

IBM Project Report

On

Detecting Malware Infection on Infrastructure Hosted in IaaS Cloud using Cloud Visibility and Forensics

Developed By: -

Jainam Shah (18162121033)

Het Patel (18162171018)

Harshvardhansinh Rahevar (18162101028)

Guided By: -

Prof. Ravindra Patel (Internal)

Mr. Anoj Dixit (External)

Submitted to
Department of Computer Science & Engineering
Institute of Computer Technology



Year: 2022



CERTIFICATE

This is to certify that the **IBM** Project work entitled “**Detecting Malware Infection on Infrastructure Hosted in IaaS Cloud using Cloud Visibility and Forensics**” by Jainam Shah(Enrolment No.18162121033), Het Patel(Enrolment No.18162171018) and Harshvardhansinh Rahevar (EnrolmentNo.18162101028) of Ganpat University, towards the partial fulfillment of requirements of the degree of Bachelor of Technology – Computer Science and Engineering, carried out by them in the CSE(CBA/BDA/CS) Department at Ganpat University Institute of Computer Technology. The results/findings contained in this Project have not been submitted in part or full to any other University / Institute for award of any other Degree/Diploma.

Name & Signature of Internal Guide

Name & Signature of Head

Place: ICT - GUNI

Date:

ACKNOWLEDGEMENT

IBM/Industry Internship project is a golden opportunity for learning and self-development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project. First and foremost, I would like to thank Dr. Hemal Shah, Principal, ICT, and Prof. Dharmesh Darji , Head, ICT who gave us an opportunity to undertake this project. My grateful thanks to Prof. Ravindra Patel & Mr. Anoj Dixit (Internal & External Guides) for their guidance in project work Detecting Malware Infection on Infrastructure Hosted in IaaS Cloud using Cloud Visibility and Forensics, who despite being extraordinarily busy with academics, took time out to hear, guide and keep us on the correct path. We do not know where would have been without their help. CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

JAINAM SHAH (Enrollment No:18162121033)

ABSTRACT

As Cloud computing has been adopted very rapidly by organizations with different businesses and sizes, the usage of cloud services is skyrocketing at an unprecedented rate these days especially IaaS services as cloud providers provide more robust resources with flexible offerings and models. This increasing adoption gives rise to new surface attacks to organizations that attackers abuse with their malware to take advantage of these powerful resources and the valuable data that exist on them. Therefore, for organizations to well defend against malware attacks they need to have full visibility not only on their data centers but also on their resources hosted on the cloud and don't take their security for granted. This proposed project discusses and aims to provide the best approaches to achieve continuous monitoring of malware attacks on the cloud along with their phases (before, during, and after). This project aims to defines the best methods to bring loggings and forensics to the cloud and integrate them with on-premises visibility, thus achieving the full monitoring over the whole security posture of the organization assets whether they are on-premises or on the cloud.

INDEX

Title	Page No
CHAPTER 1: INTRODUCTION	01-02
CHAPTER 2: PROJECT SCOPE	03-04
CHAPTER 3: SOFTWARE AND HARDWARE REQUIREMENT	05-06
CHAPTER 4: PROCESS MODEL	07-08
CHAPTER 5: PROJECT PLAN	09-11
5.1 List of Major Activities	10
5.1.1 Tasks for Building Prototype Model in First Phase	10
5.1.2 Time Duration to Complete First Phase	10
5.1.3 Task for Implementing Defect Detection in Second Phase	11
5.1.4 Time Duration to Complete Second Phase	11
CHAPTER 6: IMPLEMENTATION DETAILS	11-34
6.1 Background	12
6.2 Methodology	12
6.2.1 Gathering Data	12
6.3 Cloud Analysis to Malware Detection	13
6.3.1 Testing Environment	13
6.3.2 Data Set	13
6.3.2 Testing and Analysis	13
6.3.4 Testing Phases	13-25
6.3.5 Generating Data Logs from AWS CloudTrail	25-27
6.3.6 Integrating AWS CloudTrail with Splunk	27-31
6.3.7 Forensic Analysis After Malware Attack	32-34
CHAPTER 7: CONCLUSION AND FUTURE WORK	35-36
CHAPTER 8: REFERENCES	37-38

CHAPTER: 1 INTRODUCTION

CHAPTER 1 INTRODUCTION

The cloud is a technology that's not new anymore. Nowadays, using cloud services is increasing at an unprecedented pace, it has become more popular after the advent of the Fourth Industrial Revolution. In 2020, about 83% of business workloads operate in the cloud, and a whopping 94% of companies now use a cloud service in one form or shape. There are three most utilized cloud services include Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Infrastructure as a Service (IaaS) is one of the most critical and fastest-growing services in Cloud Computing.

However due to ample number of exquisite features being available on infrastructure hosted on the IaaS cloud, it is becoming targets to many attacks like malware for the following reasons:

- 1) Cloud service providers steadily offer higher performance with high computation power for their customers. These VMs are big targets for crypto currency mining malware.
- 2) The increase of remote working and globally dispersed workforce and application accessibility especially after the COVID 19 give the attackers more chances to hide their malicious traffic to compromise the cloud-hosted VMs, and use them for their malicious campaigns (phishing campaigns, botnet command, and control, so on).
- 3) The increase in IoT applications that use cloud-hosted infrastructure to analyze the enormous amounts of data generated by these applications to create business value and insights.

By considering this above scenario we decided to perform monitoring and analysis of data uploaded by user on cloud premises and how/she can mitigate the dangers if they are trapped in such circumstances. The main objectives of this project are as follows: -

- It aims to provide the best approaches to achieve continuous monitoring of malware attacks on the cloud along with their phases (before, during, and after).
- Logging and forensics techniques have always been the cornerstone of achieving continuous monitoring and detection of malware attacks on-premises.
- To adopt the best methods to bring loggings and forensics to the cloud and integrate them with on-premises visibility.
- Achieving the full monitoring over the whole security posture of the organization assets whether they are on-premises or on the cloud.

Below is the list of the tools and technologies which we have used in this project: -

- AWS CloudTrail for creating data log files.
- AWS CloudWatch for monitoring.

CHAPTER: 2 PROJECT SCOPE

CHAPTER 2 PROJECT SCOPE

The project is limited to only Desktop/Service system because data which is considered for malware analysis and monitoring must be uploaded by the user on cloud premises.

CHAPTER: 3 SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER 3 SOFTWARE AND HARDWARE REQUIREMENTS

Minimum Hardware Requirements

Processor	2.0 GHz
RAM	4GB
HDD	40GB

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

Operating System	Any operating system which can support an internet browser.
Programming language	-
Other tools & tech	AWS, Splunk, kali Linux

Table 3.2 Minimum Software Requirements

CHAPTER: 4 PROCESS MODEL

CHAPTER 4 PROCESS MODEL

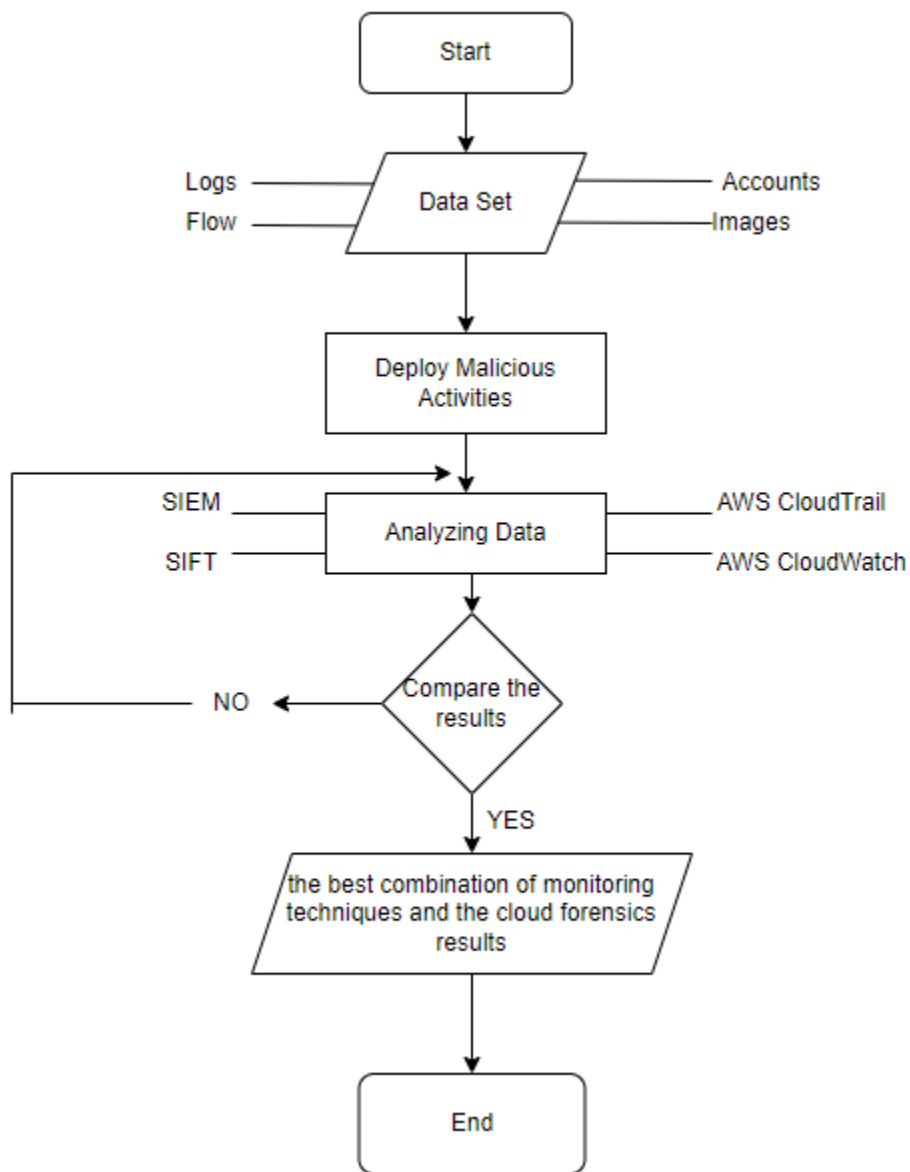


Figure 4.1 Process Model of Project

CHAPTER: 5 PROJECT PLAN

CHAPTER 5 PROJECT PLAN

5.1 List of Major Activities

5.1.1 Tasks for Implementing Data Monitoring in First Phase

- Task: - 1 Exploring NIST and MITRE ATT&CK Frameworks
- Task: - 2 Exploring AWS Tools (CloudTrail and CloudWatch) to generate data log files
- Task: - 3 Creating and uploading data files on Amazon S3 for Analysis
- Task: - 4 Malware Attack and Monitoring

5.1.2 Time Duration to Complete First Phase

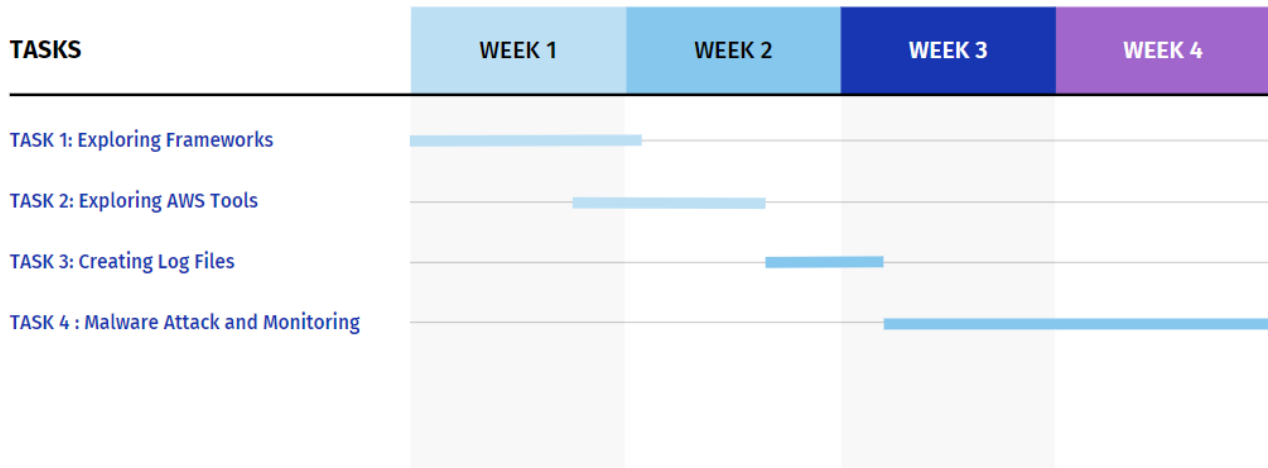


Figure 5.1 Task Completion Time Duration in First Phase

5.1.3 Tasks for Implementing Data Logging and Integration in Second Phase

- Task: - 1 Exploring AWS CloudTrail and Gathering Data Log Files
- Task: - 2 Implementing Data Monitoring and Logging on AWS Config
- Task: - 3 Exploring to SIEM Tools to transfer logs
- Task: - 4 Integrating Splunk with AWS CloudTrail Logs

