

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



Institute of
Computer
Technology

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 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

With the advent of Cloud computing being espoused very briskly by organizations with diverse businesses and sizes, the utilization of cloud services is soaring at an untrackable rate these days more importantly IaaS services as cloud providers allow more secured resources with supple offerings and models. This escalating adoption gives birth to new surface attacks to organizations that attackers tend to abuse with their malware to dominate these invaluable resources and the important data that is stored on them. Therefore, for organizations in order to well guard against these malware attacks they need to have full discernibility not only on their data centers but also on their resources which are stored on the cloud. This proposed project discusses and aims to yield the best approaches to attain continuous monitoring of malware attacks on the cloud along with their phases (before, during, and after). This project line ups to defines the best methods to bring loggings and forensics to the cloud platform and integrate them with on-premises visibility, thus attaining the full monitoring over the whole security standpoint 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 | 10 |
| 5.1.4 Time Duration to Complete Second Phase | 11 |
| 5.1.5 Tasks for Evidence Capturing and Forensic Analysis in Third Phase | 11 |
| 5.1.6 Time Duration to Complete Third Phase | 11 |
| CHAPTER 6: IMPLEMENTATION DETAILS | 12-42 |
| 6.1 Background | 13 |
| 6.2 Methodology | 13 |
| 6.2.1 Gathering Data | 13 |
| 6.3 Cloud Analysis to Malware Detection | 14 |
| 6.3.1 Testing Environment | 14 |
| 6.3.2 Data Set | 14 |
| 6.3.2 Testing and Analysis | 14 |
| 6.3.4 Testing Phases | 14-25 |
| 6.3.5 Generating Data Logs from AWS CloudTrail | 25-27 |
| 6.3.6 Integrating AWS CloudTrail with Splunk | 27-29 |
| 6.3.7 Forensic Analysis After Malware Attack | 29-32 |
| 6.4 Forensic Analysis in IaaS Cloud | 32-37 |
| 6.4.1 Additional Forensic Analysis | 38-42 |
| CHAPTER 7: CONCLUSION AND FUTURE WORK | 43-44 |
| CHAPTER 8: REFERENCES | 45-46 |

CHAPTER: 1 INTRODUCTION

CHAPTER 1 INTRODUCTION

The availability to store data on cloud is a modern technology. In recent years, utilization of services on cloud platform is skyrocketing at an alarming pace, now it has turned more famous after the emergence of the 4th Economic Industrial Revolution. In the year of 2020, nearly 83 percentage of workloads in business are functional on the cloud platform, and around 94 percentage of companies in today's market utilize a cloud service in one of their services. There are approximately 3 most used cloud services including Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Monitoring the current market, (IaaS) is the most rapidly growing service in Cloud Computing.

However due to ample number of exquisite features being available on the IaaS cloud, it is turning into a playground to many attacks of malware for the following reasons:

- 1) Companies which are providing cloud services unremittingly offer off the chart's performance with more computation power for their clients. These virtual machines on cloud are primary targets for crypto currency mining malware.
- 2) The rise of work from home era and internationally separated manpower and resources accessibility after the Sars-Cov2 (corona virus) has provided the attackers more ways to conceal their detrimental traffic to take over the cloud-hosted virtual machines, and utilize them for their malicious activities.
- 3) The definite rise in IoT applications that use cloud-hosted data and services to analyze the gigantic quantity of data created by these applications to construct 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 directs its goal to utilize the best ways to attain non-stop monitoring of malware attacks on the cloud.
- The techniques of logging data and performing forensics have always been the foundations of accomplishing non-stop monitoring and detection of malware attacks.
- To arrogate the perfect methods to bring the concept of forensics and logging to the cloud and integrate them with on-premises visibility.
- Attaining the proper monitoring considering the whole security standpoint of the organization assets whether they are stored on an on-premises system or on the cloud platform.

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 | 8GB |
| HDD | Minimum 30GB |

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

| | |
|-------------------------------|---|
| OS | Any operating system which can support an internet browser. |
| Programming languages | - |
| Tools and Technologies | AWS, Splunk, kali Linux |

