

[illegible]

Name	GuardDuty : S3 Findings
URL	https://attackdefense.com/challengedetails?cid=2472
Type	AWS Cloud Security : Defense

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

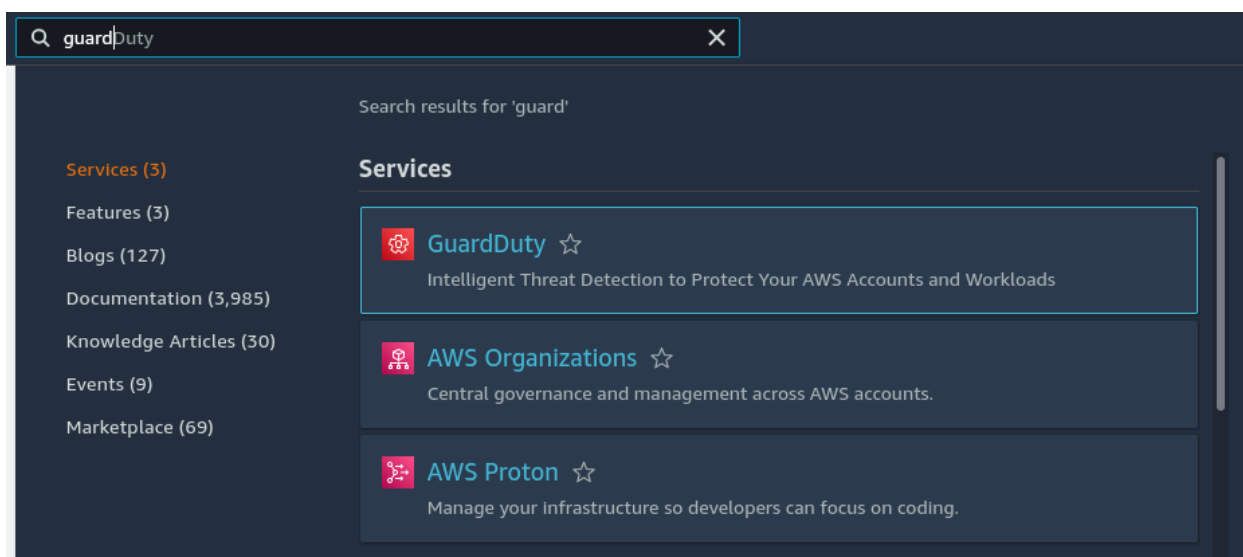
Solution:

Step 1: Click the lab link button to get access credentials.

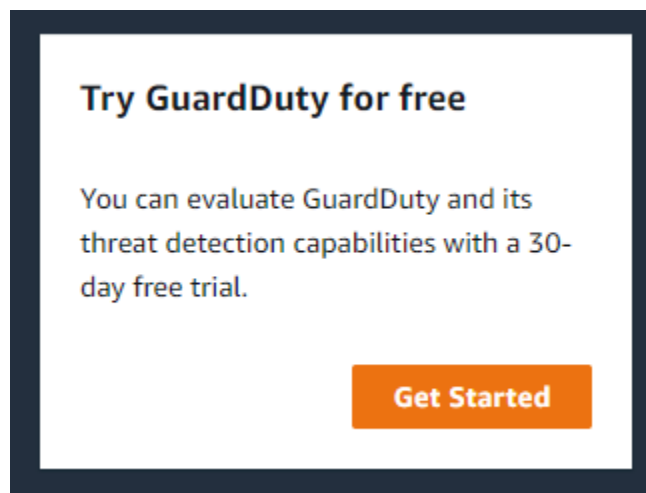
Access Credentials to your AWS lab Account

Login URL	https://230155374719.signin.aws.amazon.com/console
Region	US East (N. Virginia) us-east-1
Username	student
Password	Ad6fg9q7t554mRUD
Access Key ID	AKIATLFSUKB7V3OZJJ4W
Secret Access Key	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

Step 2: Enable GuardDuty from the console. Search for GuardDuty in the search bar and navigate to the GuardDuty dashboard.



Step 3: Click on Get Started.



Step 4: Click on Enable GuardDuty.

data, events, or logs available to you. You can configure

in a 30 day [GuardDuty free trial](#). [Learn more](#)

Enable GuardDuty

There will not be any findings at first.



Resource



You don't have any findings.

GuardDuty continuously monitors your AWS environment and reports findings on this page.

[Learn more](#)

Step 5: Navigate to Usage. This will display the Estimated total daily cost by the data source.

