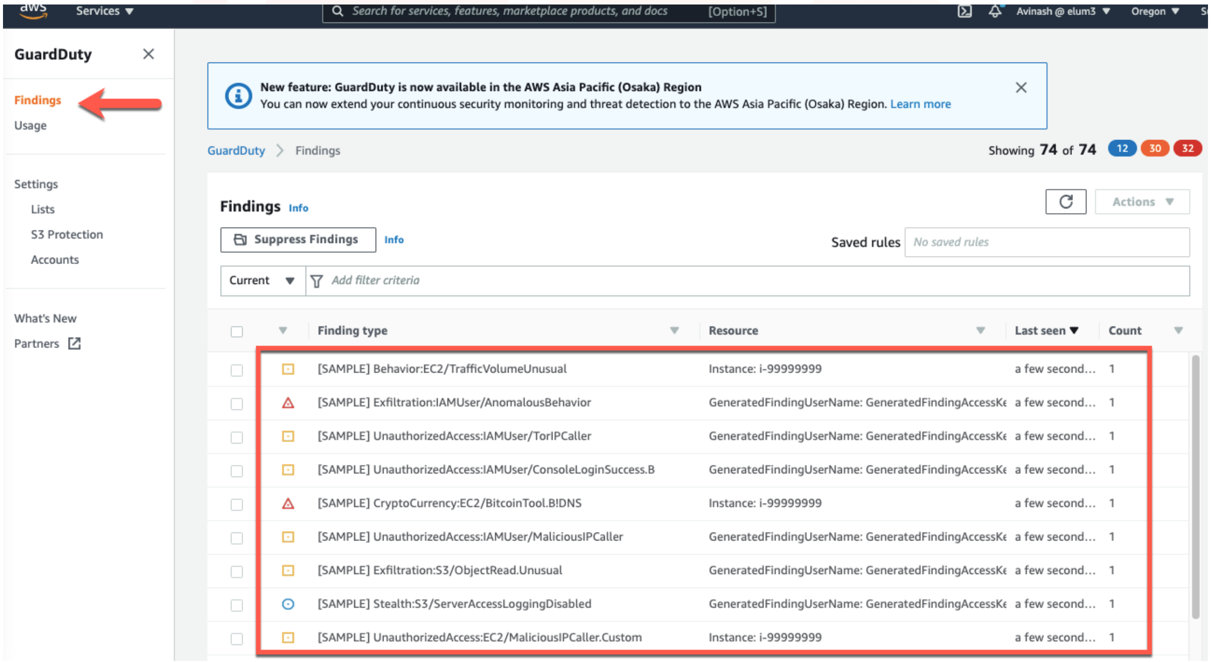
S3 bucket

ⓘ You have not configured findings export to S3 — Configure now

Sample findings Info

Sample findings help you visualize and analyze the finding types that GuardDuty generates. When you generate sample findings, GuardDuty populates your current findings list with one sample finding for each finding type.

Generate sample findings



New feature: GuardDuty is now available in the AWS Asia Pacific (Osaka) Region
You can now extend your continuous security monitoring and threat detection to the AWS Asia Pacific (Osaka) Region. Learn more