Table 3.2 Minimum Software Requirements

CHAPTER: 4 PROCESS MODEL

CHAPTER 4 PROCESS MODEL

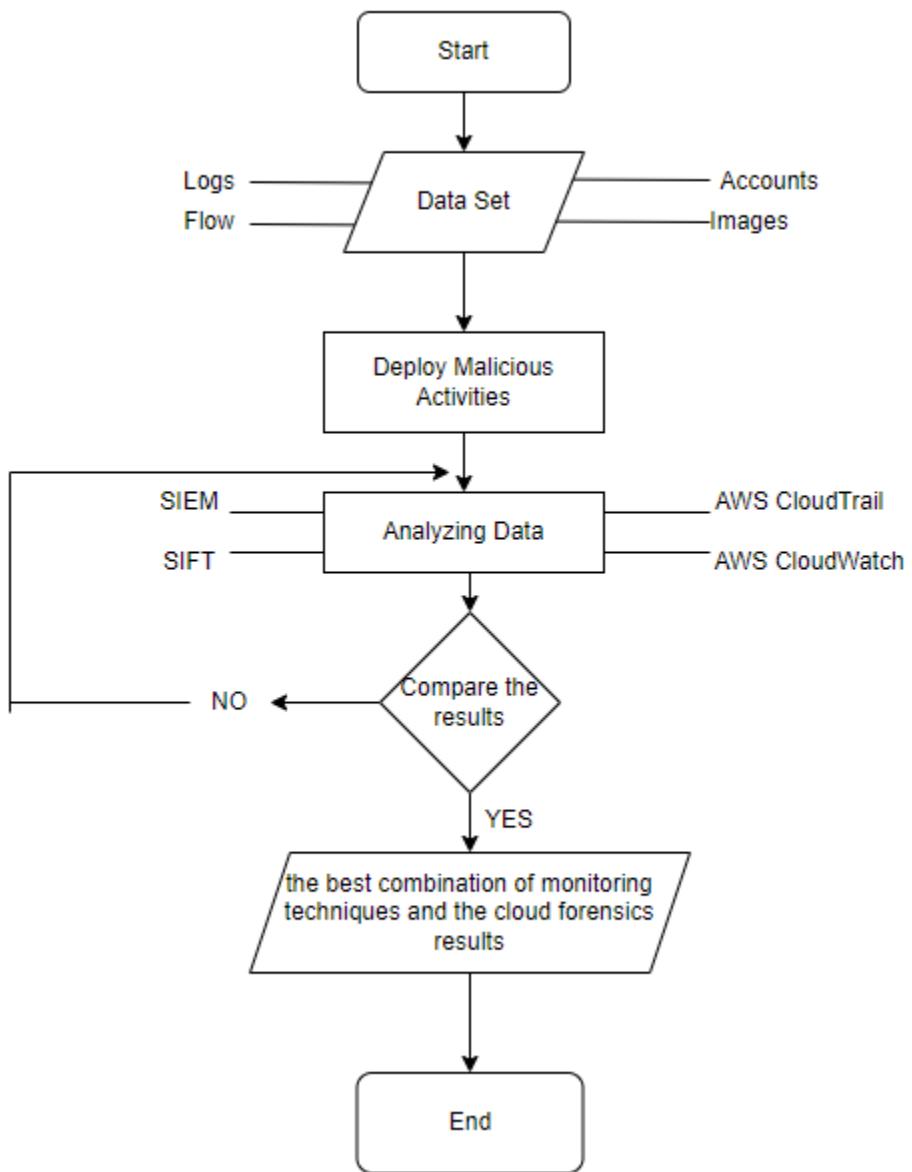


Figure 4.1 Process Model of Project

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

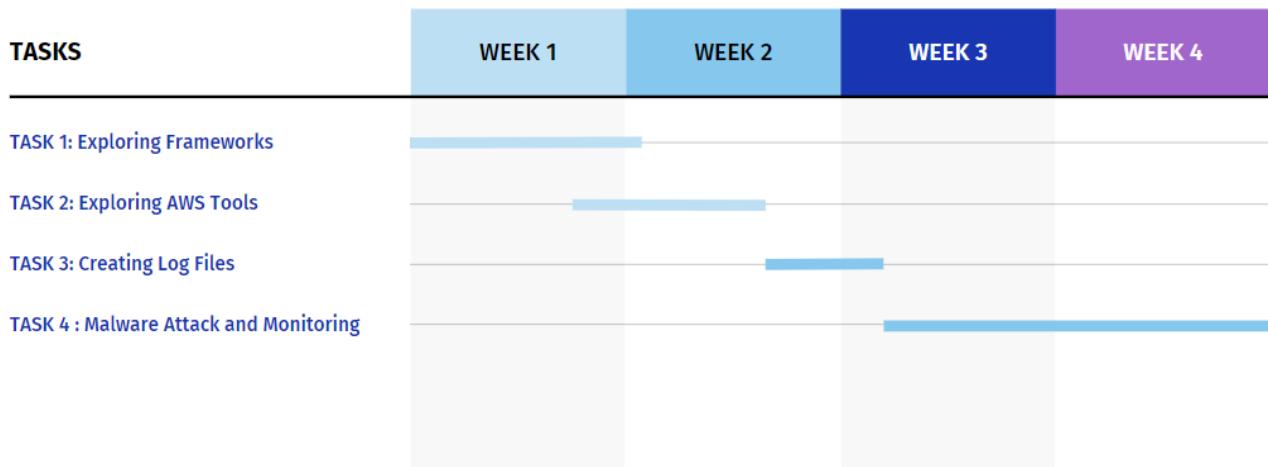


Figure 5.1 Task Completion Time Duration in First Phase

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

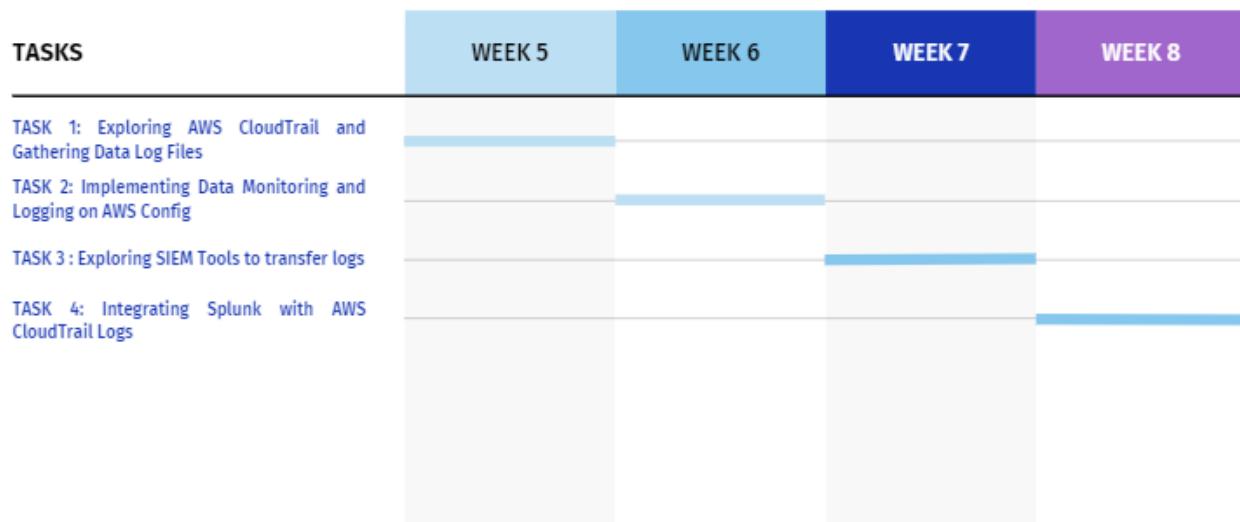


Figure 5.2 Task Completion Time Duration in Second Phase

5.1.5 Tasks for Evidence Capturing and Forensic Analysis in Third Phase

- Task: - 1 SIFT Exploration
- Task: - 2 AWS EC2 Exploration and setting up investigation tools on EC2
- Task: - 3 Cloud Forensic Analysis and Evidence Capturing
- Task: - 4 Additional Cloud Forensics

5.1.6 Time Duration to Complete Third Phase

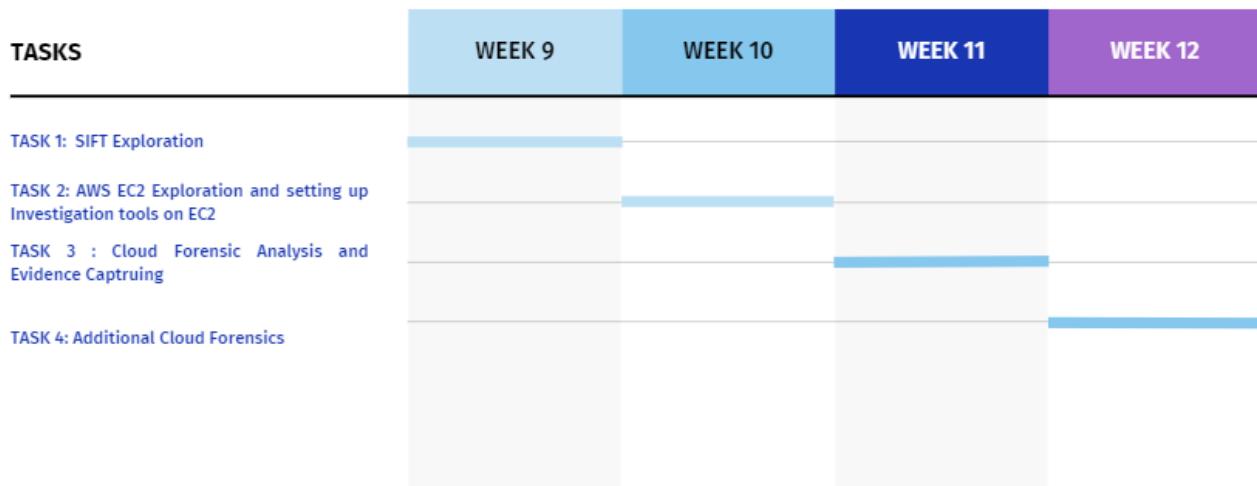


Figure 5.3 Task Completion Time Duration in Third Phase

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 can be considered as the most important service, which provides basic computational services such as servers, networking, and storage. This service supplements the availability of the system on lower costs and providing a more pliable system.

2. Malware Attacks - In layman terms the term malware means dangerous and harmful, it has similar effect on software, networks, OS, or other components. The most major challenge while using IaaS cloud world is its vulnerability and the possibility of malware attacks; it is a vital concern to devices present at home as well as the devices used for business in a corporate and also on cloud VMs.

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

4. Cloud Forensics – It can be defined as techniques that are utilized in order to perform collecting and storing incidents happening around and their visibility, remodeling events, recognizing when an incident is happening, how an incident is happening, and where an incident is happening, and implementing information/data regarding that.

6.2 Methodology

The methodology of the mentioned project has been segregated into two parts:

The First: Whenever or wherever a malware attack takes place, the investigator should make cloud analysis for that particular malware detection.

The Second: is to perform forensics analysis in the IaaS Cloud after the attack happens.

6.2.1 Gathering Data

In today's, there are two initiatives that define and segregate in an orderly manner how each cloud attack technique is witnessed; they are 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 for this mentioned project were performed on Amazon Web services (AWS), the main reason for choosing AWS for this project was because it is currently the market leader and provides so many public cloud services and possesses a varied range of service catalog and thus in turn making it a more suitable choice for the mentioned problem.

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.

Non-Stop monitoring on IaaS can be attained by collecting and processing the following

- Monitoring API Calls (CloudTrail's logs in AWS).
- Hosting logs and logs of deployed HIDs.
- VPC flows.
- Logs of numerous cloud resources (CloudWatch Logs in AWS)
- Validation and Integrity of images and instances.

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

AWS CloudWatch – to gather and monitor metrics, gather and monitor log files, setting 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

As per the MITRE ATT&CK framework, most majorly used attack vectors for attacks on cloud and malware attacks aiming cloud-hosted environments is the cloud account takeover. There are multiple ways to detect that, the most promising way is detecting changes in the billings on AWS. Most public cloud providers provide features to allow their customers to make billing tags and then send them emails whenever alarms are triggered.

The screenshot shows the AWS Billing Preferences page. On the left, there's a sidebar with navigation links like Home, Billing, Bills, Orders and invoices, Credits, Purchase orders, Cost & Usage Reports, Cost Categories, Cost allocation tags, Cost Management, Cost Explorer, Budgets, Budgets Reports, Savings Plans, Preferences, Billing preferences (which is highlighted in orange), Payment methods, Consolidated billing, and Tax settings. The main content area is titled 'Preferences' and contains sections for 'Billing Preferences' and 'Cost Management Preferences'. Under 'Billing Preferences', there are two checkboxes: 'Receive PDF Invoice By Email' (unchecked) and 'Receive Free Tier Usage Alerts' (checked). Below the second checkbox is a text input field with the email address 'jainamvshah18@gnu.ac.in'. Under 'Cost Management Preferences', there is another checkbox 'Receive Billing Alerts' (checked) with a note explaining its function. A 'Save preferences' button is at the bottom of the page. At the very bottom, there are links for Feedback, English (US), © 2022, Amazon Internet Services Private Ltd. or its affiliates., Privacy, Terms, and Cookie preferences.