GuardDuty ×

Findings
Usage
Malware scans

Settings
Lists
S3 Protection
Kubernetes Protection
Malware Protection **New!**
Accounts

What's New
Partners

Usage [Info](#)

Estimated total daily cost **\$0.00**

[About GuardDuty pricing](#)

- Some features are still in free trial. You pay nothing for these features while free trials are in effect. These estimates reflect what you can expect to pay after your free trial ends.

Breakdown by data source Average daily cost

CloudTrail Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)
VPC Flow Logs Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)
DNS Logs Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)
S3 Data Events Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)
Kubernetes Audit Logs Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)
Malware Protection Daily cost will be available 7 days after enabling data source.	Pending Free trial ends September 15 (30 days remaining)

Step 6: Configure AWS CLI with given credentials and interact with S3 buckets.
Configure AWS CLI using the following command.

Command: aws configure

```
(kali㉿kali)-[~/lab]
$ aws configure
AWS Access Key ID [*****JCY6]: AKIAU77NXJR6BHEBSZXY
AWS Secret Access Key [*****r3qV]: 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
Default region name [us-east-1]:
Default output format [None]:

(kali㉿kali)-[~/lab]
$
```

Step 7: List the S3 buckets using S3 API.

Command: `aws s3api list-buckets --query "Buckets[].Name"`

```
(kali㉿kali)-[~/lab]
$ aws s3api list-buckets --query "Buckets[].Name"
[]

(kali㉿kali)-[~/lab]
$
```

Step 8: Fetch the account id.

Command: `aws sts get-caller-identity`

```
(kali㉿kali)-[~/lab]
$ aws sts get-caller-identity
{
  "UserId": "AIDAU77NXJR6MGX3JILYU",
  "Account": "343559064700",
  "Arn": "arn:aws:iam::343559064700:user/student"
}

(kali㉿kali)-[~/lab]
$
```

Step 9: Create an S3 bucket.

Command:

```
aws s3api create-bucket \
  --bucket lab-bucket-343559064700 \
  --region us-east-1
```

```
(kali㉿kali)-[~/lab]
$ aws s3api create-bucket \
  --bucket lab-bucket-343559064700 \
  --region us-east-1
{
  "Location": "/lab-bucket-343559064700"
}
```

Step 10: Allow the public access to the bucket by removing “public block access”.

Command:

```
aws s3api delete-public-access-block \
  --bucket lab-bucket-343559064700
```

```
(kali㉿kali)-[~/lab]
$ aws s3api delete-public-access-block \
  --bucket lab-bucket-343559064700
```

Step 11: Grant public access to an S3 bucket to all AWS users by changing ACL.

Command:

```
aws s3api put-bucket-acl --bucket lab-bucket-343559064700 \
  --grant-full-control uri=http://acs.amazonaws.com/groups/global/AuthenticatedUsers
```

```
(kali㉿kali)-[~/lab]
$ aws s3api put-bucket-acl --bucket lab-bucket-343559064700 \
  --grant-full-control uri=http://acs.amazonaws.com/groups/global/AuthenticatedUsers
```

Step 12: Fetch your public IP address.

Command: echo \$(curl -s <https://api.ipify.org>)

This will output the public IP address.


```
(kali㉿kali)-[~/lab]
$ echo $(curl -s https://api.ipify.org)
103.177.252.62
```

Step 13: Add the public IP to a plain text file and name it as “ip_list.txt”.

Command: echo “103.177.252.62” > ip_list.txt

```
(kali㉿kali)-[~/lab]
$ echo "103.177.252.62" > ip_list.txt
```

Step 14: Upload the “ip_list.txt” to the created S3 bucket.

Command: aws s3 cp ip_list.txt s3://lab-bucket-343559064700/

```
(kali㉿kali)-[~/lab]
$ aws s3 cp ip_list.txt s3://lab-bucket-343559064700/
upload: ./ip_list.txt to s3://lab-bucket-343559064700/ip_list.txt
```

We got an S3 URI as an output. Convert the URI to an object URL using this syntax.

Syntax: https://<bucket-name>.s3.amazonaws.com/<object or key name>

So finally the object URL will be similar to the following one.

Object URL: https://lab-bucket-343559064700.s3.amazonaws.com/ip_list.txt

Step 15: Navigate back to the GuardDuty dashboard click on “Lists” from the side navigation bar.

Settings

Lists

S3 Protection

Kubernetes Protection

Malware Protection New!

Accounts

Step 16: Click on “Add a threat IP list”.