5.1.4 Time Duration to Complete Second Phase

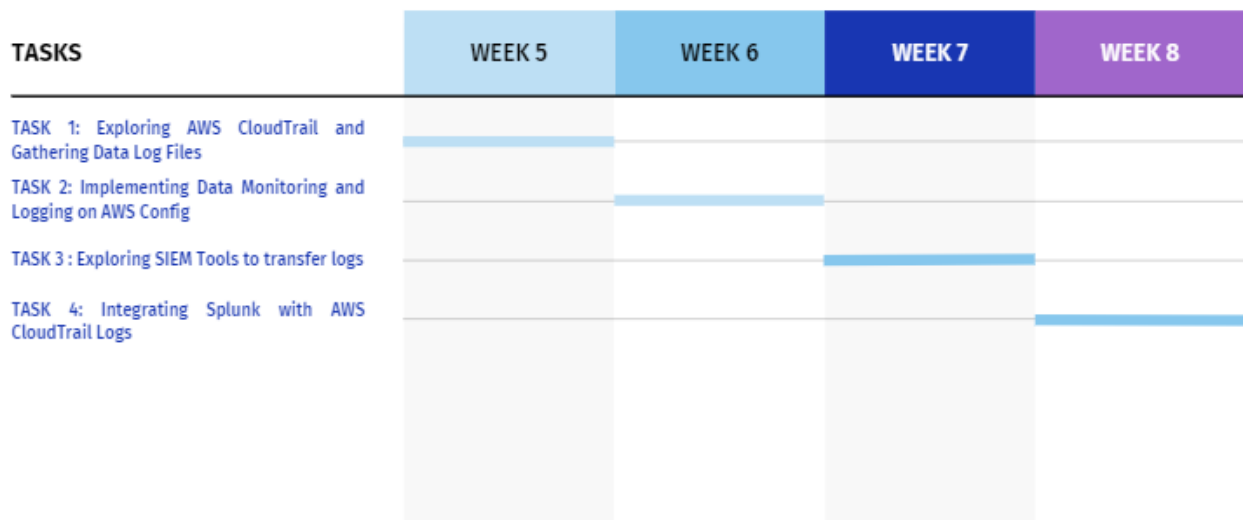


Figure 5.2 Task Completion Time Duration in Second Phase

CHAPTER: 6 IMPLEMENTATION DETAILS

CHAPTER 6 IMPLEMENTATION DETAILS

6.1 Background

The proposed project is based on 4 fundamental parts as follows: -

1. Infrastructure as a Service (IaaS) Cloud - It is the most fundamental and critical service, offering basic computing services such as servers, networking, and storage. This service enhances system availability while also lowering costs and offering a more flexible system.

2. Malware Attacks - Malware is a term that means malicious and harmful, it has similar effect on networks, software, operating systems, or other components. One of the biggest challenges in the IaaS cloud world is malware attacks; it is a major concern to home and business devices, as well as cloud virtual machines.

3. Malware Detection Methods – In order to prevent malware from hampering networks malware detections methods are necessary to implement in order for its proper functioning, a number of malware detection methods can be applied for e.g.: - Signature/Behavior based techniques for malware detection malware detection, Machine Learning Based malware detection methods etc.

4. Cloud Forensics - Cloud Digital Forensic techniques are typically used to gathering and preserving evidence, reconstructing incidents, deciding how, where, and where an incident happening, and producing threat information.

6.2 Methodology

The methodology has been divided into two practical parts:

The First: when the malware attack happened, make cloud analysis for malware detection.

The Second: is Forensics Analysis in the IaaS Cloud after the malware attack happens.

6.2.1 Gathering Data

Fortunately, there are community initiatives that define and classify each cloud attack technique publicly witnessed; such as the NIST Cybersecurity Framework and MITRE ATT&CK cloud framework.

For this project multiple csv files and data log files uploaded on NIST and MITRE ATT&CK website have been used for performing monitoring of data. Otherwise, any type of data can be used by a user as monitoring and analysis is done on cloud.

6.3 Cloud Analysis to Malware Detection

6.3.1. Test Environment - The tests were performed on Amazon Web services (AWS) hosted infrastructure. choosing the Amazon Web services (AWS) for this research because it the market leader for public cloud services offering and has a wide service catalog making it a suitable choice for most organizations.

6.3.2. Data Set – Any data can be considered by a user for testing this module, for the sake of testing we have selected data which provided by NIST and MITRE ATT&CK frameworks from their websites.

Continuous monitoring on IaaS can be accomplished by gathering and processing the following

- API calls Monitoring (In AWS it can be achieved through CloudTrail's logs).
- Host logs and logs of deployed Host Intrusion detection System (HIDs).
- VPC flows.
- Logs of the cloud resources (in AWS it's the CloudWatch Logs)
- Image and instance integrity validation.

6.3.3. Testing and Analysis – For performing testing and analysis multiple tools have been utilized to store data and perform malware attacks on it

AWS CloudWatch - Collect and track metrics, collect and monitor log files, set alarms, and automatically react to changes.

AWS CloudTrail - A web service that logs your account's AWS API calls and provides you log files.

AWS S3 – For storing data and hosting a static website

Kali Linux – For performing malware attacks

6.3.4 Testing Phases

1. Creating AWS Billing Alarm

According to the MITRE ATT&CK framework for cloud attacks, one of the most used attack vectors for Cloud attacks and malware attacks targeting cloud-hosted environments is cloud account takeover. There are many ways to detect cloud account takeover, one of the best ways is detecting changes in the usual billing on AWS. Most public cloud providers provide features to enable their customers to create billing and send them emails when these alarms are triggered

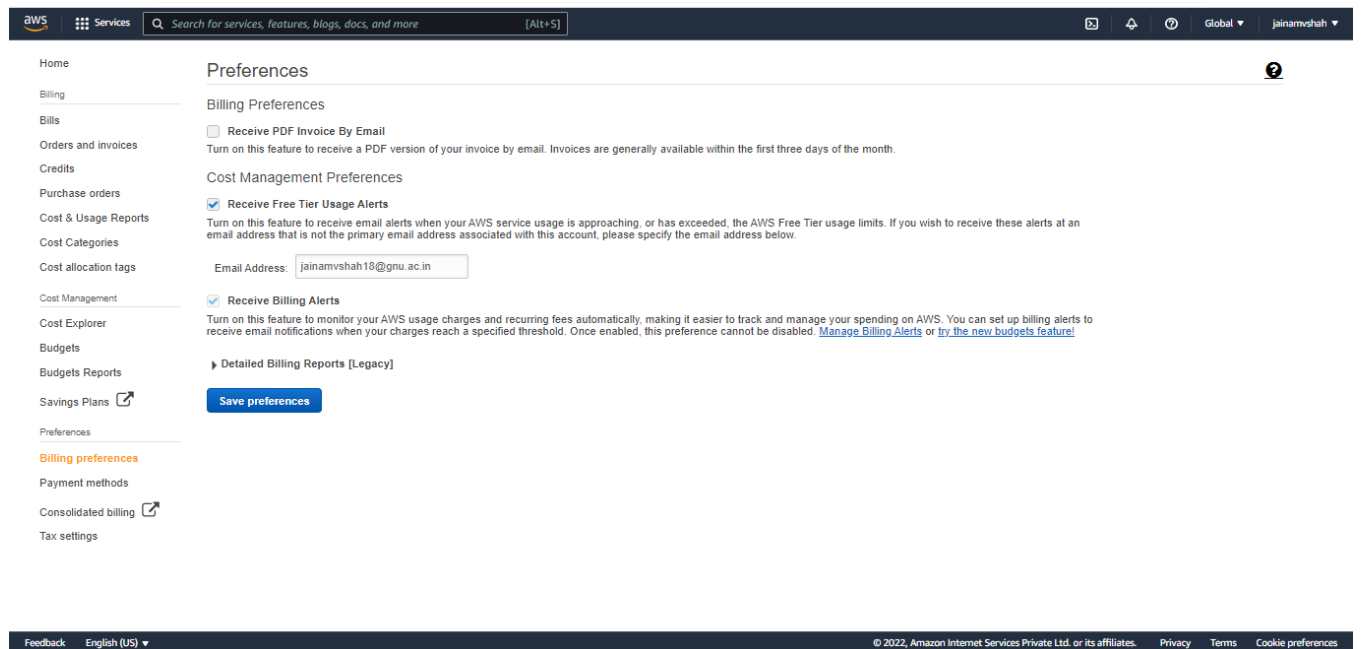


Figure 6.1 AWS Billing Preferences

If there is usage of any service on the respective cloud account AWS will send notification to the respective email.

2. Performing Continuous Monitoring in AWS Environment

AWS offers a service called AWS Config, this service allows monitoring AWS resource configurations and track resource inventory and changes, which can be used to detect any malicious configuration changes the attacker tries to make to gain control or persistence over the compromised account's resources. This monitoring feeds then can be consumed using AWS CloudWatch and SNS Notifications can be created based on them.

Malware attacks target and modify the data stored and any misconfigured cloud storage leading to leaked data. By using AWS Config to make many rules like sure storage versioning is enabled for AWS storage (S3). By enabling the s3-bucketversioning- enabled rule, another action performed by attackers is to try to hide their malicious API calls by disabling API calls monitoring, configured a rule to detect if CloudTrail enabled or not and another rule to detect whether the volumes used are encrypted or not.

Initially starting by making S3 buckets in respective AWS account in order to perform monitoring and also to do malware attacks

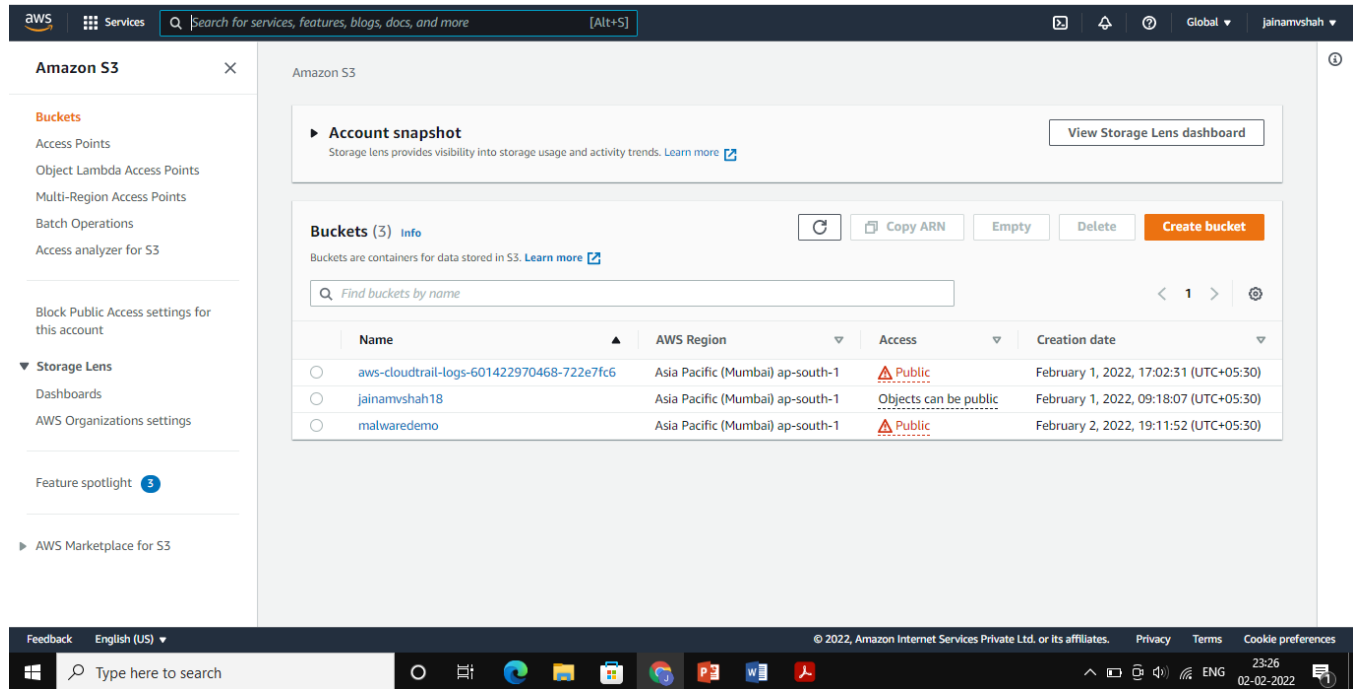


Figure 6.2 S3 Buckets

After that additional charts are being created for request and storage metrics in order to perform monitoring on our respective bucket