Figure 6.1 AWS Billing Preferences

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

2. Performing Continuous Monitoring in AWS Environment

AWS Cloud platform provides a service called AWS Config, this service enables monitoring AWS resource configurations and keep tracks on resource inventory and varied changes that might take place within it, which can be utilized to detect any malicious configuration changes that an attacker might try to make in order to gain control over the compromised account's resources. The monitoring information gathered from this then can be absorbed using AWS CloudWatch and SNS Notifications can be created based on them.

Malware attacks earmark and make changes to the data stored and any misconfigured cloud storage leading to data leak. By making use of AWS Config, the rules like proper storage versioning is kept on for AWS storage (S3). By enabling the s3-bucketversioning- enabled rule, another activity auditioned by attackers is to hide their vindictive API calls by disabling API calls monitoring, another rule that can be configured is to check if CloudTrail service is enabled or not and yet another rule to detect whether the volumes which are utilized are having encryption or not.

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

The screenshot shows the AWS S3 Buckets page. On the left, there's a sidebar with options like Buckets, Storage Lens, and Feature spotlight. The main area displays an account snapshot and a list of buckets. The bucket list table has columns for Name, AWS Region, Access, and Creation date. The buckets listed are:

| Name | AWS Region | Access | Creation date |
|---|----------------------------------|-----------------------|--|
| aws-cloudtrail-logs-601422970468-722e7fc6 | Asia Pacific (Mumbai) ap-south-1 | Public | February 1, 2022, 17:02:31 (UTC+05:30) |
| jainamvshah18 | Asia Pacific (Mumbai) ap-south-1 | Objects can be public | February 1, 2022, 09:18:07 (UTC+05:30) |
| malwaredemo | Asia Pacific (Mumbai) ap-south-1 | Public | February 2, 2022, 19:11:52 (UTC+05:30) |

Figure 6.2 S3 Buckets

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

The screenshot shows the AWS Bucket metrics page for the 'malwaredemo' bucket. The left sidebar is identical to Figure 6.2. The main area shows 'Bucket metrics' with tabs for 'Storage metrics' and 'Request metrics'. Under 'Request metrics', there are filters for 'Choose filters' (set to 'mal_demo_filter' and 'Entire bucket') and time intervals (1h, 3h, 12h, 1d, 1w, 2w). Two charts are displayed: 'All requests' (HTTP requests made to the bucket) and 'Get requests' (HTTP GET requests for objects in the bucket). Both charts show a single data series with a blue dot at the end of the timeline.

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.

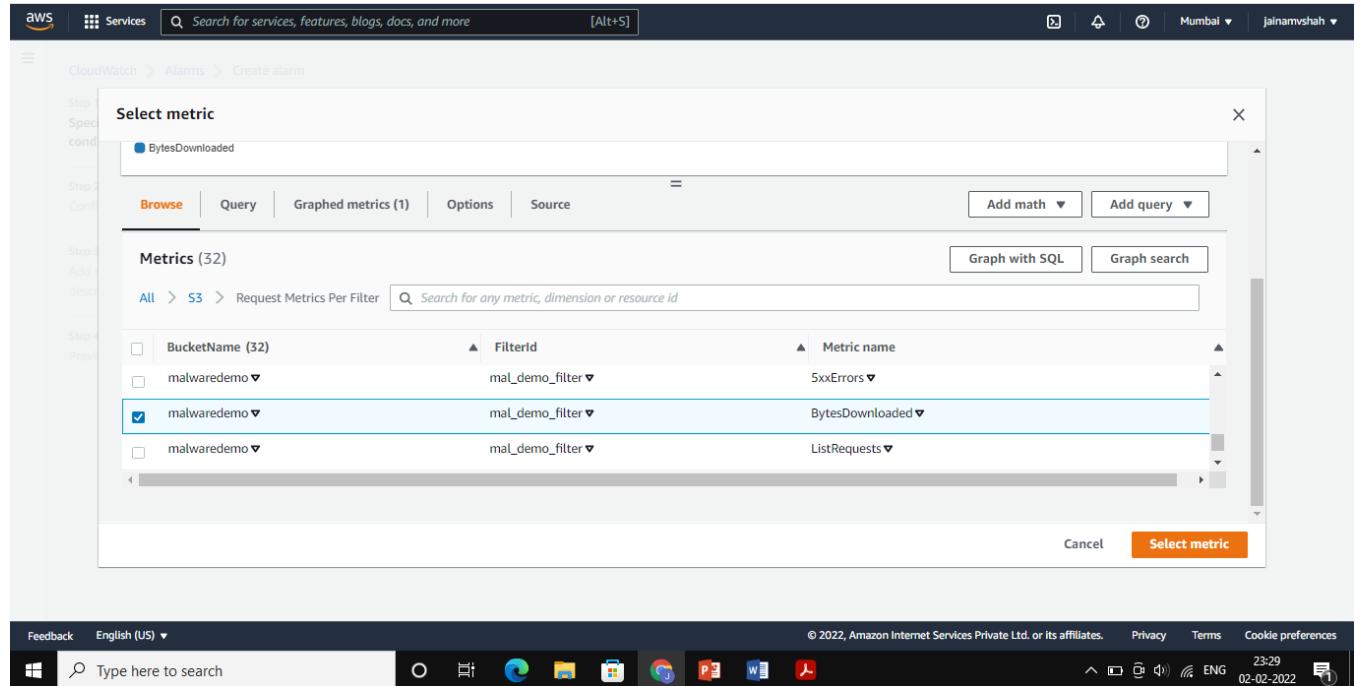


Figure 6.4 Setting up Alarm

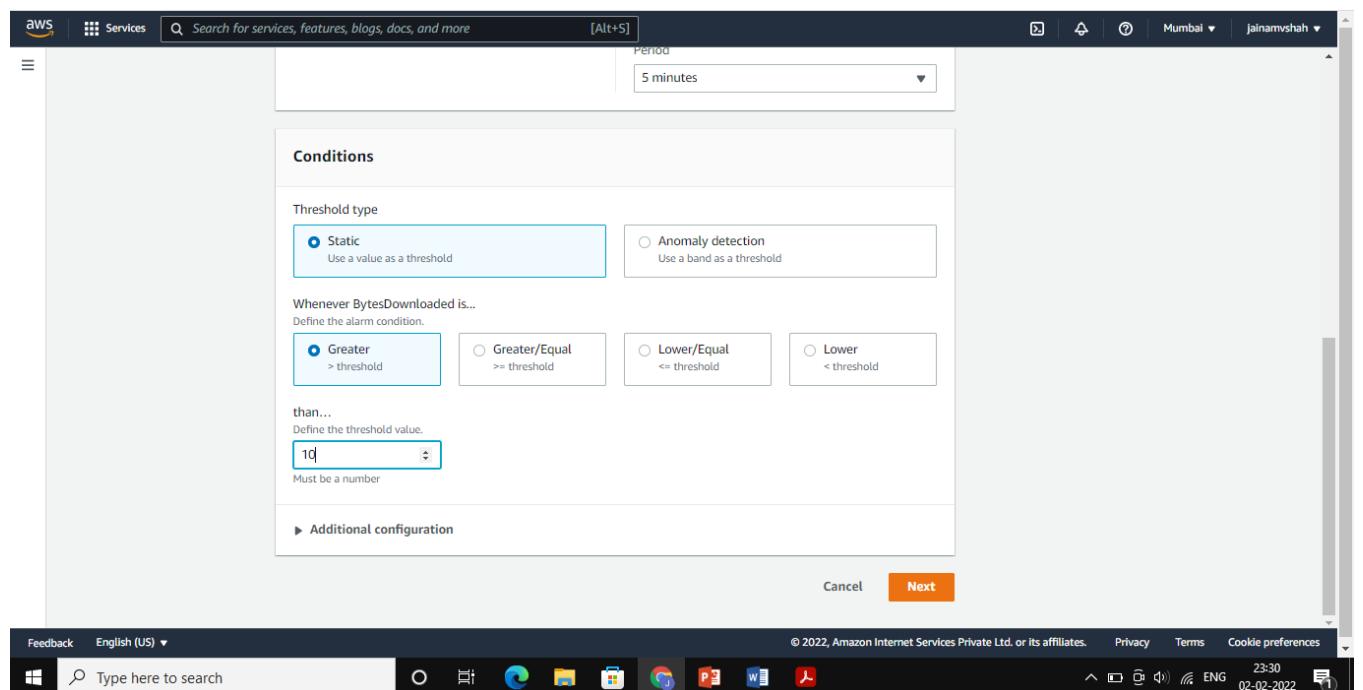
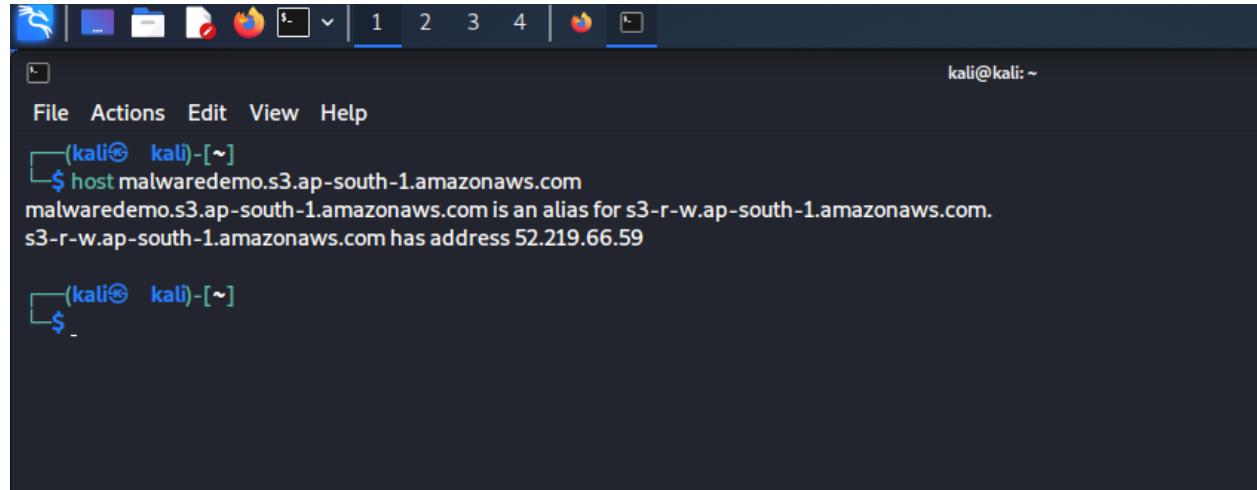