Threat lists consist of known malicious IP addresses. These lists can be supplied by third party threat intelligence or created specifically for your organization. GuardDuty generates findings based on threat lists. You can include a maximum of 250,000 IP addresses and CIDR ranges in a single threat list. GuardDuty only generates findings based on activity that involves IP addresses and CIDR ranges in your threat lists, findings will not be generated based on domain names. At any given time, you can have up to six uploaded threat lists per AWS account per Region.

Threat IP lists

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

Add a threat IP list

List name	List file URL
<div><div></div><div>Threat IP lists Threat IP lists consist of known malicious IP addresses. GuardDuty generates findings based on threat lists. Learn more</div></div>	

Step 17: Enter the “List name” as “ip_list”. Paste object URL in the “Location” field and choose “Plaintext” as format.

Add a threat IP list



GuardDuty generates findings based on threat IP lists. [Learn more](#)

List name

Location

Format

Plaintext ▼

Use TXT for files that contain simple IP lists.

By adding this list, you accept and agree to the GuardDuty service terms, including those related to third-party threat intelligence, and direct GuardDuty to read data from this resource.



☒ I agree

Cancel

Add list

Step 18: After adding the list, activate the list by checking the checkbox.

Added in threat IP lists.

	Active
<input checked="" type="checkbox"/>	 

Step 19: Go back to the console and try to interact with S3 buckets. List S3 buckets.

Command: aws s3api list-buckets --query "Buckets[].Name"

```
(kali㉿kali)-[~/lab]
$ aws s3api list-buckets --query "Buckets[].Name"
[
    "lab-bucket-343559064700"
]

(kali㉿kali)-[~/lab]
$
```

Step 20: List the objects inside the buckets.

Command: aws s3api list-objects --bucket lab-bucket-343559064700 --query 'Contents[].{Key: Key, Size: Size}'

```
(kali㉿kali)-[~/lab]
$ aws s3api list-objects --bucket lab-bucket-343559064700 --query 'Contents[].{Key: Key, Size: Size}'
[
    {
        "Key": "ip_list.txt",
        "Size": 15
    }
]

(kali㉿kali)-[~/lab]
$
```

Step 21: Download the object.

Command: aws s3api get-object --bucket lab-bucket-343559064700 --key ip_list.txt out.txt

```
(kali㉿kali)-[~/lab]
$ aws s3api get-object --bucket lab-bucket-343559064700 --key ip_list.txt out.txt
{
  "AcceptRanges": "bytes",
  "LastModified": "2022-08-25T08:46:31+00:00",
  "ContentLength": 15,
  "ETag": "\"7a31ef71ed931eb701a38567a5d79a7d\"",
  "ContentType": "text/plain",
  "Metadata": {}
}

(kali㉿kali)-[~/lab]
$
```

Step 21: Navigate back to the GuardDuty dashboard and check the findings.

Note: Refresh the findings page if the findings are empty. Detecting findings might get delayed.

Each GuardDuty finding has an assigned severity level and value that reflects the potential risk. GuardDuty breaks down this range into High, Medium, and Low severity levels.

<input type="checkbox"/>	▼	Finding type
<input type="checkbox"/>	🟡	Recon:IAMUser/MaliciousIPCaller.Custom
<input type="checkbox"/>	🟡	UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom
<input type="checkbox"/>	🟡	PenTest:IAMUser/KaliLinux
<input type="checkbox"/>	🔴	Policy:S3/BucketPublicAccessGranted
<input type="checkbox"/>	🟢	Policy:S3/BucketBlockPublicAccessDisabled

Step 22: Click on finding with the name “PenTest:IAMUser/KaliLinux”.

This finding informs you that a machine running Kali Linux is making API calls using credentials that belong to the listed AWS account in your environment.

PenTest:IAMUser/KaliLinux 🔍

Finding ID: 2ec16a16620b24f2b647e6042560730e

[Feedback](#)

Medium API PutBucketAcl was invoked from a remote host with IP address 103.177.252.62 that is potentially running the Kali Linux penetration testing tool. [Info](#)

[Investigate with Detective](#)

Overview

Severity	MEDIUM	🔍
Region	us-east-1	
Count	4	
Account ID	343559064700	🔍
Resource ID	No information available	
Created at	08-25-2022 14:14:39 (20 minutes ago)	
Updated at	08-25-2022 14:19:50 (15 minutes ago)	

This finding will provide the details about the affected resource. Here the affected resources are IAM role and S3 bucket. It will also provide the action done. Here the action was to change bucket ACL and it is listed as "PutBucketAcl".