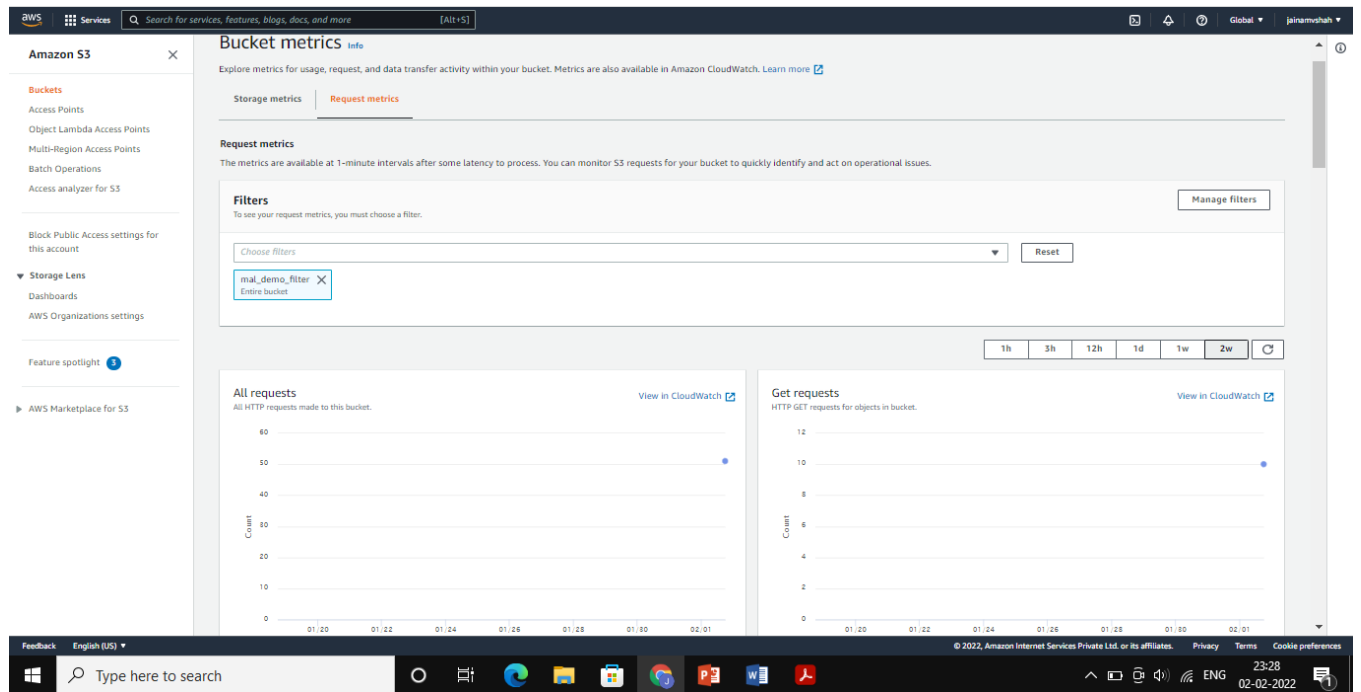


Figure 6.3 Creating Metrics for bucket

In order to monitor activities taking place within the S3 bucket like uploading or downloading files by a user or any malicious activities taking place without the awareness of the respective user AWS CloudWatch comes into place. An alarm configured on CloudWatch helps a user to track and monitor the S3 bucket in an efficient manner. An alarm for the respective bucket is created in the following manner.

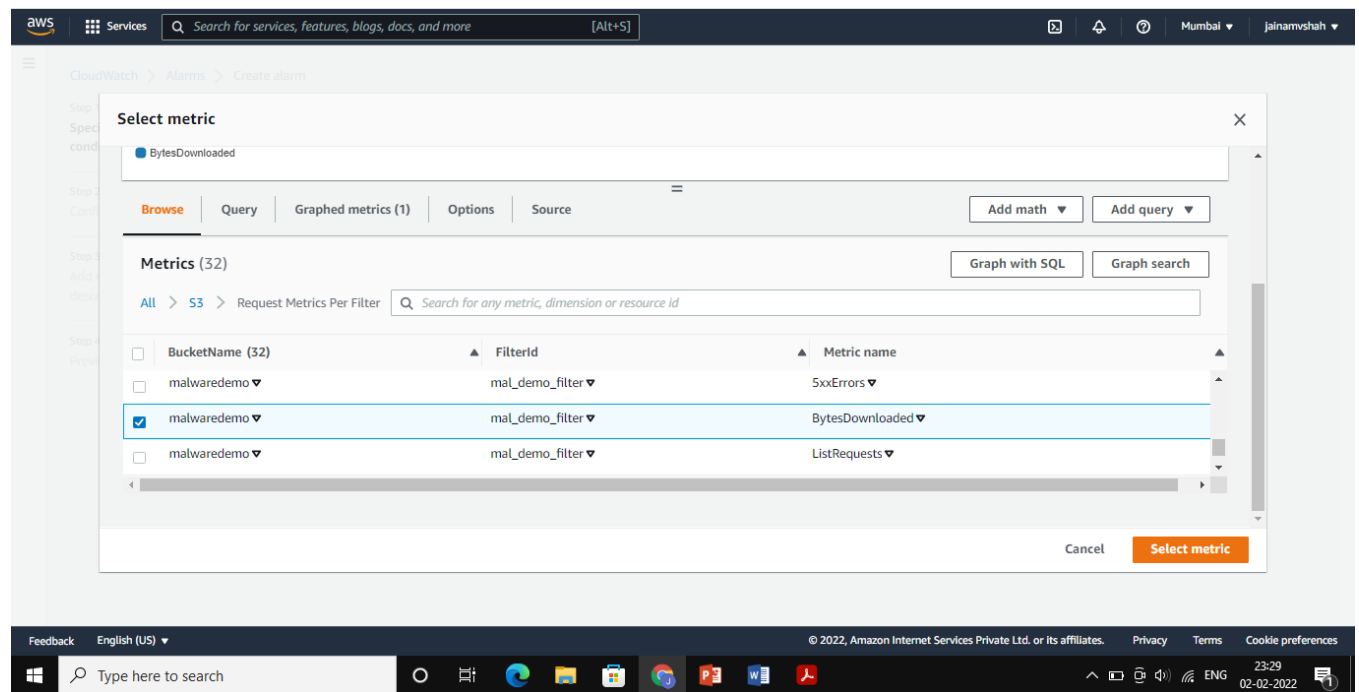


Figure 6.4 Setting up Alarm

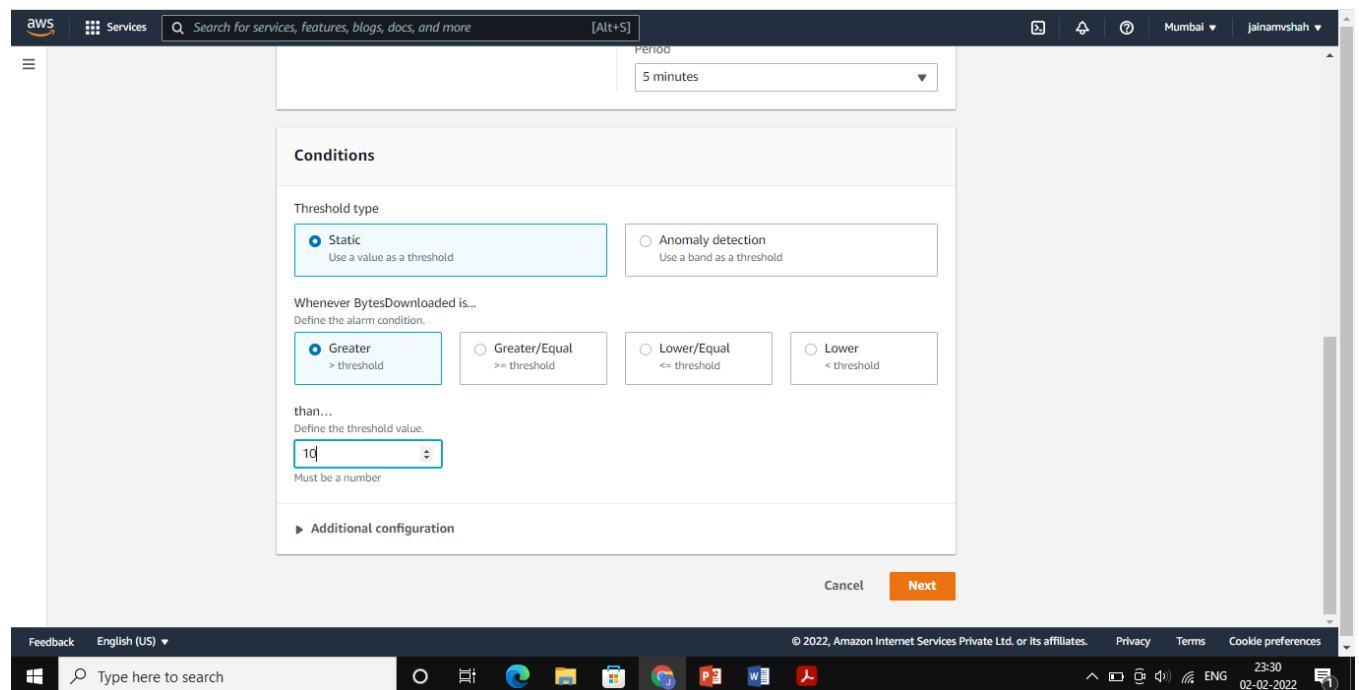
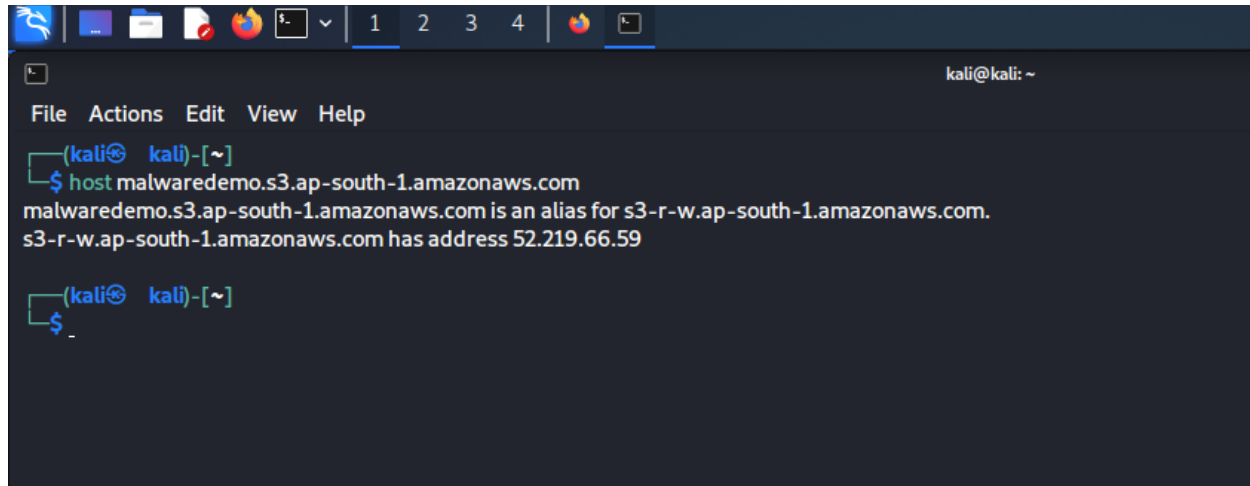


Figure 6.5 Defining threshold value for a definite amount of size

3. Performing a Malware Attack

After creating S3 bucket a malware attack has been initiated on the created S3 bucket using its respective URL. Following steps are performed in order to complete a malware attack on S3 Bucket

Step 1: First we identified the IP Address of this bucket URL using the following command

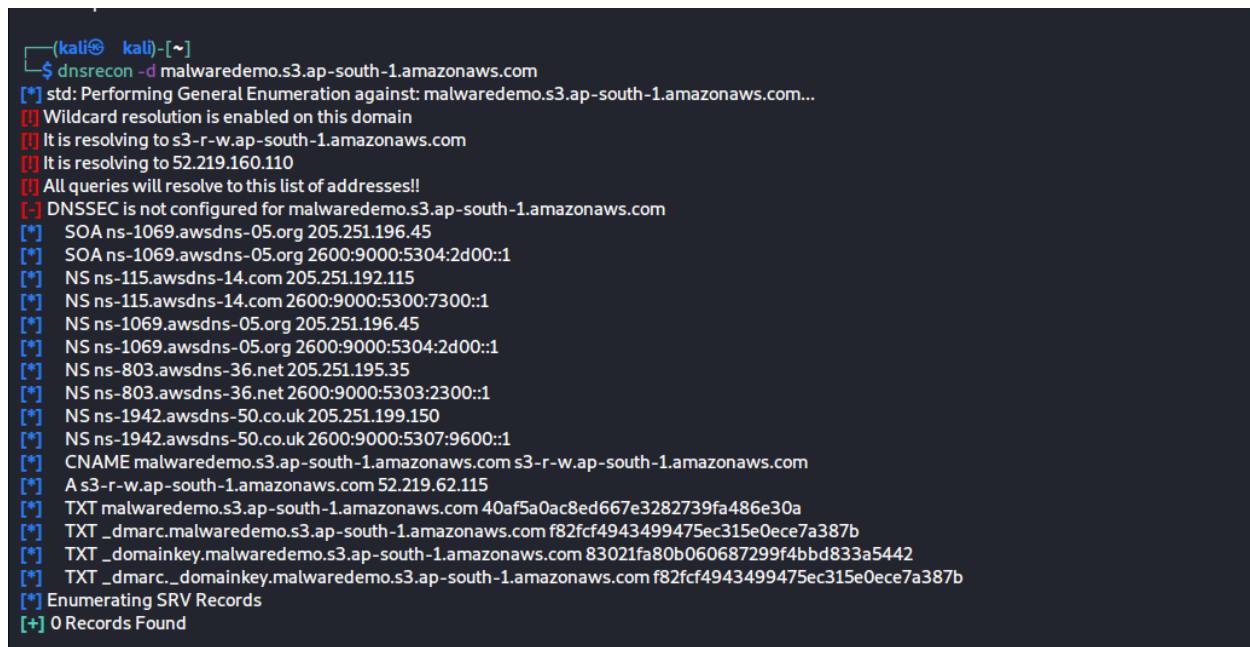


```
(kali㉿ kali)-[~]
$ host malwaredemo.s3.ap-south-1.amazonaws.com
malwaredemo.s3.ap-south-1.amazonaws.com is an alias for s3-r-w.ap-south-1.amazonaws.com.
s3-r-w.ap-south-1.amazonaws.com has address 52.219.66.59

(kali㉿ kali)-[~]
$ _
```