GuardDuty > Findings

Showing 74 of 74 12 30 32

Findings Info

Suppress Findings Info

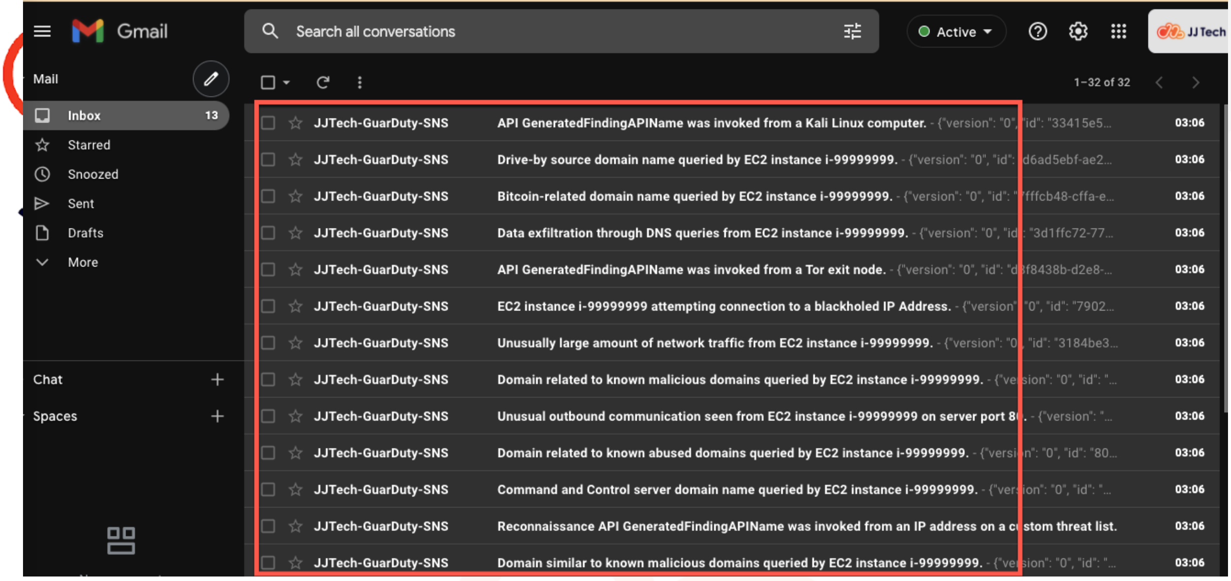
Saved rules: No saved rules

Actions ▼

Current ▼ Add filter criteria

□		Finding type	Resource	Last seen ▼	Count
□	□	[SAMPLE] Behavior:EC2/TrafficVolumeUnusual	Instance: i-99999999	a few second...	1
□	△	[SAMPLE] Exfiltration:IAMUser/AnomalousBehavior	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	□	[SAMPLE] UnauthorizedAccess:IAMUser/TorIPCaller	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	□	[SAMPLE] UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	△	[SAMPLE] CryptoCurrency:EC2/BitcoinTool.B!DNS	Instance: i-99999999	a few second...	1
□	□	[SAMPLE] UnauthorizedAccess:IAMUser/MaliciousIPCaller	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	□	[SAMPLE] Exfiltration:S3/ObjectRead.Unusual	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	○	[SAMPLE] Stealth:S3/ServerAccessLoggingDisabled	GeneratedFindingUserName: GeneratedFindingAccessKe	a few second...	1
□	□	[SAMPLE] UnauthorizedAccess:EC2/MaliciousIPCaller.Custom	Instance: i-99999999	a few second...	1

- You will receive mails whenever new finding is available in GuardDuty
- Based on the issue you have to fix them



JJTech-GuarDuty-SNS	API GeneratedFindingAPIName was invoked from a Kali Linux computer. - {"version": "0", "id": "33415e5...	03:06
JJTech-GuarDuty-SNS	Drive-by source domain name queried by EC2 instance i-99999999. - {"version": "0", "id": "d6ad5ebf-ae2...	03:06
JJTech-GuarDuty-SNS	Bitcoin-related domain name queried by EC2 instance i-99999999. - {"version": "0", "id": "fffcb48-cffa-e...	03:06
JJTech-GuarDuty-SNS	Data exfiltration through DNS queries from EC2 instance i-99999999. - {"version": "0", "id": "3d1ffc72-77...	03:06
JJTech-GuarDuty-SNS	API GeneratedFindingAPIName was invoked from a Tor exit node. - {"version": "0", "id": "d3f8438b-d2e8-...	03:06
JJTech-GuarDuty-SNS	EC2 instance i-99999999 attempting connection to a blackholed IP Address. - {"version": "0", "id": "7902...	03:06
JJTech-GuarDuty-SNS	Unusually large amount of network traffic from EC2 instance i-99999999. - {"version": "0", "id": "3184be3...	03:06
JJTech-GuarDuty-SNS	Domain related to known malicious domains queried by EC2 instance i-99999999. - {"version": "0", "id": "...	03:06
JJTech-GuarDuty-SNS	Unusual outbound communication seen from EC2 instance i-99999999 on server port 80. - {"version": "...	03:06
JJTech-GuarDuty-SNS	Domain related to known abused domains queried by EC2 instance i-99999999. - {"version": "0", "id": "80...	03:06
JJTech-GuarDuty-SNS	Command and Control server domain name queried by EC2 instance i-99999999. - {"version": "0", "id": "...	03:06
JJTech-GuarDuty-SNS	Reconnaissance API GeneratedFindingAPIName was invoked from an IP address on a custom threat list.	03:06
JJTech-GuarDuty-SNS	Domain similar to known malicious domains queried by EC2 instance i-99999999. - {"version": "0", "id": "...	03:06

- For example If an EC2 instance got compromised try to terminate
- If an EC2 instance is getting malicious traffic try to block them in NACL or restrict access to respective networks in Security group and so on