Resource affected

Resource role	TARGET	🔍
Resource type	AccessKey	🔍
Access key ID	AKIAU77NXJR6BHEBSZXY	🔍
Principal ID	AIDAU77NXJR6MGX3JILYU	🔍
User type	IAMUser	🔍
User name	student	🔍

Affected resources

Action

Action type	AWS_API_CALL	🔍
API	PutBucketAcl	🔍
Service name	s3.amazonaws.com	🔍
First seen	08-25-2022 14:08:46 (26 minutes ago)	
Last seen	08-25-2022 14:13:31 (21 minutes ago)	

Step 23: Click on finding with the name “Policy:S3/BucketPublicAccessGranted”.

This finding informs you that the listed S3 bucket has been publicly exposed to all authenticated AWS users because an IAM entity has changed a bucket policy or ACL on that S3 bucket.

Policy:S3/BucketPublicAccessGranted 🔍

Finding ID: 38c16a18c0b6595497cbaa718ac2cb69 [Feedback](#)

High The Amazon S3 bucket lab-bucket-343559064700 was granted public authenticated access by student calling PutBucketAcl. If this behavior is not expected, it may indicate a configuration mistake or that your credentials are compromised. [Info](#)

[Investigate with Detective](#)

Overview	
Severity	HIGH 🔍
Region	us-east-1
Count	1
Account ID	343559064700 🔍
Resource ID	lab-bucket-343559064700 🔗
Created at	08-25-2022 14:19:50 (16 minutes ago)
Updated at	08-25-2022 14:19:50 (16 minutes ago)

Here the affected resources are IAM role and S3 bucket.

Resource affected		
Resource role	TARGET	🔍 🔍
Resource type	AccessKey	🔍 🔍
Access key ID	AKIAU77NXJR6BHEBSZXY	🔍 🔍
Principal ID	AIDAU77NXJR6MGX3JILYU	🔍 🔍
User type	IAMUser	🔍 🔍
User name	student	🔍 🔍
Affected resources		
AWS::S3::Bucket	lab-bucket-343559064700	
S3 buckets		
Destination: lab-bucket-343559064700		
Name	lab-bucket-343559064700 🔗	🔍 🔍
Type	Destination	🔍 🔍
ARN	arn:aws:s3:::lab-bucket-343559064700	
Effective permission	PUBLIC	🔍 🔍
Created at	08-25-2022 08:40:25 UTC	
Owner		
ID	9025e4e628e49f9028ed2106f8812264ff6768df55f...	

It will also provide the action done. Here the action was to change bucket ACL and it is listed as “PutBucketAcl”.

Action		
Action type	AWS_API_CALL	🔍 🔍
API	PutBucketAcl	🔍 🔍
Service name	s3.amazonaws.com	🔍 🔍
First seen	08-25-2022 14:13:31 (22 minutes ago)	
Last seen	08-25-2022 14:13:31 (22 minutes ago)	

Step 24: Click on finding with the name “Policy:S3/BucketBlockPublicAccessDisabled”.

This finding informs you that Block Public Access was disabled for the listed S3 bucket. When enabled, S3 Block Public Access settings are used to filter the policies or access control lists (ACLs) applied to buckets as a security measure to prevent inadvertent public exposure of data.

Typically, S3 Block Public Access is turned off on a bucket to allow public access to the bucket or to the objects within. When S3 Block Public Access is disabled for a bucket, access to the bucket is controlled by the policies or ACLs applied to it. This does not mean that the bucket is shared publicly, but you should audit the policies and ACLs applied to the bucket to confirm that appropriate permissions are applied.

Policy:S3/BucketBlockPublicAccessDisabled 🔍 🔍
Finding ID: 02c16a17c21a556c5c818ae31701a808 [Feedback](#)

Low Amazon S3 Block Public Access was disabled for S3 bucket lab-bucket-343559064700 by student calling DeleteBucketPublicAccessBlock. If this behavior is not expected, it may indicate a configuration mistake or that your credentials are compromised. [Info](#)

[Investigate with Detective](#)

Overview

Severity	LOW	🔍 🔍
Region	us-east-1	
Count	1	
Account ID	343559064700	🔍 🔍
Resource ID	lab-bucket-343559064700 🔗	
Created at	08-25-2022 14:17:40 (19 minutes ago)	
Updated at	08-25-2022 14:17:40 (19 minutes ago)	

Resource affected

Resource role	TARGET	🔍 🔍
Resource type	AccessKey	🔍 🔍
Access key ID	AKIAU77NXJR6BHEBSZXY	🔍 🔍
Principal ID	AIDAU77NXJR6MGX3JILYU	🔍 🔍
User type	IAMUser	🔍 🔍
User name	student	🔍 🔍

Here the affected resources are IAM role and S3 bucket. The action was to disable Block Public Access.