Figure 6.6 Identifying IP Address

Step 2: DNS attack on bucket URL to know number of servers through which that URL request passed



```
(kali㉿ kali)-[~]
$ dnsrecon -d malwaredemo.s3.ap-south-1.amazonaws.com
[*] std: Performing General Enumeration against: malwaredemo.s3.ap-south-1.amazonaws.com...
[!] Wildcard resolution is enabled on this domain
[!] It is resolving to s3-r-w.ap-south-1.amazonaws.com
[!] It is resolving to 52.219.160.110
[!] All queries will resolve to this list of addresses!!
[-] DNSSEC is not configured for malwaredemo.s3.ap-south-1.amazonaws.com
[*] SOA ns-1069.awsdns-05.org 205.251.196.45
[*] SOA ns-1069.awsdns-05.org 2600:9000:5304:2d00::1
[*] NS ns-115.awsdns-14.com 205.251.192.115
[*] NS ns-115.awsdns-14.com 2600:9000:5300:7300::1
[*] NS ns-1069.awsdns-05.org 205.251.196.45
[*] NS ns-1069.awsdns-05.org 2600:9000:5304:2d00::1
[*] NS ns-803.awsdns-36.net 205.251.195.35
[*] NS ns-803.awsdns-36.net 2600:9000:5303:2300::1
[*] NS ns-1942.awsdns-50.co.uk 205.251.199.150
[*] NS ns-1942.awsdns-50.co.uk 2600:9000:5307:9600::1
[*] CNAME malwaredemo.s3.ap-south-1.amazonaws.com s3-r-w.ap-south-1.amazonaws.com
[*] A s3-r-w.ap-south-1.amazonaws.com 52.219.62.115
[*] TXT malwaredemo.s3.ap-south-1.amazonaws.com 40af5a0ac8ed667e3282739fa486e30a
[*] TXT _dmarc.malwaredemo.s3.ap-south-1.amazonaws.com f82fcf4943499475ec315e0ece7a387b
[*] TXT _domainkey.malwaredemo.s3.ap-south-1.amazonaws.com 83021fa80b060687299f4bbd833a5442
[*] TXT _dmarc._domainkey.malwaredemo.s3.ap-south-1.amazonaws.com f82fcf4943499475ec315e0ece7a387b
[*] Enumerating SRV Records
[+] 0 Records Found
```

Figure 6.7 Initiating DNS Attack on the respective bucket

Step 3: Here we try to fetch the actual name of bucket URL.

```
(kali) [~]
$ nslookup 52.219.66.59 1 x

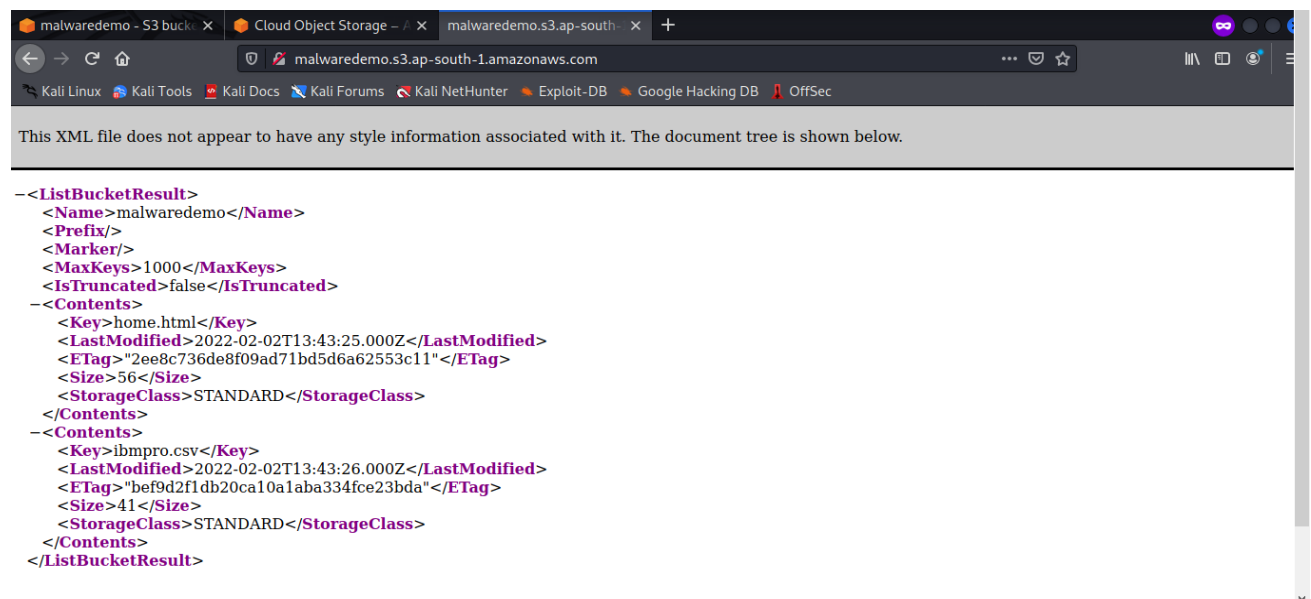
59.66.219.52.in-addr.arpa    name = s3-r-w.ap-south-1.amazonaws.com.

Authoritative answers can be found from:

(kali) [~]
$ _
```

Figure 6.8 Name Revealing of S3 Bucket

We paste the acquired name in the browser to see the tree structure of files in the respective bucket. It is in XML format.



```
-<ListBucketResult>
  <Name>malwaredemo</Name>
  <Prefix/>
  <Marker/>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>>false</IsTruncated>
  -<Contents>
    <Key>home.html</Key>
    <LastModified>2022-02-02T13:43:25.000Z</LastModified>
    <ETag>"2ee8c736de8f09ad71bd5d6a62553c11"</ETag>
    <Size>56</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  -<Contents>
    <Key>ibmpro.csv</Key>
    <LastModified>2022-02-02T13:43:26.000Z</LastModified>
    <ETag>"bef9d2f1db20ca10a1aba334fce23bda"</ETag>
    <Size>41</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
</ListBucketResult>
```

Figure 6.9 Tree structure of files present in the bucket

Step 4: Using the following command we get the list of files present in the bucket which do not require authentication to access it.

```
(kali) kali-[~]
$ aws s3 ls s3://malwaredemo --no-sign-request
2022-02-02 08:43:25    56 home.html
2022-02-02 08:43:26    41 ibmpro.csv

(kali) kali-[~]
$ aws s3 ls s3://malwaredemo --no-sign-request_
```

Figure 6.10 Revealing files not needing authentication to access

Here we tried to open the listed files in browser

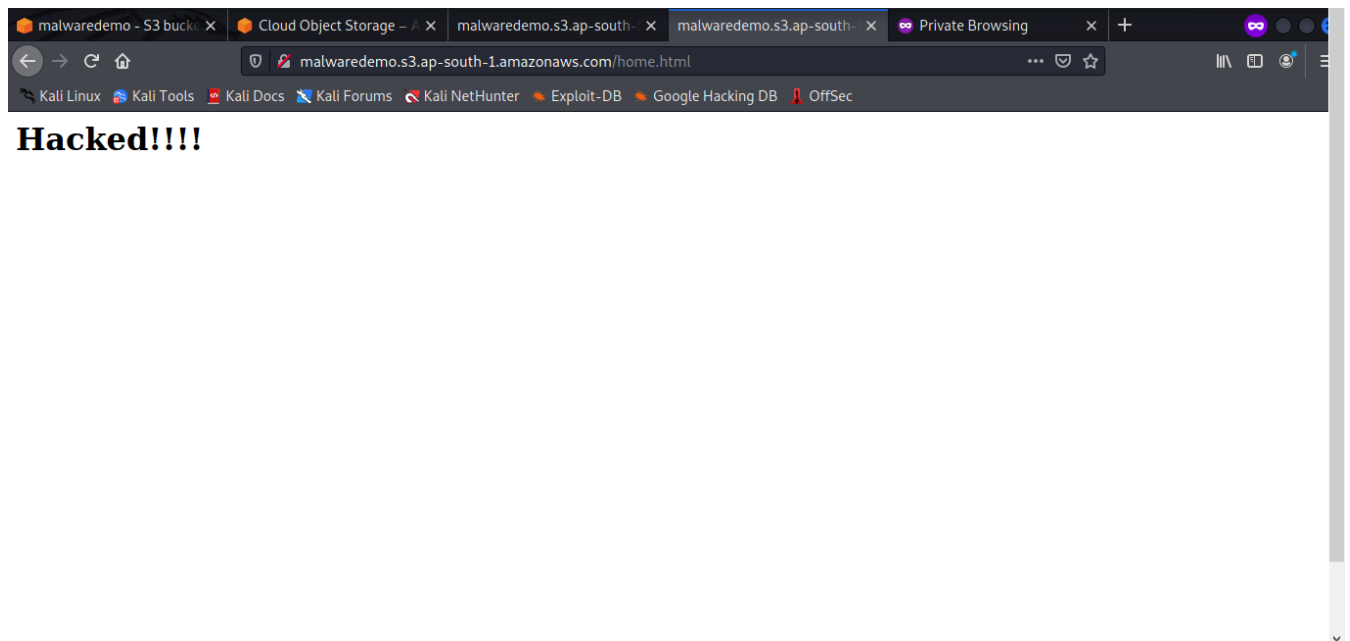


Figure 6.11 Home.html file

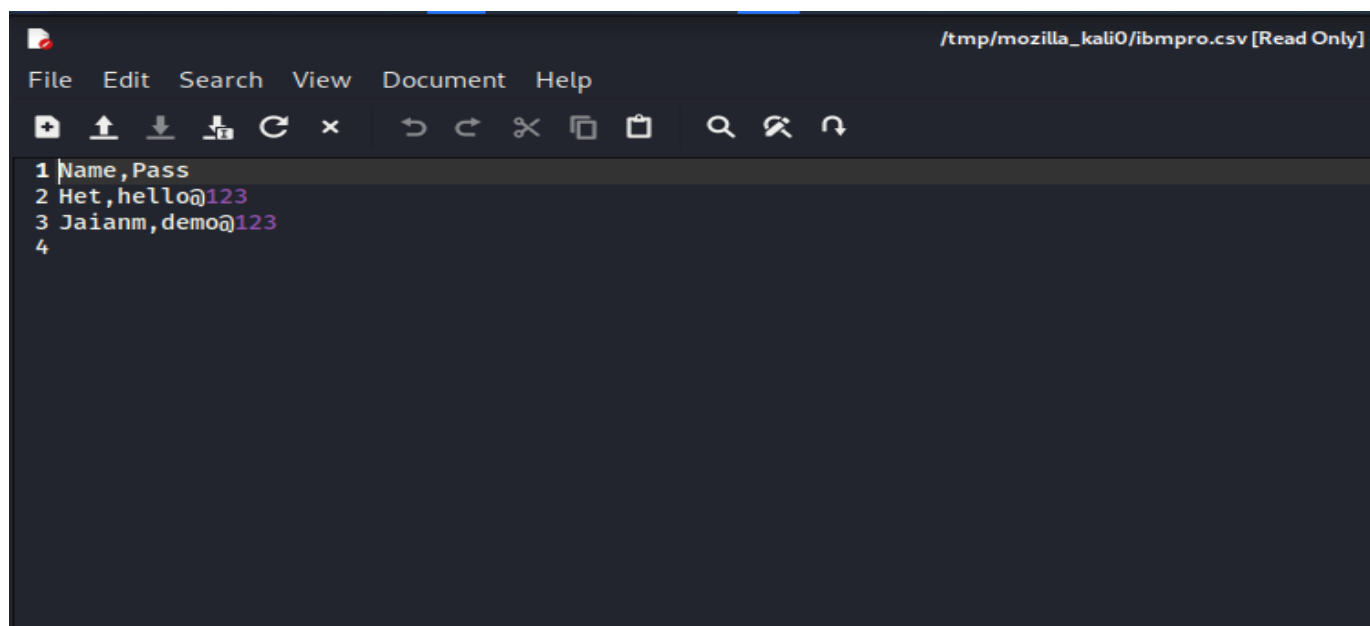


Figure 6.12 ibmpro.csv

Step 5: Then S3Scanner python file is used to find the S3 bucket data and dump its content to the local machine. Here this following command has been used to scan whether bucket is present or not and also lists out AuthUsers and AllUsers permissions

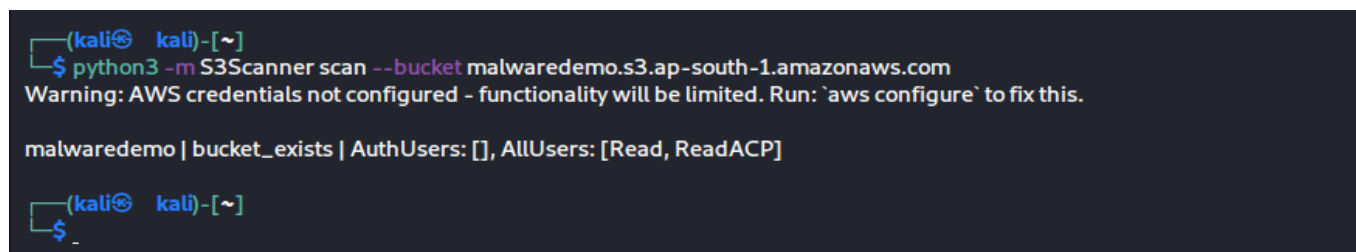


Figure 6.13 Listing Users

Step 6: The following command is used to dump all content from bucket to local machine at any location.