Figure 6.5 Defining threshold value for a definite amount of size

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

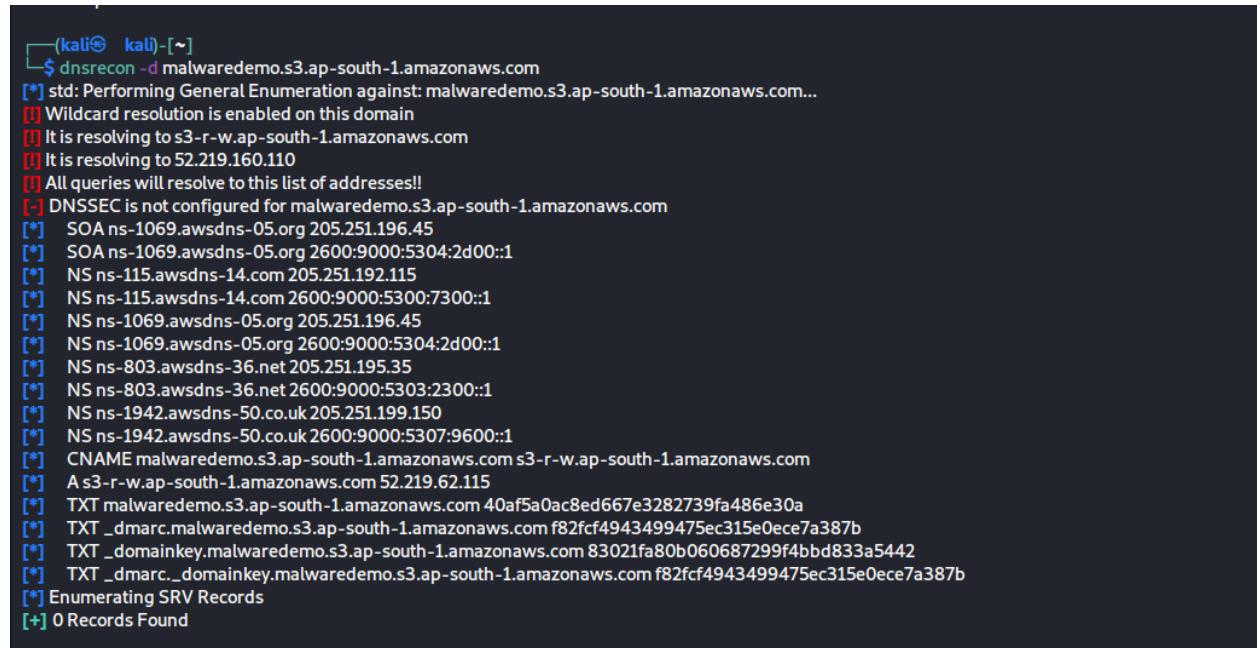


A screenshot of a terminal window titled '(kali㉿ kali)-[~]'. The window shows the command '\$ host malwaredemo.s3.ap-south-1.amazonaws.com' being run. The output indicates that 'malwaredemo.s3.ap-south-1.amazonaws.com' is an alias for 's3-r-w.ap-south-1.amazonaws.com', which has the address '52.219.66.59'. The terminal prompt '\$' is visible at the bottom.

```
$ 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
```

Figure 6.6 Identifying IP Address

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



A screenshot of a terminal window titled '(kali㉿ kali)-[~]'. The window shows the command '\$ dnsrecon -d malwaredemo.s3.ap-south-1.amazonaws.com' being run. The output provides detailed DNS enumeration information for the specified domain, including SOA records, NS records, and CNAME mappings. It also lists TXT records for DMARC and domainkey services.

```
$ 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㉿ kali)-[~]
└─$ nslookup 52.219.66.59
      1 ×

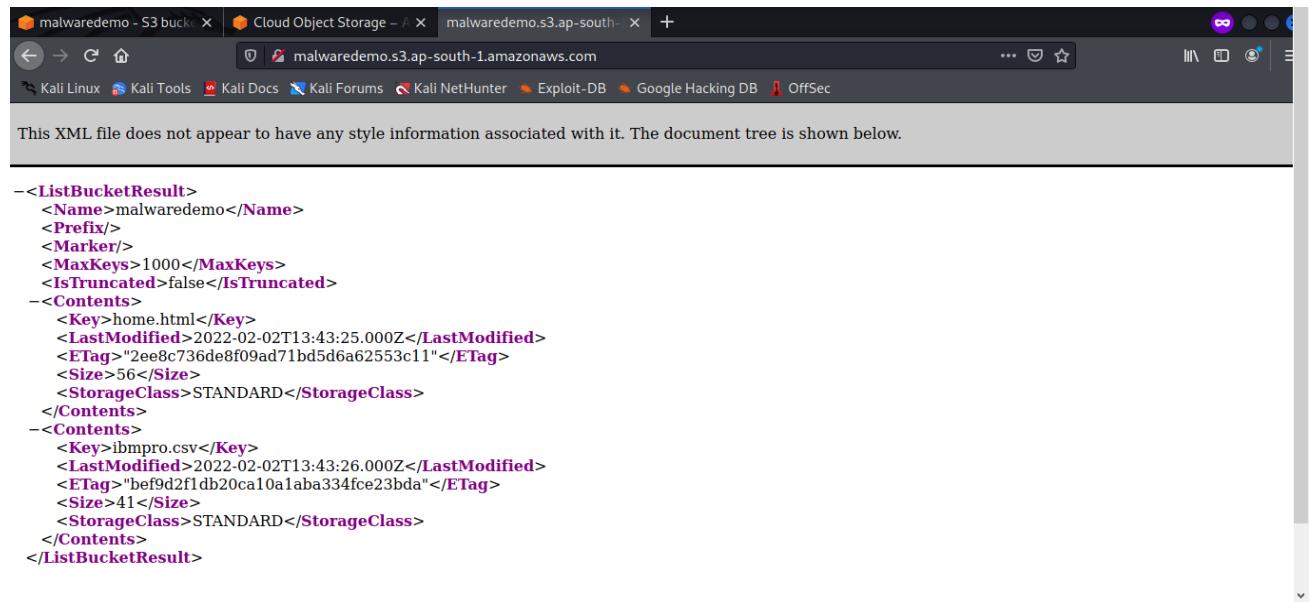
59.66.219.52.in-addr.arpa    name = s3-r-w.ap-south-1.amazonaws.com.

Authoritative answers can be found from:

└─(kali㉿ 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.



This screenshot shows a web browser window with three tabs open. The active tab is titled "malwaredemo.s3.ap-south-1.amazonaws.com". The address bar also displays this URL. Below the tabs, there's a navigation bar with links like "Kali Linux", "Kali Tools", "Kali Docs", "Kali Forums", "Kali NetHunter", "Exploit-DB", "Google Hacking DB", and "OffSec". The main content area of the browser shows the XML output of an S3 bucket listing. The XML is as follows:

```
<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      in-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

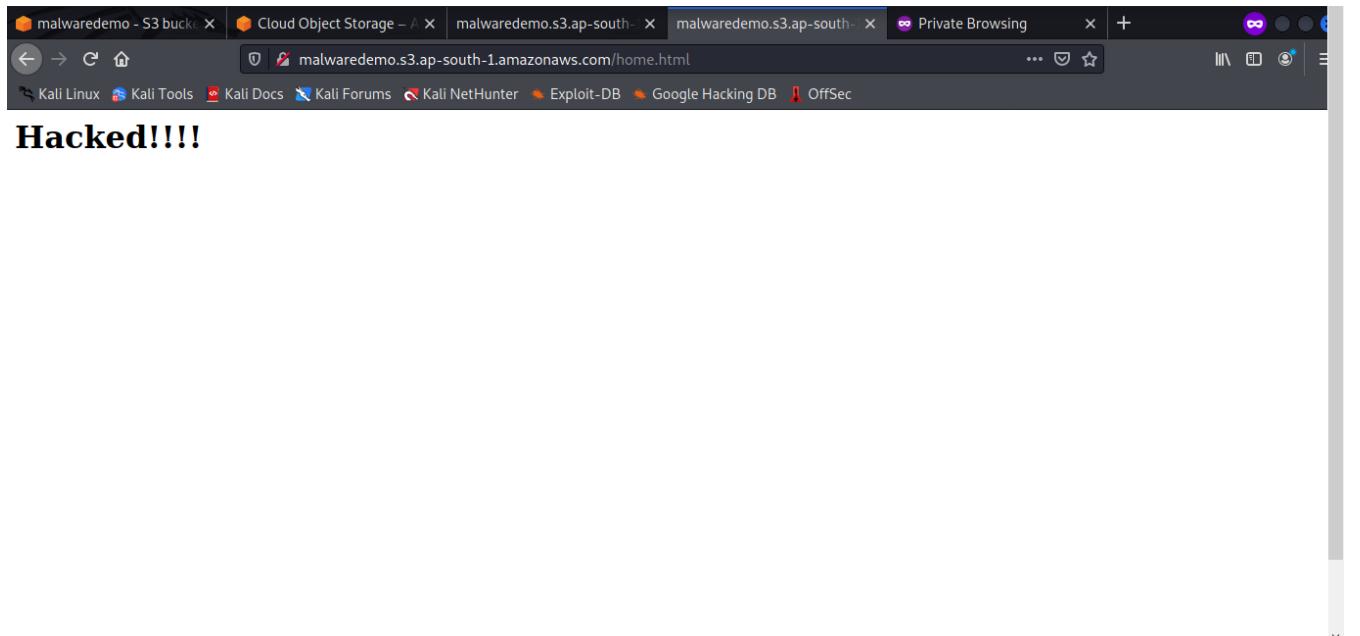


Figure 6.11 Home.html file

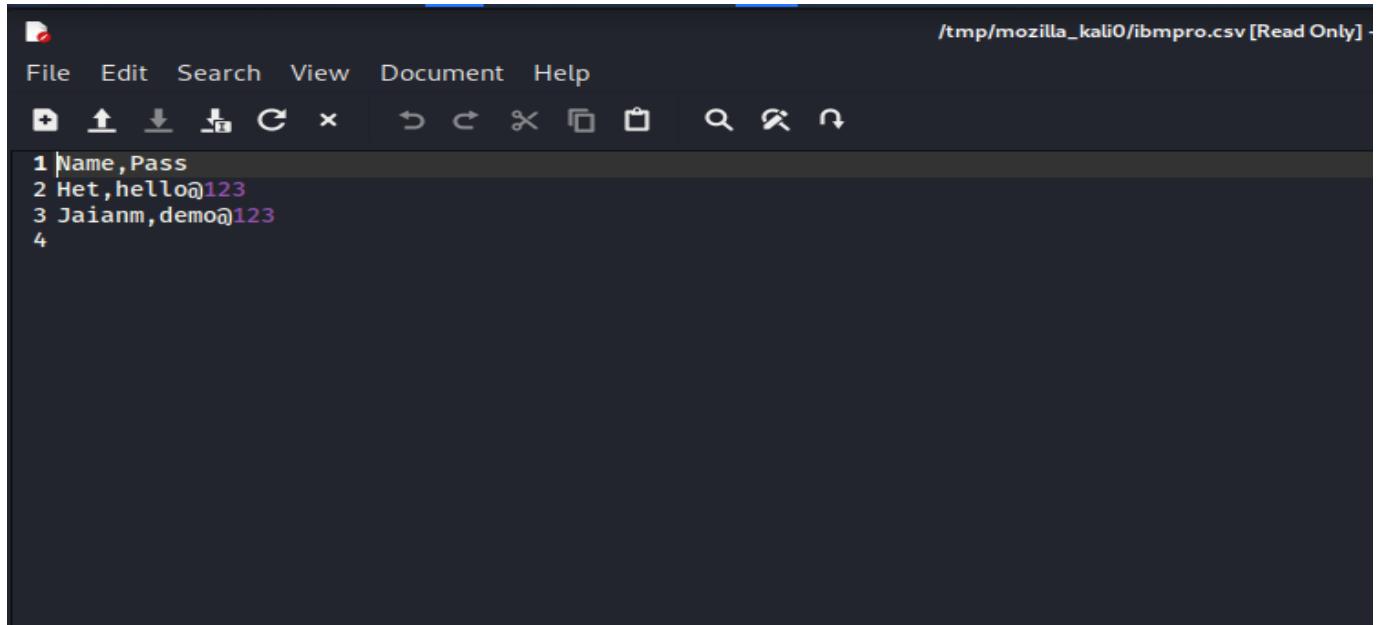


Figure 6.12 ibmpro.csv

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

```
(kali㉿ kali)-[~]
$ python3 -m S3Scanner scan --bucket malwaredemo.s3.ap-south-1.amazonaws.com
Warning: AWS credentials not configured - functionality will be limited. Run: `aws configure` to fix this.

malwaredemo | bucket_exists | AuthUsers: [], AllUsers: [Read, ReadACP]

(kali㉿ kali)-[~]
$
```

Figure 6.13 Listing Users

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

```
(kali㉿ kali)-[~]
$ python3 -m S3Scanner dump --bucket malwaredemo.s3.ap-south-1.amazonaws.com --dump-dir ~/Desktop/S3Dump/
Warning: AWS credentials not configured - functionality will be limited. Run: `aws configure` to fix this.

malwaredemo | Enumerating bucket objects...
malwaredemo | Total Objects: 2, Total Size: 97.0B
malwaredemo | Dumping contents using 4 threads...
malwaredemo | Dumping completed

(kali㉿ kali)-[~]
$
```