Affected resources		
AWS::S3::Bucket	lab-bucket-343559064700	
S3 buckets		
Destination: lab-bucket-343559064700		
Name	lab-bucket-343559064700 ↗	🔍 🔍
Type	Destination	🔍 🔍
ARN	arn:aws:s3:::lab-bucket-343559064700	
Effective permission	PUBLIC	🔍 🔍
Created at	08-25-2022 08:40:25 UTC	
Owner		
ID	9025e4e628e49f9028ed2106f8812264ff6768df55f...	
Action		
Action type	AWS_API_CALL	🔍 🔍
API	DeleteBucketPublicAccessBlock	🔍 🔍
Service name	s3.amazonaws.com	🔍 🔍
First seen	08-25-2022 14:12:20 (24 minutes ago)	
Last seen	08-25-2022 14:12:20 (24 minutes ago)	

Step 25: Click on finding with the name “Recon:IAMUser/MaliciousIPCaller.Custom”

This finding informs you that an API operation that can list or describe AWS resources in an account within your environment was invoked from an IP address that is included on a custom threat list. The threat list used will be listed in the finding's details. An attacker might use stolen credentials to perform this type of reconnaissance of your AWS resources in order to find more valuable credentials or determine the capabilities of the credentials they already have.

Recon:IAMUser/MaliciousIPCaller.Custom 🔍

Finding ID: 30c16a26e7c21f1d1a57be640a45fdb9

[Feedback](#)

Medium API ListFindings, commonly used in reconnaissance attacks, was invoked from an IP address 103.177.252.62 on the custom threat list ip_list. Unauthorized actors perform such activity to gather information and discover resources like databases, S3 buckets etc., in order to further tailor the attack. [Info](#)

[Investigate with Detective](#)

Overview

Severity	MEDIUM	🔍
Region	us-east-1	
Count	2	
Account ID	343559064700	🔍
Resource ID	No information available	
Created at	08-25-2022 14:50:45 (4 minutes ago)	
Updated at	08-25-2022 14:50:45 (4 minutes ago)	

Resource affected

Resource role	TARGET	🔍
Resource type	AccessKey	🔍
Access key ID	ASIAU77NXJR6CIJ6LI6W	🔍
Principal ID	AIDAU77NXJR6MGX3JILYU	🔍
User type	IAMUser	🔍
User name	student	🔍

Affected resources

Action

Action type	AWS_API_CALL	🔍
API	ListFindings	🔍
Service name	guardduty.amazonaws.com	🔍
First seen	08-25-2022 14:41:43 (13 minutes ago)	
Last seen	08-25-2022 14:41:45 (13 minutes ago)	

Step 26: Click on finding with the name “UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom”.

This finding informs you that an API operation was invoked from an IP address that is included on a threat list that you uploaded. In , a threat list consists of known malicious IP addresses. This can indicate unauthorized access to AWS resources within your environment.

Medium API GetFindings was invoked from an IP address 103.177.252.62 on the custom threat list ip_list. [Info](#)

[Investigate with Detective](#)

Overview

Severity	MEDIUM	🔍🔍
Region	us-east-1	
Count	4	
Account ID	343559064700	🔍🔍
Resource ID	No information available	
Created at	08-25-2022 14:50:45 (4 minutes ago)	
Updated at	08-25-2022 14:50:45 (4 minutes ago)	


Resource affected

Resource role	TARGET	🔍🔍
Resource type	AccessKey	🔍🔍
Access key ID	ASIAU77NXJR6CIJ6LI6W	🔍🔍
Principal ID	AIDAU77NXJR6MGX3JILYU	🔍🔍
User type	IAMUser	🔍🔍
User name	student	🔍🔍

Affected resources

Action

Action type	AWS_API_CALL	🔍🔍
API	GetFindings	🔍🔍
Service name	guardduty.amazonaws.com	🔍🔍
First seen	08-25-2022 14:41:43 (13 minutes ago)	
Last seen	08-25-2022 14:41:45 (13 minutes ago)	



Thus GuardDuty generates a finding whenever it detects unexpected and potentially malicious activity in your AWS environment.

References:

1. Amazon GuardDuty
(https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_setup.html)