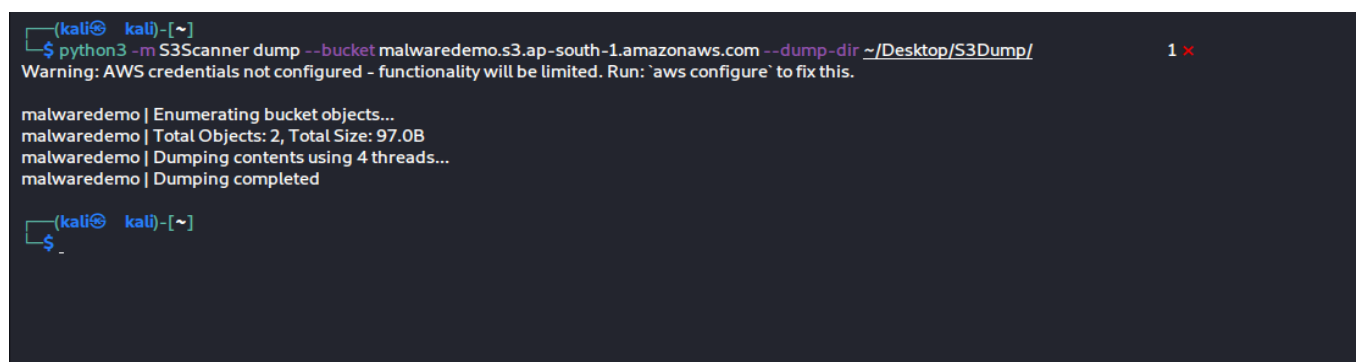


Figure 6.14 Dump status

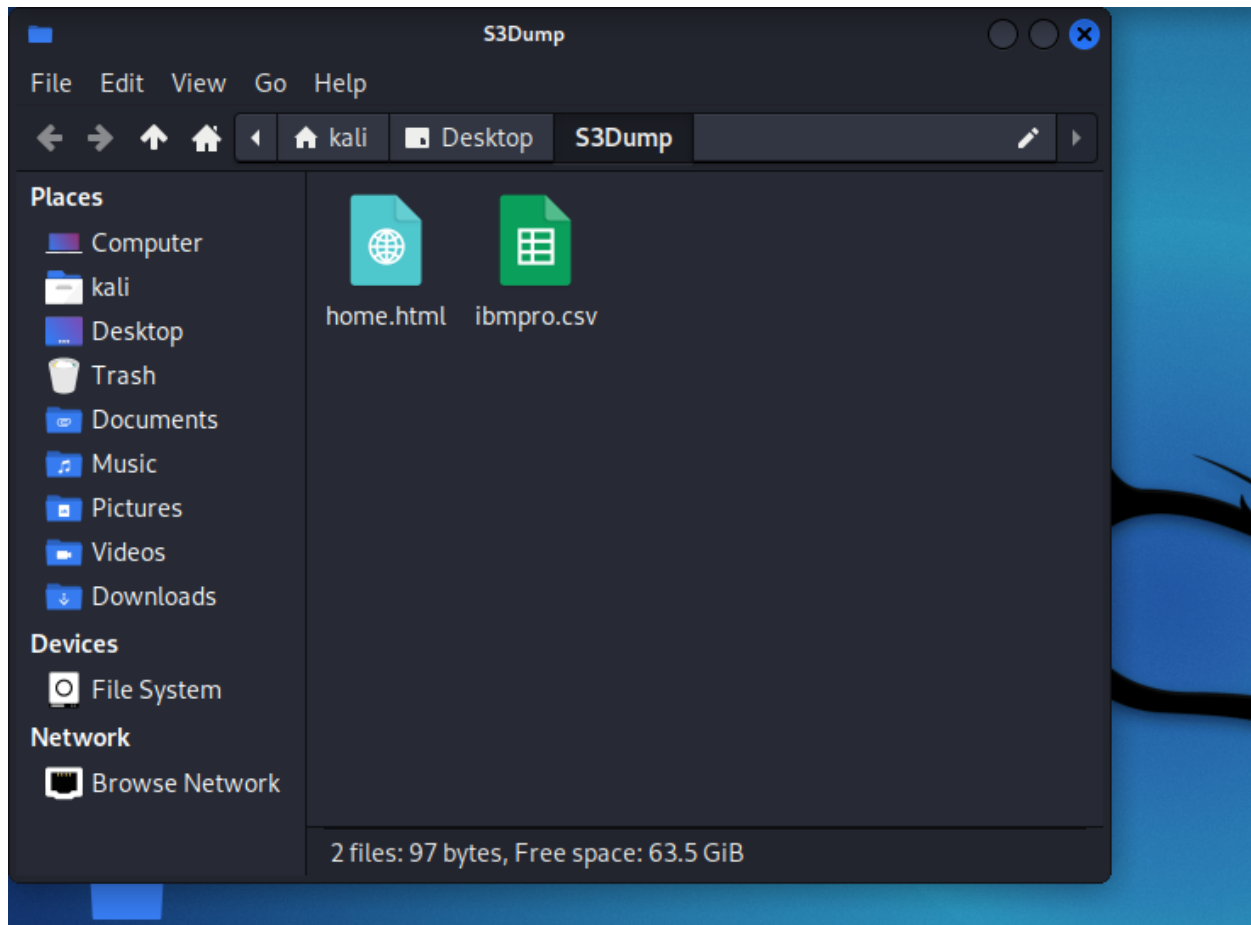


Figure 6.15 Files downloaded on local machine

As we can see malware alarm created earlier to monitor S3 bucket has been triggered based on intrusion being detected and we can see the size of files being downloaded from the bucket respectively.

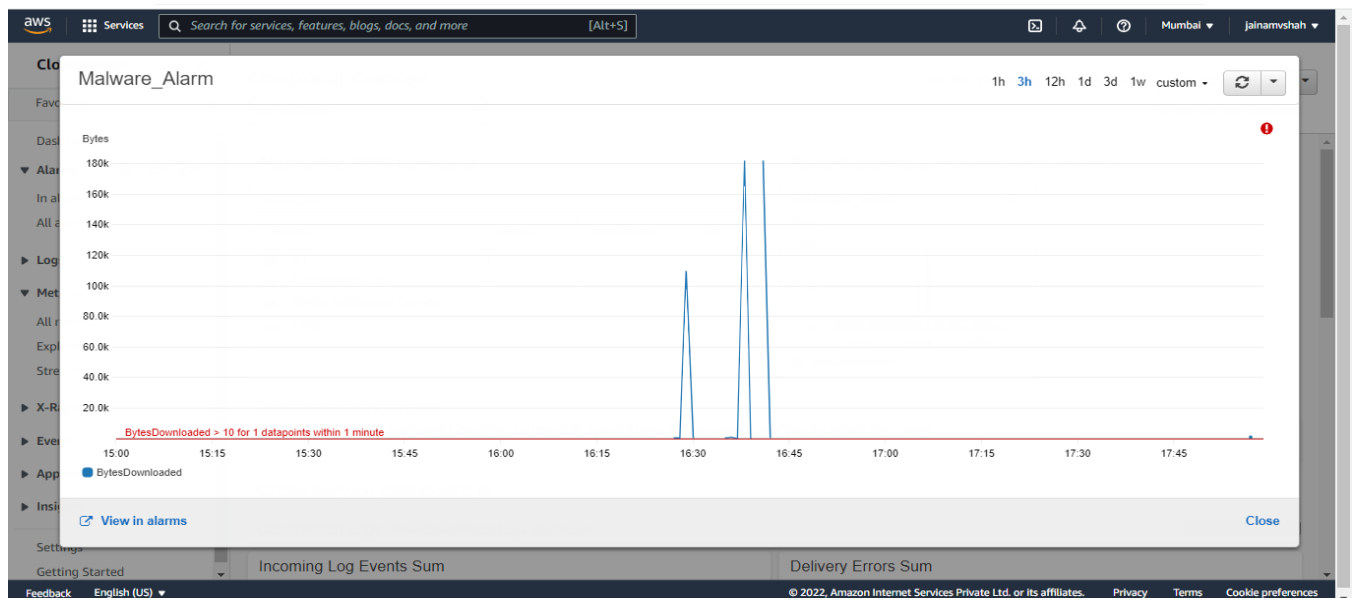


Figure 6.16 Malware Alarm

4. Implementing Data Monitoring using SNS

Amazon Simple Notification Service (Amazon SNS) is a fully managed messaging service for both application-to-application (A2A) and application-to-person (A2P) communication.

Through SNS we can monitor AWS services and it provides notification whenever there is a change happening in any respective service within respective AWS account.

For the purpose of this project an SNS Topic is created as follows on AWS CloudTrail to monitor data logging taking place within S3 Bucket.

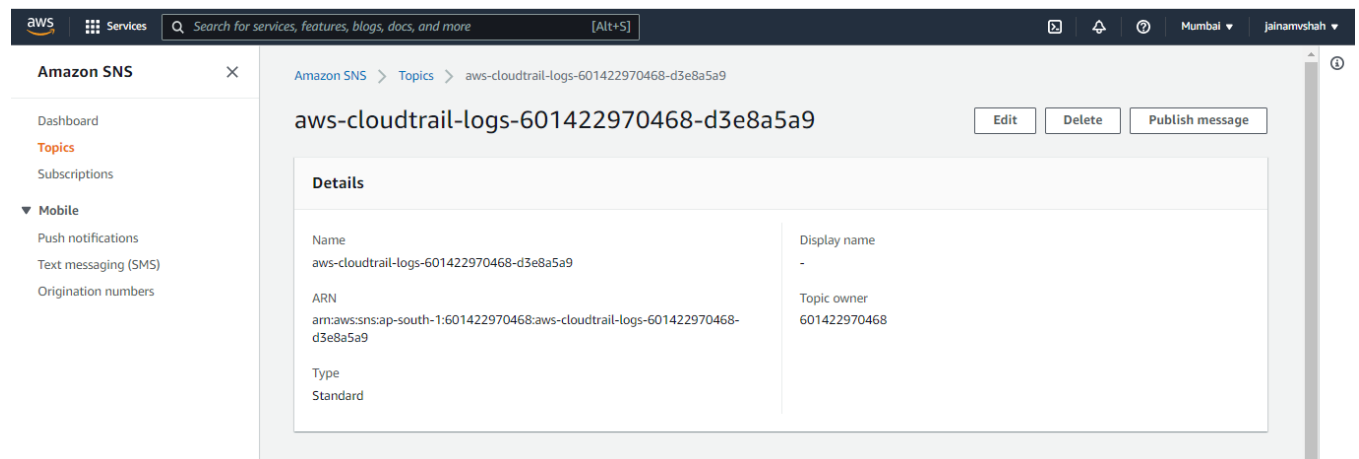


Figure 6.17 SNS Topic

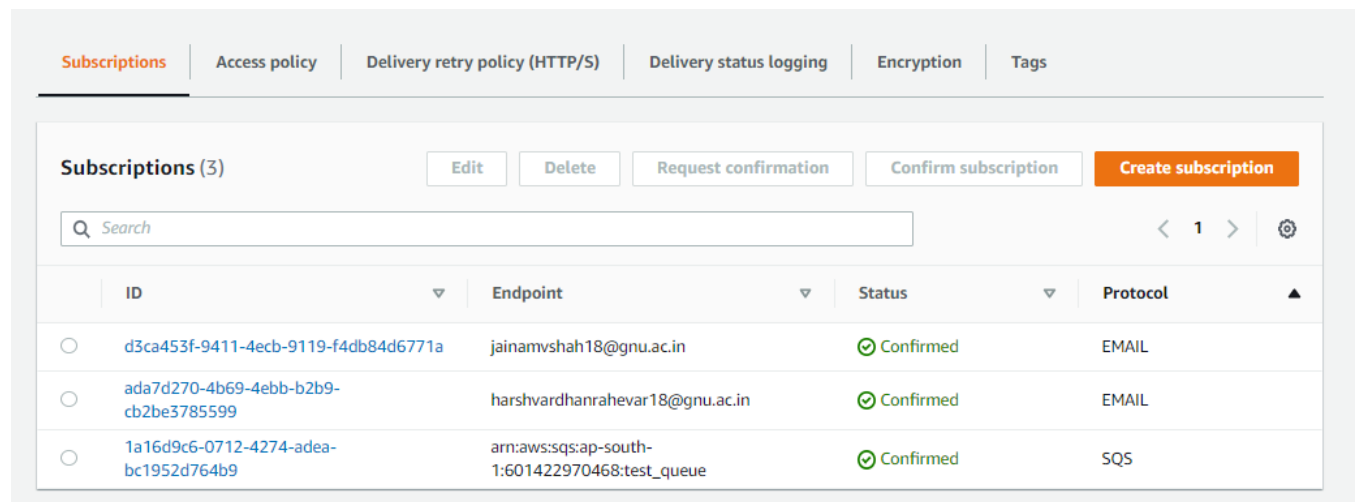


Figure 6.18 SNS Notification Subscriptions

Subscription: ada7d270-4b69-4ebb-b2b9-cb2be3785599

EditDelete

Details

ARN arn:aws:sns:ap-south-1:601422970468:aws-cloudtrail-logs-601422970468-d3e8a5a9:ada7d270-4b69-4ebb-b2b9-cb2be3785599	Status Confirmed
Endpoint harshvardhanrahevar18@gnu.ac.in	Protocol EMAIL
Topic aws-cloudtrail-logs-601422970468-d3e8a5a9	