Figure 6.14 Dump status

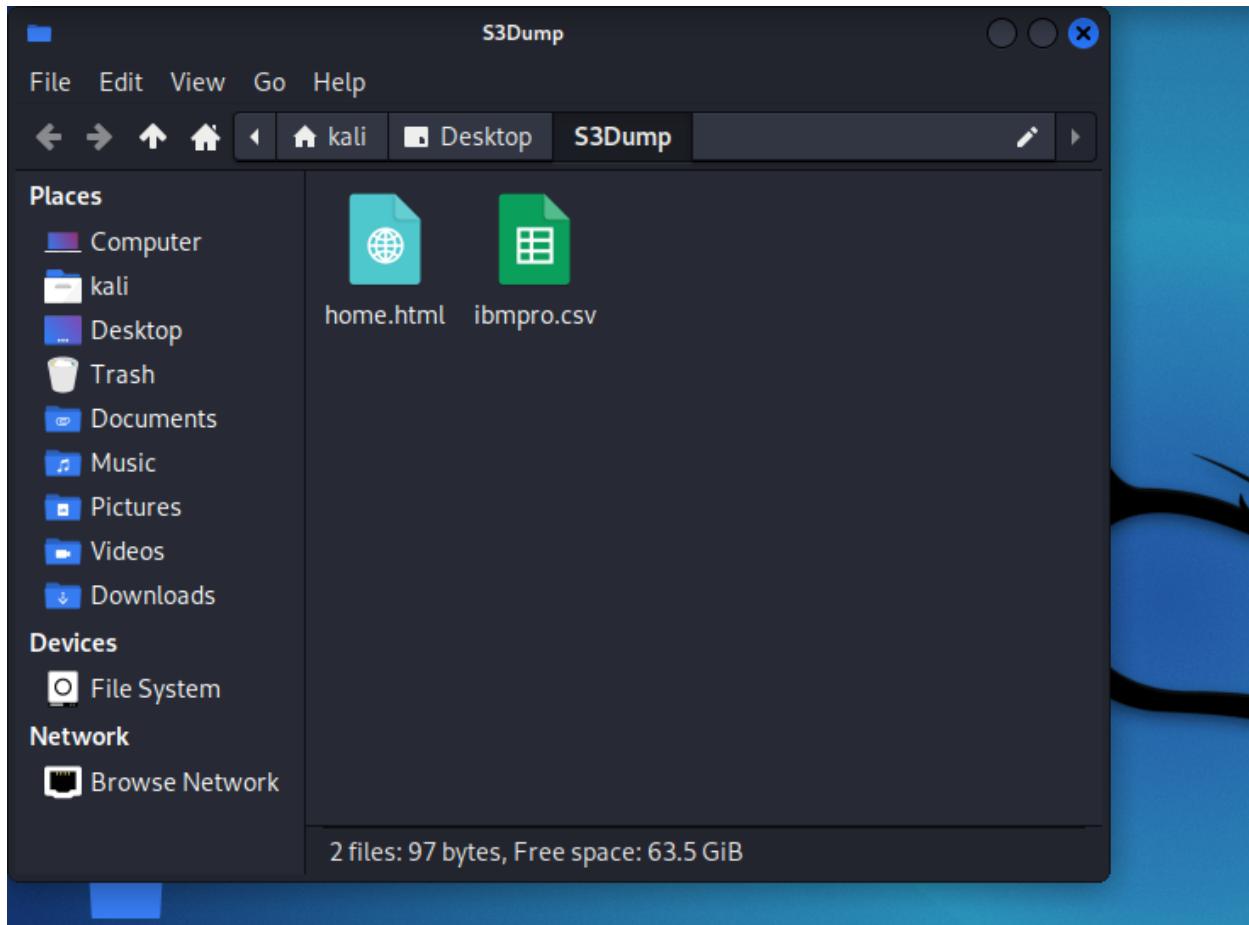


Figure 6.15 Files downloaded on local machine

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.

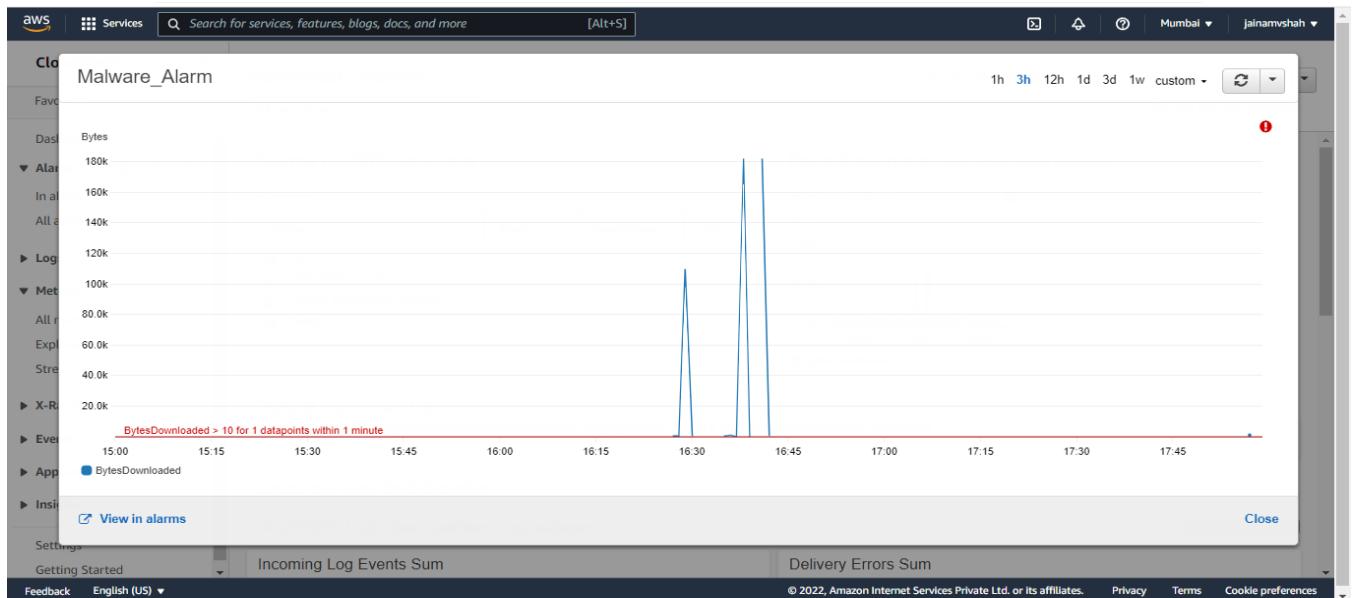


Figure 6.16 Malware Alarm

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.

The screenshot shows the AWS SNS Topics page. On the left, a sidebar menu includes 'Dashboard', 'Topics' (which is selected and highlighted in orange), 'Subscriptions', and 'Mobile' sections with 'Push notifications', 'Text messaging (SMS)', and 'Origination numbers'. The main content area shows a single topic named 'aws-cloudtrail-logs-601422970468-d3e8a5a9'. The topic details are as follows:

| Name | Display name |
|--|--------------|
| aws-cloudtrail-logs-601422970468-d3e8a5a9 | - |
| ARN | Topic owner |
| arnaws:sns:ap-south-1:601422970468:aws-cloudtrail-logs-601422970468-d3e8a5a9 | 601422970468 |
| Type | |
| Standard | |

Buttons for 'Edit', 'Delete', and 'Publish message' are located at the top right of the topic details section. The top navigation bar includes the AWS logo, a search bar, and links for 'Services', 'Mumbai', and 'jainamshah'.

Figure 6.17 SNS Topic

The screenshot shows the AWS SNS Subscriptions page. The top navigation bar includes tabs for 'Subscriptions' (which is selected and highlighted in orange), 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. Below the tabs, a table displays three confirmed subscriptions:

| ID | Endpoint | Status | Protocol |
|--------------------------------------|--|-----------|----------|
| d3ca453f-9411-4ecb-9119-f4db84d6771a | jainamshah18@gnu.ac.in | Confirmed | EMAIL |
| ada7d270-4b69-4ebb-b2b9-cb2be3785599 | harshvardhanrahevar18@gnu.ac.in | Confirmed | EMAIL |
| 1a16d9c6-0712-4274-adea-bc1952d764b9 | arn:aws:sqs:ap-south-1:601422970468:test_queue | Confirmed | SQS |

Buttons for 'Edit', 'Delete', 'Request confirmation', 'Confirm subscription', and 'Create subscription' are located at the top right of the table. A search bar and navigation controls are also present.

Figure 6.18 SNS Notification Subscriptions

| Subscription: ada7d270-4b69-4ebb-b2b9-cb2be3785599 | |
|---|--|
| Edit Delete | |
| 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 | |
|---|--|
| Edit Delete | |
| 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 utilized to track account activity and API calls, as most cloud providers offer their services via APIs this is a very important service. These feeds from CloudTrail can also be integrated with AWS CloudWatch in order to create metrics for employing alarms for any suspicious account's behavior or any miscellaneous 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.

The screenshot shows the AWS CloudTrail Details page. At the top, there is a navigation bar with the AWS logo, a search bar, and user information (Mumbai, jainamvshah). Below the navigation bar, the URL is shown as CloudTrail > Dashboard > arn:aws:cloudtrail:ap-south-1:601422970468:trail/demo2.1. The main content area displays the 'General details' for the trail 'demo2.1'. The table contains the following information:

| General details | | | |
|---|---|--|--|
| Trail logging Logging | Trail log location demo211/AWSLogs/601422970468 | Log file validation Enabled | SNS notification delivery arn:aws:sns:ap-south-1:601422970468:aws-cloudtrail-logs-601422970468-d5e8a5a9 |
| Trail name demo2.1 | Last log file delivered March 10, 2022, 11:12:16 (UTC+05:30) | Last file validation delivered March 10, 2022, 11:07:02 (UTC+05:30) | Last SNS notification March 10, 2022, 11:12:16 (UTC+05:30) |
| Multi-region trail Yes | Log file SSE-KMS encryption Not enabled | | |
| Apply trail to my organization Not enabled | | | |

At the bottom right of the table, there are 'Delete', 'Stop logging', and 'Edit' buttons.

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.

| Name | Type | Last modified | Size | Storage class |
|---------------|--------|-------------------------------------|----------|---------------|
| AWSLogs/ | Folder | - | - | - |
| IBM_Demo.docx | docx | March 9, 2022, 21:39:35 (UTC+05:30) | 689.1 KB | Standard |
| ibmpro.csv | csv | March 9, 2022, 21:39:36 (UTC+05:30) | 41.0 B | Standard |
| notes.txt | txt | March 9, 2022, 21:39:37 (UTC+05:30) | 163.0 B | Standard |

Figure 6.21 S3 Bucket with CloudTrail Logs

| Name | Type | Last modified | Size | Storage class |
|---|------|--------------------------------------|---------|---------------|
| 601422970468_CloudTrail_us-west-2_20220309T1955Z_OrP6edVHmlANBkxv.json.gz | gz | March 10, 2022, 01:28:20 (UTC+05:30) | 902.0 B | Standard |
| 601422970468_CloudTrail_us-west-2_20220309T1950Z_iWcOQ1tcAk7iee6Z.json.gz | gz | March 10, 2022, 01:20:40 (UTC+05:30) | 903.0 B | Standard |
| 601422970468_CloudTrail_us-west-2_20220309T1945Z_KVU3M7J1ZpYpxbkr.json.gz | gz | March 10, 2022, 01:13:59 (UTC+05:30) | 1.0 KB | Standard |
| 601422970468_CloudTrail_us-west-2_20220309T1930Z_q3iHw64lwQ2HaDrX.json.gz | gz | March 10, 2022, 00:57:49 (UTC+05:30) | 1.0 KB | Standard |
| 601422970468_CloudTrail_us-west- | | March 10, 2022, 01:04:19 | | |

Figure 6.22 Data Log Files

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

- Detect any third-party AWS console logins from unknown places or countries.
- In most cases, companies/organizations do transfer logs to their own data center for long term storage, it's crucial to generate the logs in a text-like format such as JSON, for better understanding of complex data, from a test, notice AWS uses this concept in their generated logs and flows.

- Anchorage the storage API i.e. s3 API to import cloud trails to a search and arranging platform or security management systems like (SIEM solution) for creating more secured and robust use cases monitoring.

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 escorting cloud logs into a single point where they can be combined with on premises security events, thus enabling the investigator to have a single view of screen from which he/she can monitor the whole security strata.

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.

| Name | Key ID | Autodiscovered IAM Role | Region Category | Inputs | Actions |
|-------|----------------------|-------------------------|-----------------|--------|---|
| user1 | AKIAYYB42HZSEAT5WT50 | No | Global | 0 | Edit Clone Delete |

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.

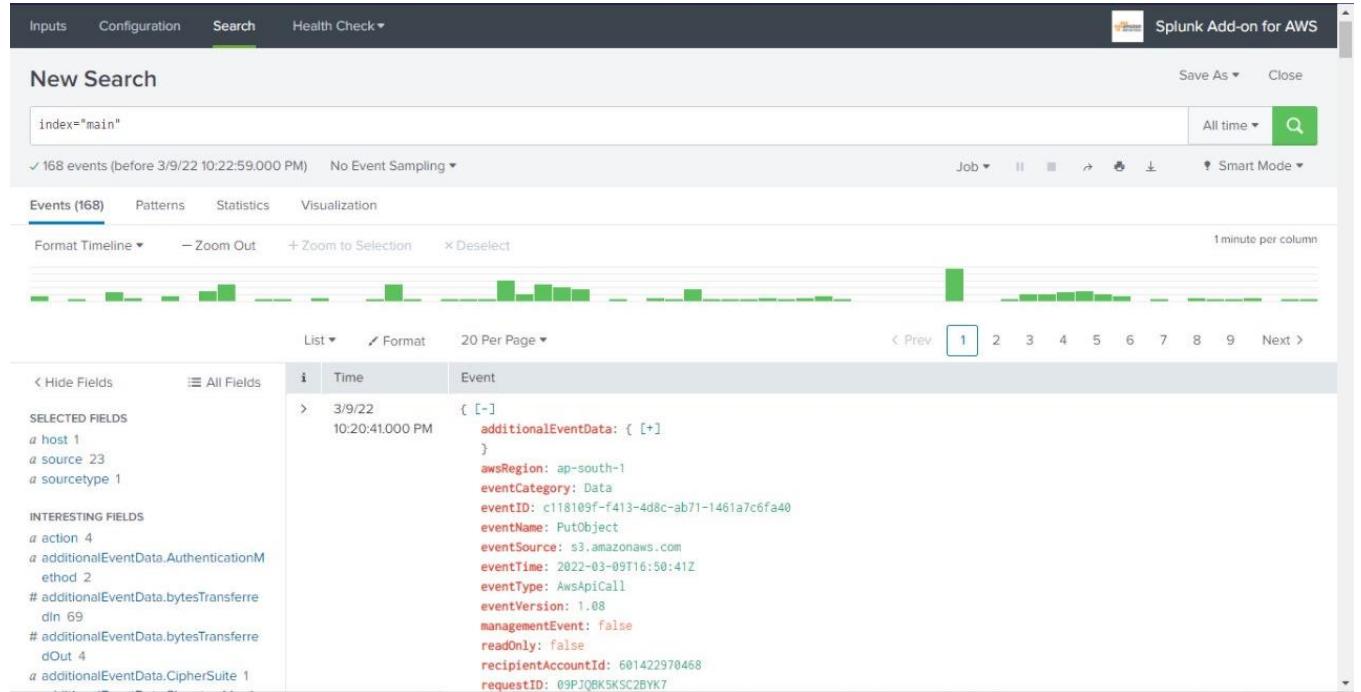


Figure 6.25 CloudTrail Logs Analysis

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, a glimpse of the created JSON Script is as follows: -