Figure 6.19.1 Subscription details of 1st Endpoint

Subscription: d3ca453f-9411-4ecb-9119-f4db84d6771a

EditDelete

Details

ARN arn:aws:sns:ap-south-1:601422970468:aws-cloudtrail-logs-601422970468-d3e8a5a9:d3ca453f-9411-4ecb-9119-f4db84d6771a	Status Confirmed
Endpoint jainamvshah18@gnu.ac.in	Protocol EMAIL
Topic aws-cloudtrail-logs-601422970468-d3e8a5a9	

Figure 6.19.2 Subscription details of 2nd Endpoint

In order to subscribe properly with the created trail from AWS CloudTrail the following script is written for S3 bucket.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSCloudTrailAclCheck20150319",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudtrail.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::demo211"
    },
    {
      "Sid": "AWSCloudTrailWrite20150319",
      "Effect": "Allow",
      "Principal": {
```

```

    "Service": "cloudtrail.amazonaws.com"
  },
  "Action": "s3:PutObject",
  "Resource": "arn:aws:s3:::demo211/AWSLogs/601422970468/*",
  "Condition": {
    "StringEquals": {
      "AWS:SourceArn": "arn:aws:cloudtrail:ap-south-1:601422970468:trail/demo2.1",
      "s3:x-amz-acl": "bucket-owner-full-control"
    }
  }
}
]
}

```

The above-mentioned script validates the destination configuration in accordance to API Response and provides SNS Notification to the respective subscription without any delay and in a timely manner.

6.3.5 Generating data logs from AWS CloudTrail

In AWS, the CloudTrail service is used to monitor account activity and API calls, this is a very important feature as cloud providers offer their services via APIs. CloudTrail feeds can also be integrated with CloudWatch to create metrics generating alarms for any suspicious account's behavior or any account misuse.

For fulfilling the purpose of generating data logs, in CloudTrail ongoing delivery of events is enabled as log files to an Amazon S3 bucket. Then the logs are and API Calls are received from CloudTrail Event history. For the sake of monitoring the activity on S3 service a trail is created on an existing S3 bucket and SNS subscription can also be enabled to keep track of how many logs and events are generated every hour in a S3 bucket.

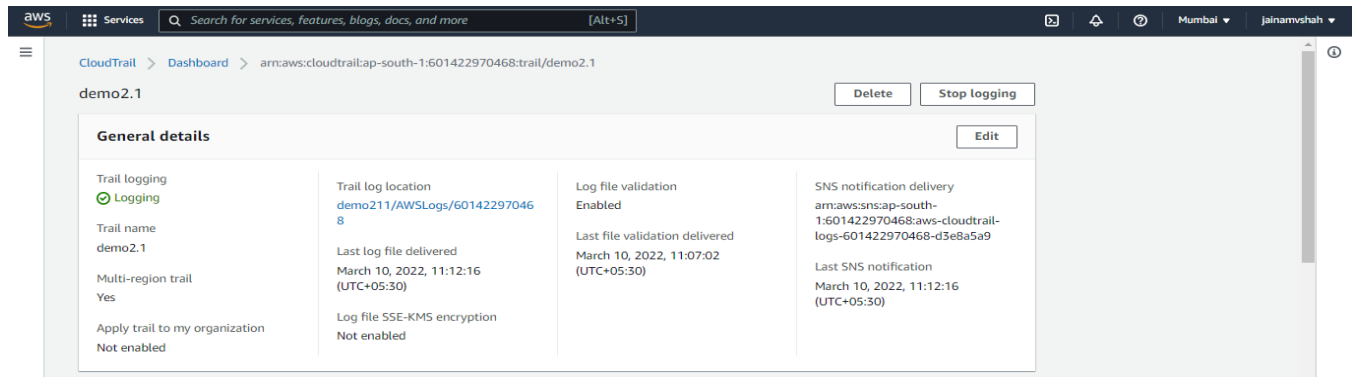


Figure 6.20 CloudTrail Details

After generating the trail, a folder is created within the S3 bucket called AWSLogs/ which stores all the log files containing activities within S3 in json.gz format.

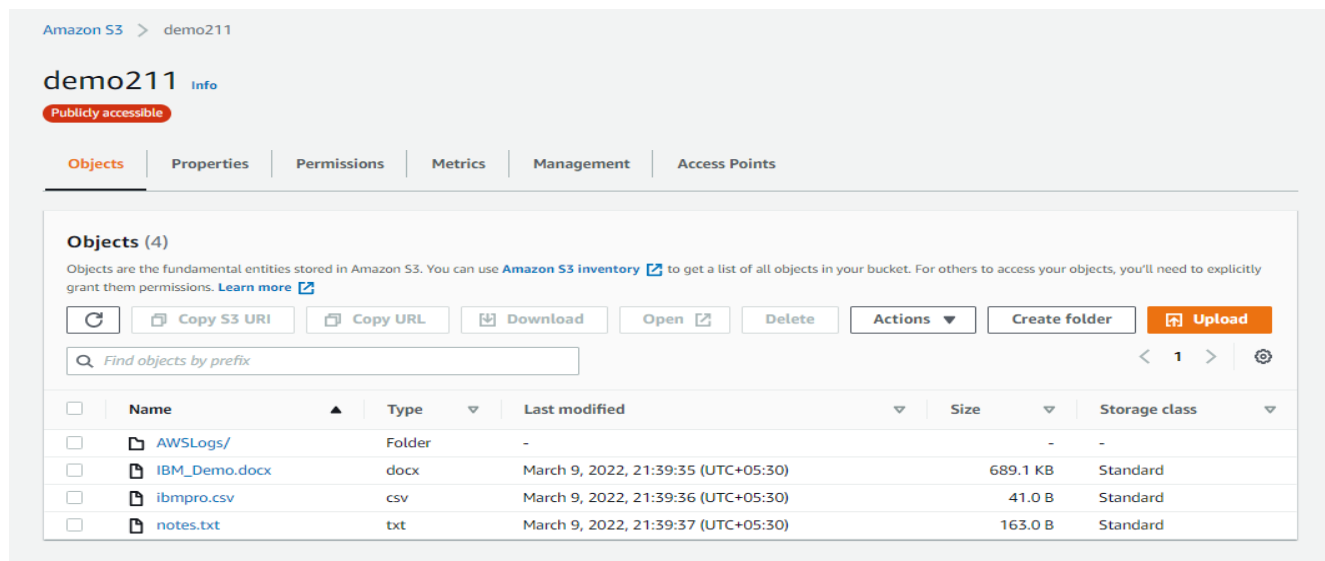


Figure 6.21 S3 Bucket with CloudTrail Logs

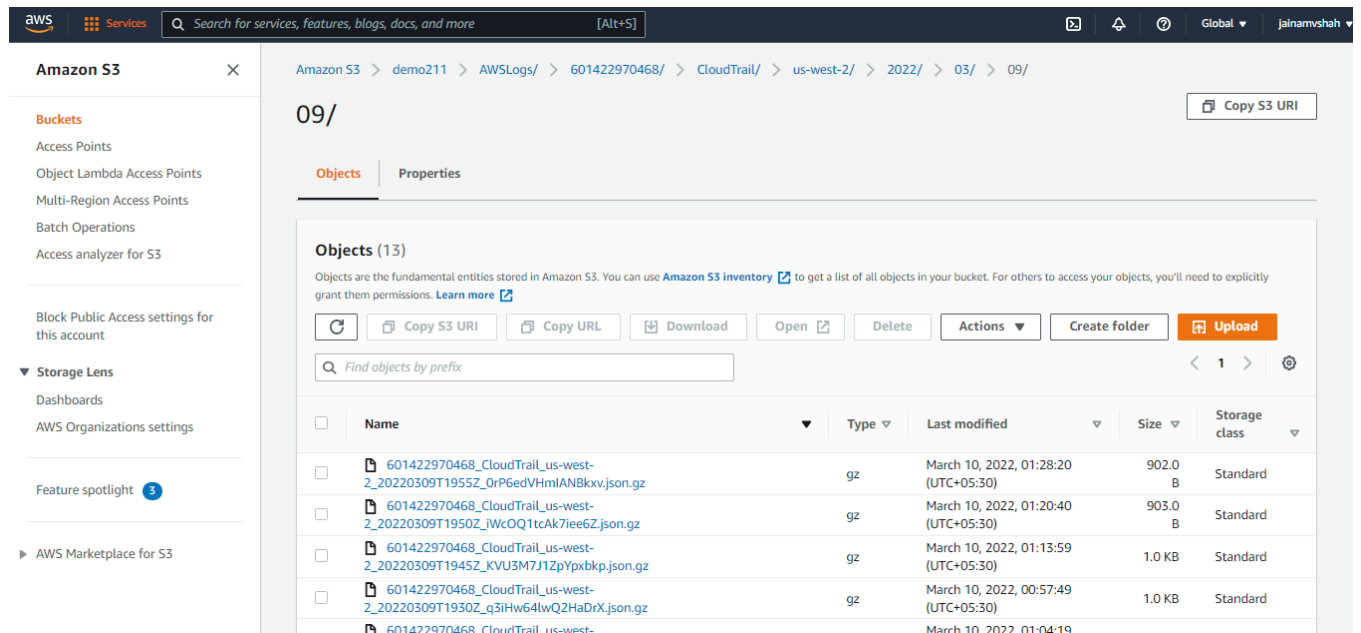


Figure 6.22 Data Log Files

CloudTrail provides detailed log data. It provides the following results: -

- Detect console logins from suspected places or countries.
- Because most of the time, organizations transfer cloud logs to their own data center for long term storage and correlation with other on-premises tools, it's very important to generate the logs in a text-like format such as JSON, thus enabling serialization of complex, high quality log data and decouple the interpretation of logs from specific solution or vendor. From a test, notice AWS uses this concept in their generated logs and flows.

- Leverage the storage API (s3 API) to import cloud trails to a search and indexing platform or security management systems like (SIEM solution) for building more unified and robust use cases monitoring the security posture over the entire environment.

6.3.6 Integrating AWS CloudTrail with Splunk

In order to have clear understanding of the logs and perform proper forensic analysis there is a must need of bringing cloud logs into a single point where they can be aggregated with on premises security events and other security and intelligence feeds, thus enabling the threat management team to have a single pane of glass from which they can monitor the whole security posture of their organization.

To achieve this purpose Splunk has been utilized and configured to receive the AWS CloudTrail data logs which configured earlier to monitor different services of AWS account and its resources.

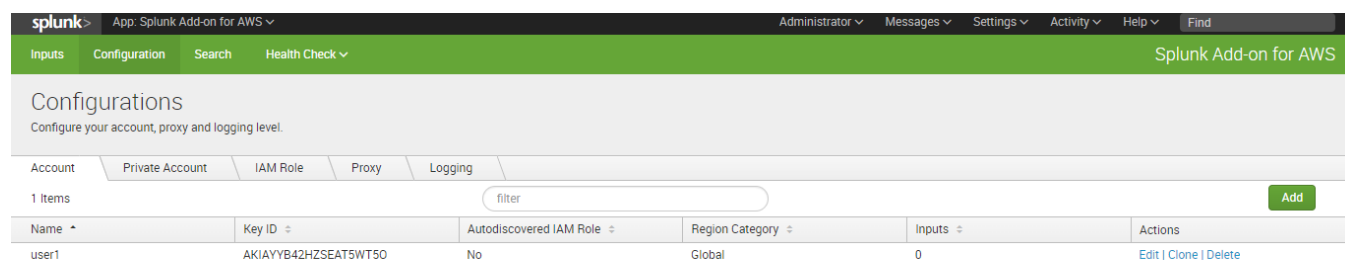


Figure 6.23 AWS Account Connection

First AWS Root account is connected and then input of CloudTrail is integrated with Splunk so the logs generated by AWS CloudTrail can be tracked from Splunk as follows: -

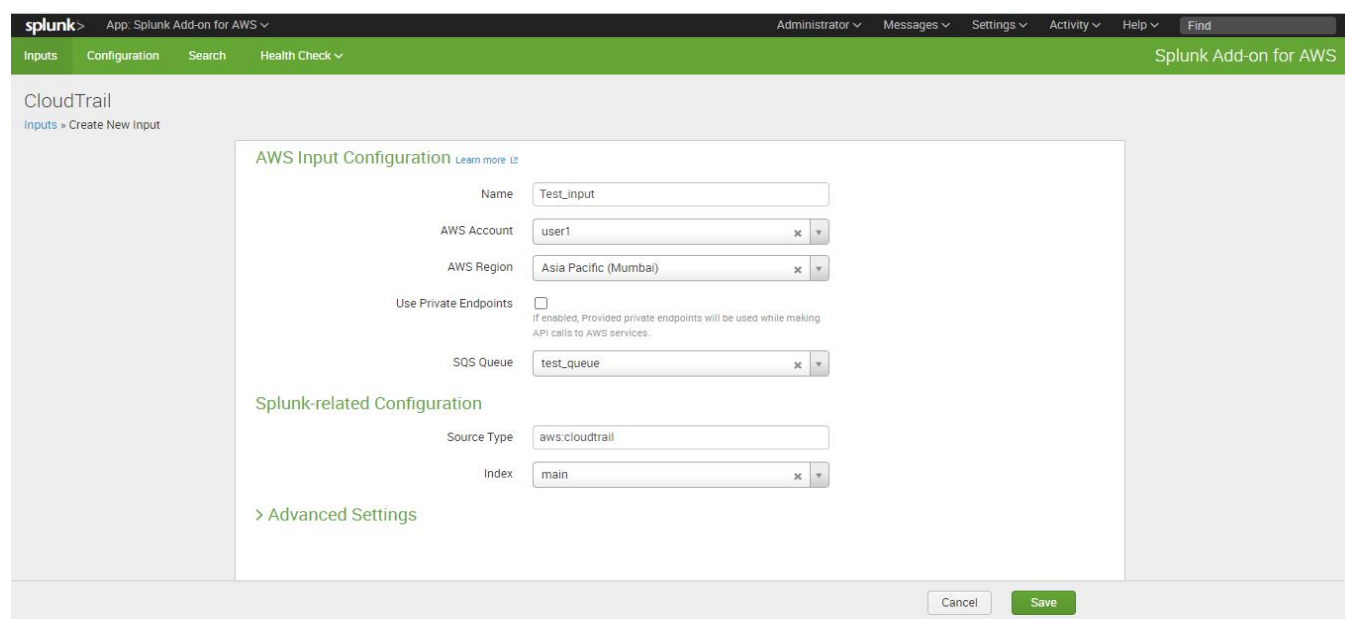


Figure 6.24 Source Input for Splunk

Figure 6.21 shows an input will be created for AWS CloudTrail to manage the logs generated by it and provide detailed information and analysis based on the fields selected in the IAM policy for Splunk Add-on.

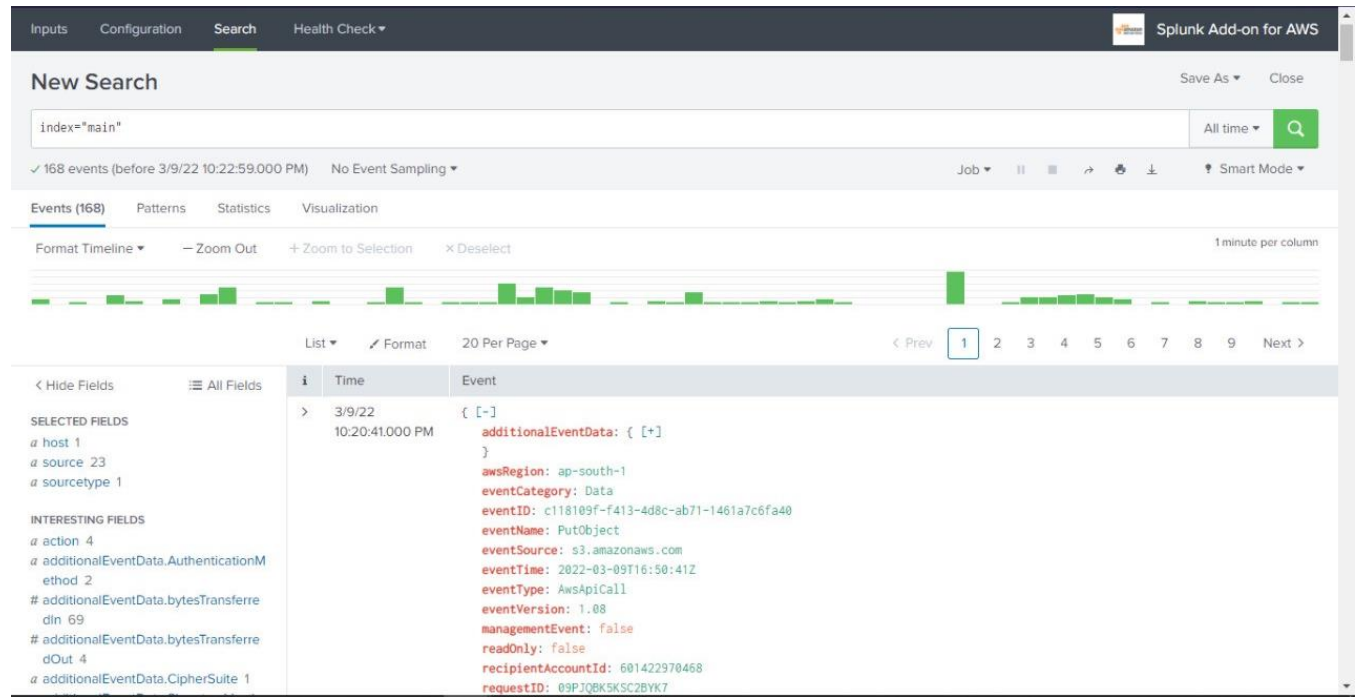


Figure 6.25 CloudTrail Logs Analysis

In order to provide the analysis based on the logs generated in the respective fields an IAM Policy called “Splunk Add-On” was created earlier by building a JSON script and integrated with respective AWS CloudTrail Trail and AWS account, the script is as follows: -

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "config:GetComplianceSummaryByConfigRule",
        "sqs:DeleteMessage",
        "iam:GetAccountPasswordPolicy",
        "s3:ListAccessPointsForObjectLambda",
        "ec2:DescribeInstances",
        "sqs:ReceiveMessage",
        "s3:DeleteAccessPoint",
        "ec2:DescribeSnapshots",
        "s3:DeleteAccessPointForObjectLambda",
        "ec2:DescribeVolumes",
        "s3:PutLifecycleConfiguration",
        "config:DescribeConfigRules",
        "ec2:DescribeKeyPairs",
```

"s3:DeleteObject",
"s3:CreateMultiRegionAccessPoint",
"lambda:ListFunctions",
"s3:GetBucketWebsite",
"s3:GetMultiRegionAccessPoint",
"s3:PutReplicationConfiguration",
"s3:GetObjectAttributes",
"sqs:SendMessage",
"s3:InitiateReplication",
"s3:GetObjectLegalHold",
"s3:GetBucketNotification",
"s3:GetReplicationConfiguration",
"s3:DescribeMultiRegionAccessPointOperation",
"s3:PutObject",
"s3:PutBucketNotification",
"s3:CreateJob",
"s3:PutBucketObjectLockConfiguration",
"ec2:DescribeSubnets",
"s3:GetStorageLensDashboard",
"s3:GetLifecycleConfiguration",
"s3:GetBucketTagging",
"s3:GetInventoryConfiguration",
"s3:GetAccessPointPolicyForObjectLambda",
"ec2:DescribeRegions",
"cloudtrail:*",
"s3:ListBucket",
"config:GetComplianceDetailsByConfigRule",
"s3:AbortMultipartUpload",
"rds:DescribeDBInstances",
"s3:UpdateJobPriority",
"s3:DeleteBucket",
"s3:PutBucketVersioning",
"iam:ListAccessKeys",
"s3:GetMultiRegionAccessPointPolicyStatus",
"s3:ListBucketMultipartUploads",
"config:DescribeConfigRuleEvaluationStatus",
"s3:PutIntelligentTieringConfiguration",
"s3:PutMetricsConfiguration",
"s3:GetBucketVersioning",
"s3:GetAccessPointConfigurationForObjectLambda",
"ec2:DescribeSecurityGroups",
"s3:PutInventoryConfiguration",
"s3:GetStorageLensConfiguration",
"s3:DeleteStorageLensConfiguration",
"s3:GetAccountPublicAccessBlock",
"s3:PutBucketWebsite",
"s3:ListAllMyBuckets",
"s3:PutBucketRequestPayment",
"s3:PutObjectRetention",
"ec2:DescribeVpcs",

"s3:CreateAccessPointForObjectLambda",
"s3:GetBucketCORS",
"iam:GetUser",
"s3:GetObjectVersion",
"s3:PutAnalyticsConfiguration",
"s3:PutAccessPointConfigurationForObjectLambda",
"s3:GetObjectVersionTagging",
"s3:PutStorageLensConfiguration",
"s3:CreateBucket",
"s3:GetStorageLensConfigurationTagging",
"s3:ReplicateObject",
"s3:GetObjectAcl",
"s3:GetBucketObjectLockConfiguration",
"s3:DeleteBucketWebsite",
"s3:GetIntelligentTieringConfiguration",
"s3:GetObjectVersionAcl",
"ec2:DescribeReservedInstances",
"ec2:DescribeNetworkAcls",
"s3:GetBucketPolicyStatus",
"sqs:GetQueueUrl",
"s3:GetObjectRetention",
"s3:GetJobTagging",
"iam:GetAccessKeyLastUsed",
"s3:ListJobs",
"sqs:GetQueueAttributes",
"s3:PutObjectLegalHold",
"s3:PutBucketCORS",
"s3:ListMultipartUploadParts",
"s3:GetObject",
"s3:DescribeJob",
"s3:PutBucketLogging",
"s3:GetAnalyticsConfiguration",
"s3:GetObjectVersionForReplication",
"s3:GetAccessPointForObjectLambda",
"s3:CreateAccessPoint",
"s3:GetAccessPoint",
"ec2:DescribeAddresses",
"s3:PutAccelerateConfiguration",
"s3:DeleteObjectVersion",
"s3:GetBucketLogging",
"s3:ListBucketVersions",
"s3:RestoreObject",
"s3:GetAccelerateConfiguration",
"s3:GetObjectVersionAttributes",
"s3:GetBucketPolicy",
"s3:PutEncryptionConfiguration",
"s3:GetEncryptionConfiguration",
"s3:GetObjectVersionTorrent",
"s3:GetBucketRequestPayment",
"s3:GetAccessPointPolicyStatus",

```

"s3:GetObjectTagging",
"s3:GetBucketOwnershipControls",
"s3:GetMetricsConfiguration",
"s3:GetBucketPublicAccessBlock",
"sqs:ListQueues",
"s3:GetMultiRegionAccessPointPolicy",
"s3:GetAccessPointPolicyStatusForObjectLambda",
"s3:ListAccessPoints",
"s3:PutBucketOwnershipControls",
"s3:DeleteMultiRegionAccessPoint",
"s3:ListMultiRegionAccessPoints",
"s3:UpdateJobStatus",
"s3:GetBucketAcl",
"ec2:DescribeImages",
"s3:ListStorageLensConfigurations",
"s3:GetObjectTorrent",
"cloudfront:ListDistributions",
"iam:ListUsers",
"s3:GetBucketLocation",
"s3:GetAccessPointPolicy",
"s3:ReplicateDelete"
],
"Resource": "*"
}
]
}

```

The above-mentioned script provides details of all CloudTrail, EC2 and S3 services and integrates with Splunk as soon as one connects his/her account and provides analysis based on that.

Furthermore, an SQS (Simple Queue Service) service is also used to align the above given services in a queue so they can be tracked easily and if there is any change or discrepancy during adding, updating or deleting a file within a S3 bucket, an SNS (Simple Notification Service) will be sent to the respective cloud account to check any changes have occurred or not.

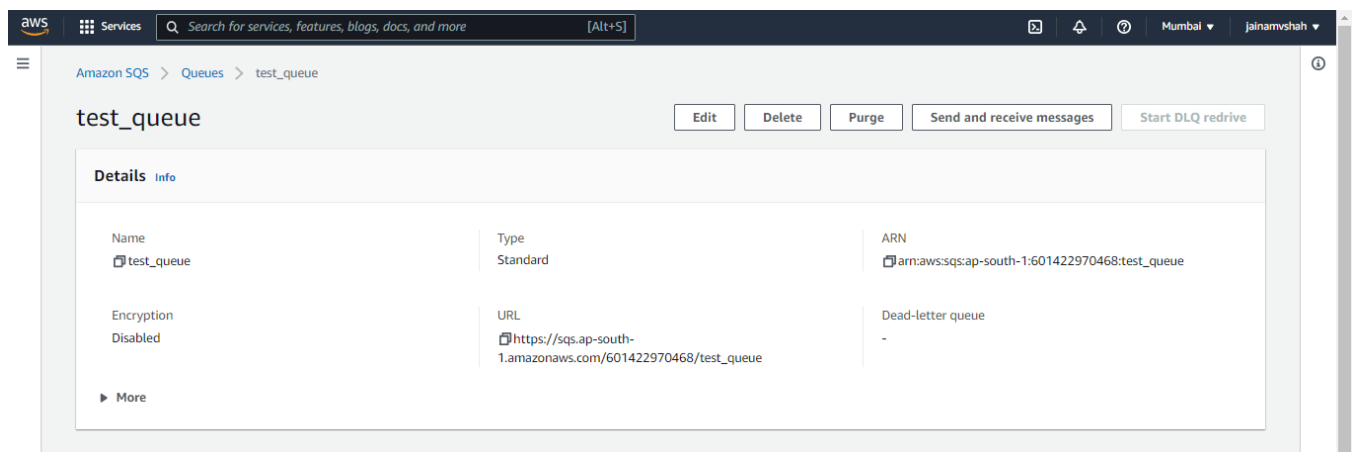


Figure 6.26 SQS Service

6.3.7 Forensic Analysis After Malware Attack

After performing a malware attack as shown above in 6.3.4 section a number of parameters can be considered to take account of from Splunk as they provide certain insights of the activities taking place within the S3 bucket.

Splunk provides a feature to export the results from CloudTrail logs in the form of a csv file as shown below.

File

Home

Insert

Page Layout

Formulas

Data

Review

View

Help

Tell me what you want to do

Cut

Copy

Paste

Format Painter

Calibri

11

A

A

B

I

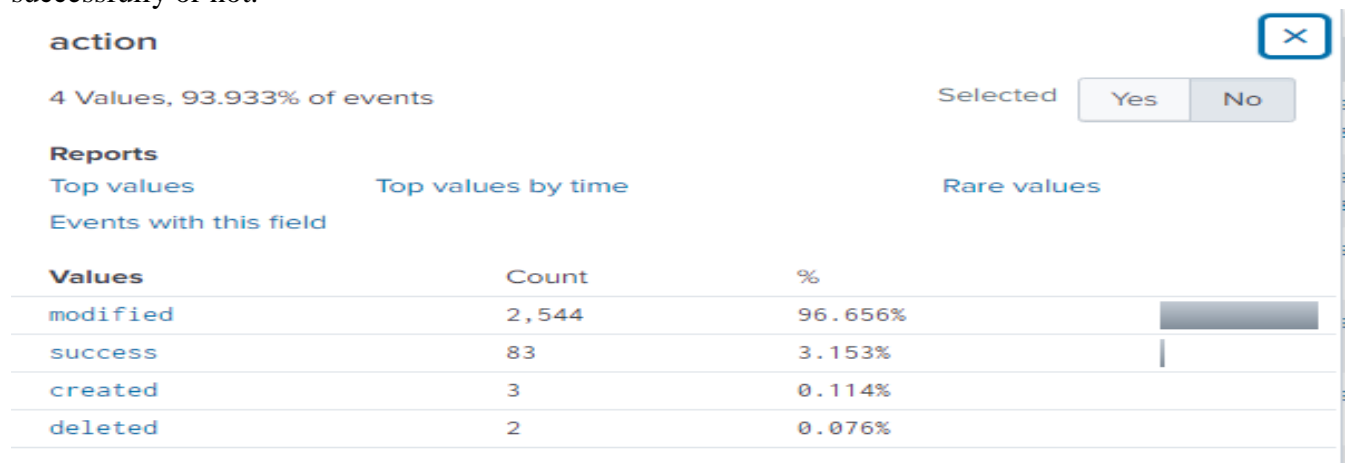
U

Figure 6.27 Results File

As shown above there is a csv file generated which contains certain important fields showing the activities performed within the S3. For example: -

Column B-time – shows the time at which a certain activity was performed within S3

Column C-action – shows which activity was performed like what was modified, deleted or was it done successfully or not.



Values	Count	%
modified	2,544	96.656%
success	83	3.153%
created	3	0.114%
deleted	2	0.076%

Figure 6.28 Action Field

Column P-command – this column shows what kind of command was executed within the bucket like PutObject, HeadObject, SetTopic, LookUpEvent etc.

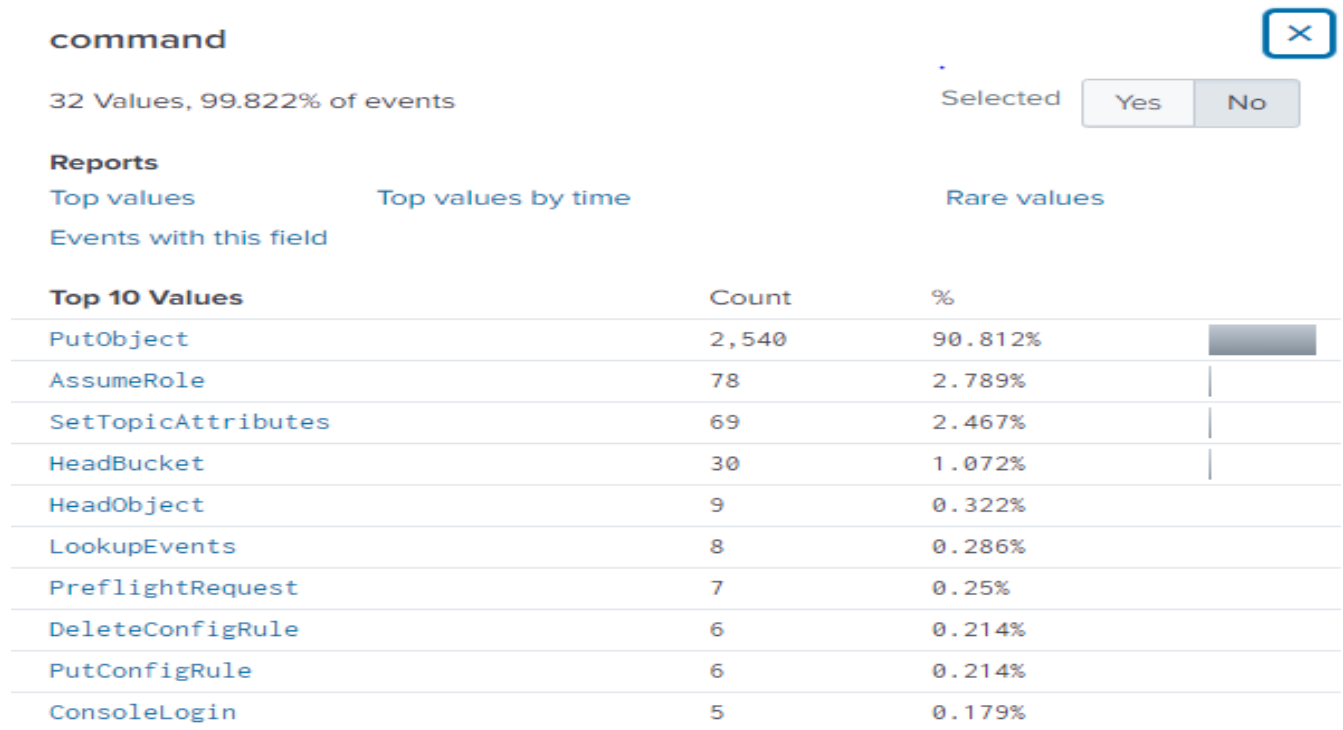


Figure 6.29 Command Field

Column AE-errorcode – shows whether the command was successful or access was denied.

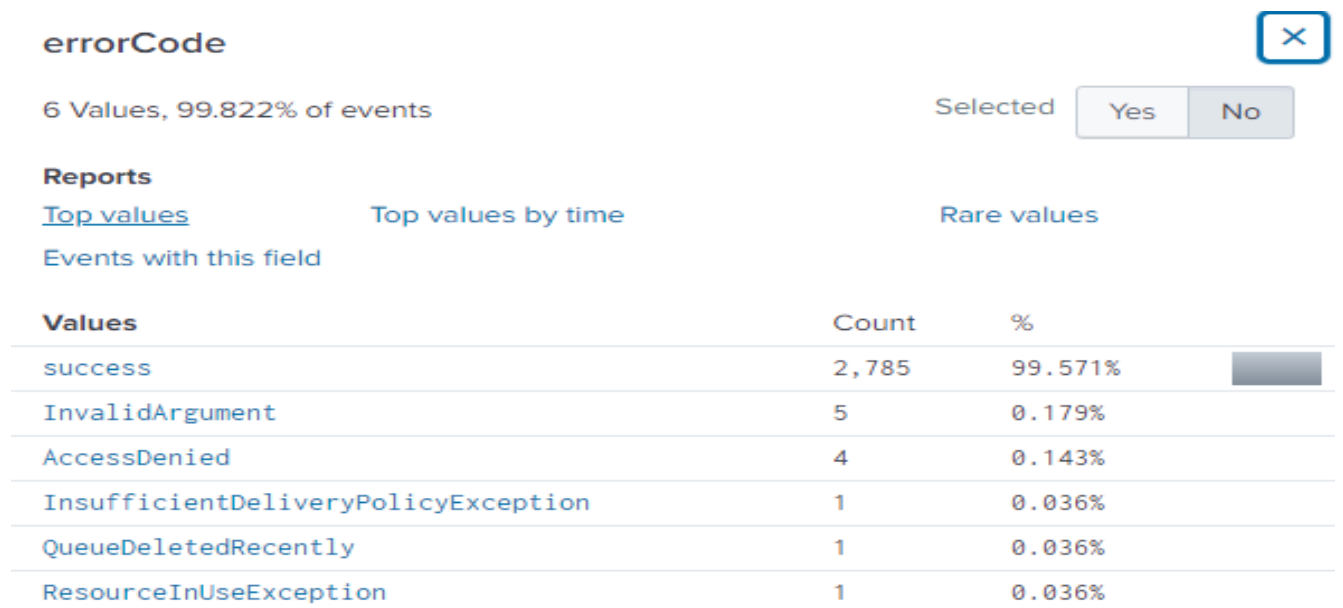


Figure 6.30 ErrorCode Field

Column AO -Host – this column shows from where was the above-mentioned commands were carried out on a respective bucket.

host ×		
1 Value, 100% of events		Selected <input type="button" value="Yes"/> <input type="button" value="No"/>
Reports		
Top values	Top values by time	Rare values
Events with this field		
Values	Count	%
DESKTOP-AIRQ0C5	339	100%

Figure 6.31 Host Field

Column CT – requestparameterkey – shows the name of files which are being uploaded within the respective S3 Bucket.

requestParameters.key ×		
>100 Values, 91.399% of events		Selected <input type="button" value="Yes"/> <input type="button" value="No"/>
Reports		
Top values	Top values by time	Rare values
Events with this field		
Top 10 Values	Count	%
Attendance 2k21 sem 7.xlsx	7	0.273%
AWSLogs/601422970468/CloudTrail/us-east-1/2022/03/09/601422970468_CloudTrail_us-east-1_20220309T1950Z_BSaYNWBn75mFghsk.json.gz	4	0.156%
AWSLogs/601422970468/CloudTrail/us-east-1/2022/03/09/601422970468_CloudTrail_us-east-1_20220309T1650Z_5HNNMlsc40sNz0v.json.gz	3	0.117%
NIST_CSF_Risk-CMM-2016.xlsx	3	0.117%
NIST_CSF_Risk_CMM-2017-markup.xlsx	3	0.117%
NIST_CSF_Risk_CMM-2017.xlsx	3	0.117%
README.md	3	0.117%
Sensitive Info Table_v3.7.xlsx	3	0.117%
AWSLogs/601422970468/CloudTrail-Digest/af-south-1/2022/03/10/601422970468_CloudTrail-Digest_af-south-1_demo2.1_ap-south-1_20220310T052918Z.json.gz	1	0.039%
AWSLogs/601422970468/CloudTrail-Digest/af-south-	1	0.039%

Figure 6.32 RequestParameterKey Field

CHAPTER: 7 CONCLUSION AND FUTURE WORK

CHAPTER 7 CONCLUSION AND FUTURE WORK

Conclusion

Hence, we can conclude that we have successfully established a solution on how a user can monitor his/her data if it uploaded on cloud premises using Billing preferences alarm and CloudWatch Alarm. Any type of malicious activity which might takes place on the cloud account can be mitigated if the data is monitored properly. Another Service called AWS CloudTrail has been utilized which generated logs of activities taking place within S3. These logs are integrated with Splunk (SIEM Tool) which provides proper analysis and insights of activities and also if any malicious activity has taken place within S3.

Future Work

For the remaining part of work, we intend to dive deep into the concept of evidence capturing and forensic evidence gathering of a malware attack which might or could take place on data uploaded on IaaS. A forensic investigation machine could be developed which is called a SIFT tool to prepare a proper report and provide necessary investigation tools if a malware attack takes place or not.

CHAPTER: 8 REFERENCES

CHAPTER 8 REFERENCES

- [1] B. Varghese and R. Buyya, "Next Generation Cloud Computing: New Trends and Research Directions," Elsevier :Future Generation Computer Systems, Vol. 79, pp. 1-22, September 2017.
- [2] <https://towardsdatascience.com/malware-detection-using-deep-learning-6c95dd235432>
- [3] www.youtube.com
- [4] Malware Detection in Cloud Computing Infrastructures By Michael R. Watson, Noor-ul-Hassan Shirazi
- [5] A. Amazon Web Services, *Amazon CloudWatch Developer Guide*, 2010.
- [6] https://www.researchgate.net/publication/304452598_Comparitive_Study_of_Cloud_Forensics_Tools
- [7] <https://docs.splunk.com/Documentation>