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "config:GetComplianceSummaryByConfigRule",
        "sns:DeleteMessage",
        "iam:GetAccountPasswordPolicy",
        "s3:GetLifecycleConfiguration",
        "s3:GetBucketTagging",
        "ec2:DescribeInstances",
        "ec2:DescribeAddresses",
        "s3:GetBucketLogging",
        "s3:PutBucketOwnershipControls",
        "ec2:DescribeRegions",
        "sns:ReceiveMessage",
        "s3:GetAccelerateConfiguration",
        "ec2:DescribeSnapshots",
        "elasticloadbalancing:DescribeLoadBalancers",
        "lambda:InvokeFunction"
      ],
      "Resource": "*"
    }
  ]
}

```

Figure 6.26 JSON Script

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.

The screenshot shows the AWS SQS console with the following details:

- Name:** test_queue
- Type:** Standard
- ARN:** arn:aws:sqs:ap-south-1:601422970468:test_queue
- Encryption:** Disabled
- URL:** https://sns.ap-south-1.amazonaws.com/601422970468/test_queue
- Dead-letter queue:** -

Figure 6.27 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.

The screenshot shows an Excel spreadsheet titled "demo - Excel" containing CloudTrail log data. The columns represent various parameters such as raw, time, action, additional, awsRegion, aws_account, changeType, command, dateHour, dateMday, dateMin, dateMonth, dateSec, dateMicrosecond, and authType. The data consists of multiple rows of log entries.

| raw | time | action | additional | awsRegion | aws_account | changeType | command | dateHour | dateMday | dateMin | dateMonth | dateSec | dateMicrosecond | authType |
|---|------|--------|------------------------|-----------|-------------|------------|------------|----------|----------|---------|-----------|---------|-----------------|----------|
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 2371 | | 0 2hwsSEAV-AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 50 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 670 | | 0 vVnm9v9f AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 49 | march | | | |
| ["eventV2 2022-03-01 success | | | AwsApiCall | ap-south- | 6.01E+11 | STS | AssumeRole | 16 | 9 | 47 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 2451 | | 0 cZWo08Cj AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 47 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 976 | | 0 oJMLzEb/ AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 46 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 1492 | | 0 x9jkRe4et AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 45 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 593 | | 0 ngB28kq AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 44 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 1006 | | 0 SV0krvww AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 44 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 3967 | | 0 RI/mYXrre AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 42 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 2217 | | 0 BjtEwWro AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 40 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 332 | | 0 oeErnDyQ1 AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 40 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 1023 | | 0 i427Xaxc/ AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 40 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 443 | | 0 tBa01dkD AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 39 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 902 | | 0 lkp9xIcc AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 39 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 443 | | 0 IQ29sCpK1 AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 39 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 443 | | 0 ZjcCysSeV AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 39 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 447 | | 0 FcfWivz16 AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 446 | | 0 lolq4TN/1 AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 449 | | 0 Ozl2mnW AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 445 | | 0 TYnPXR99 AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 334 | | 0 RyDyzh1t AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |
| ["eventV2 2022-03-01 modified AuthHead ECDHE-RS SSE_3 SigV4 | 336 | | 0 L6GxMWn AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 38 | march | | | |

Figure 6.28 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.

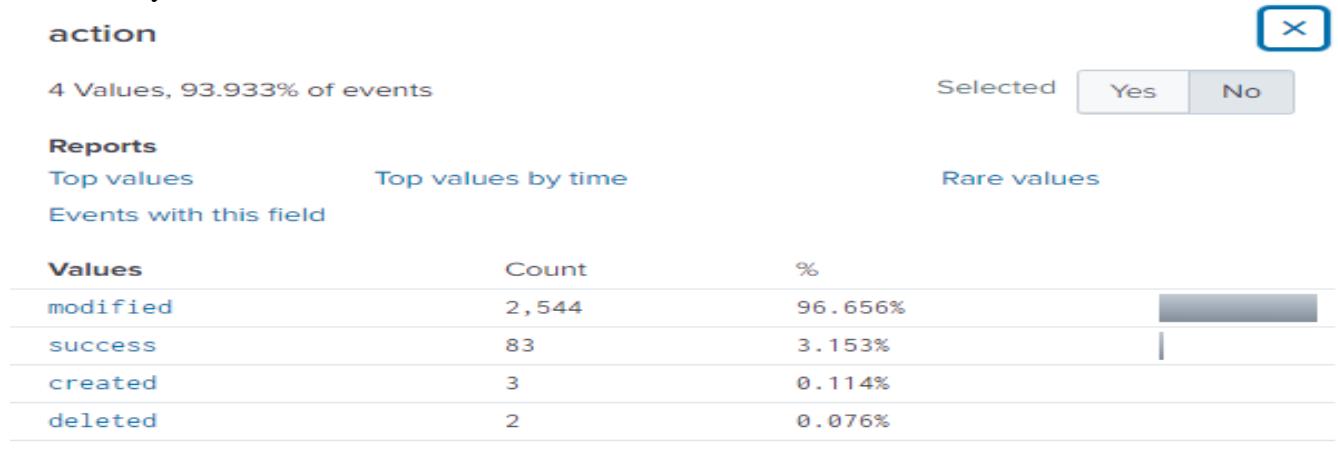


Figure 6.29 Action Field

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

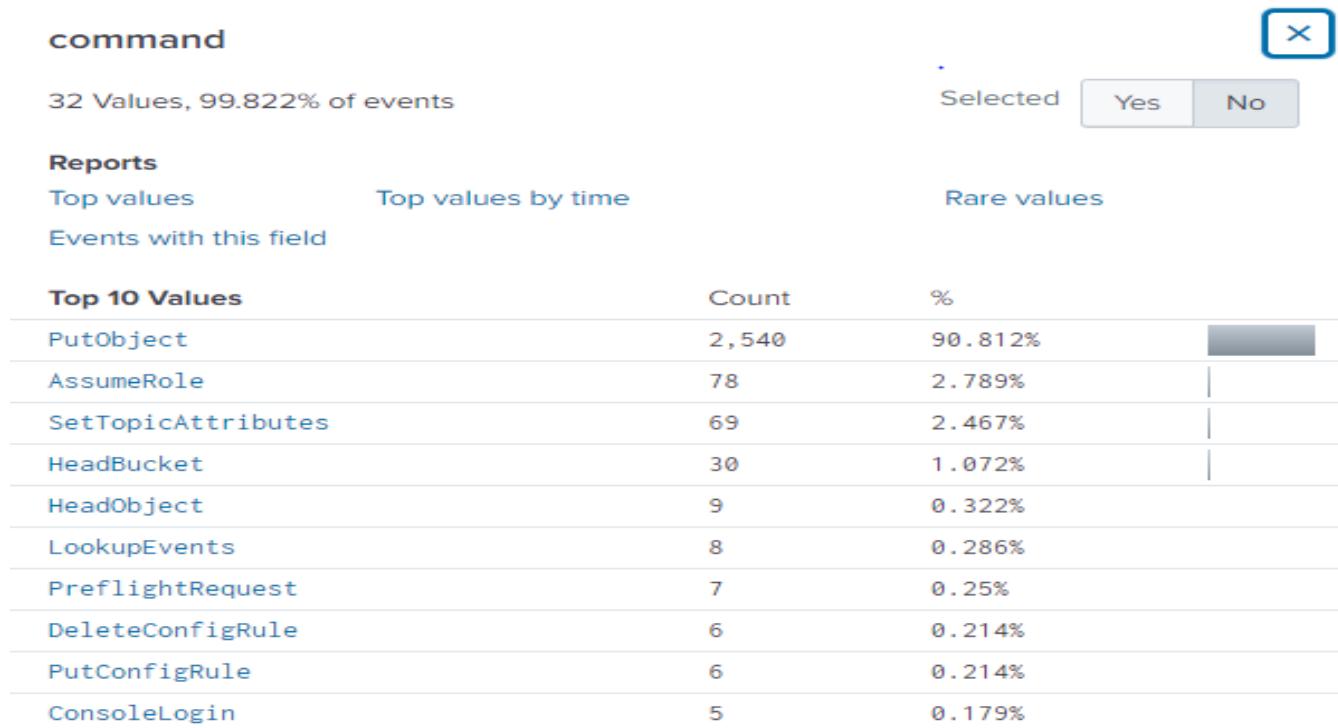


Figure 6.30 Command Field

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

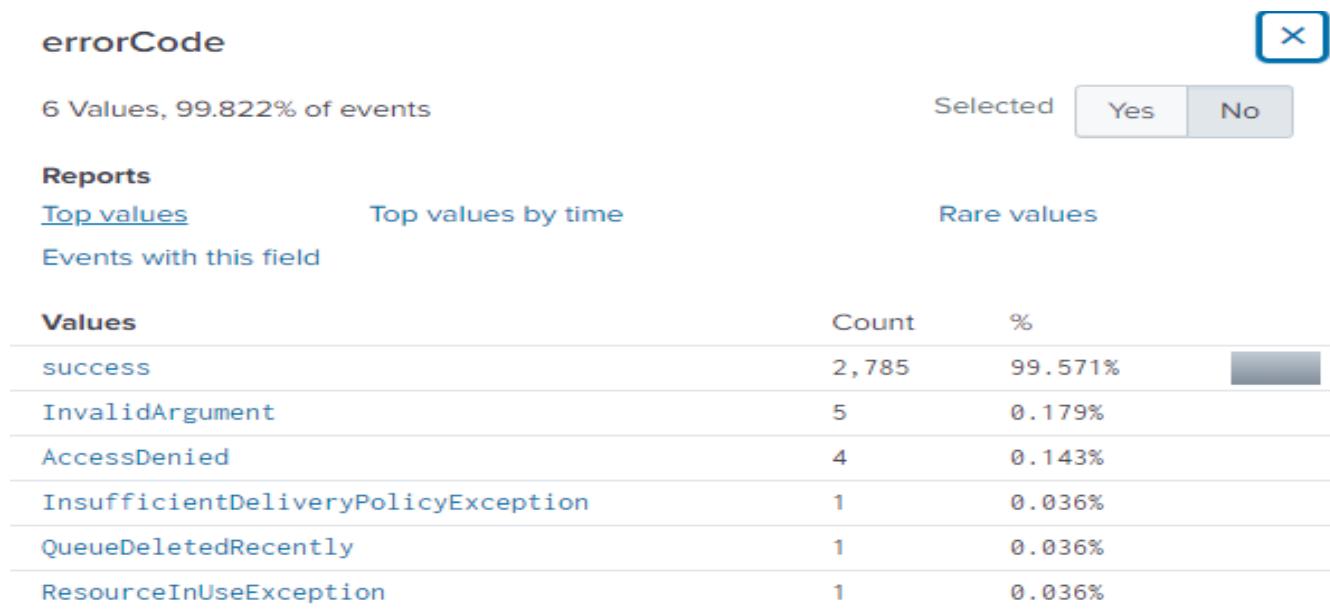


Figure 6.31 ErrorCode Field

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



Figure 6.32 Host Field

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

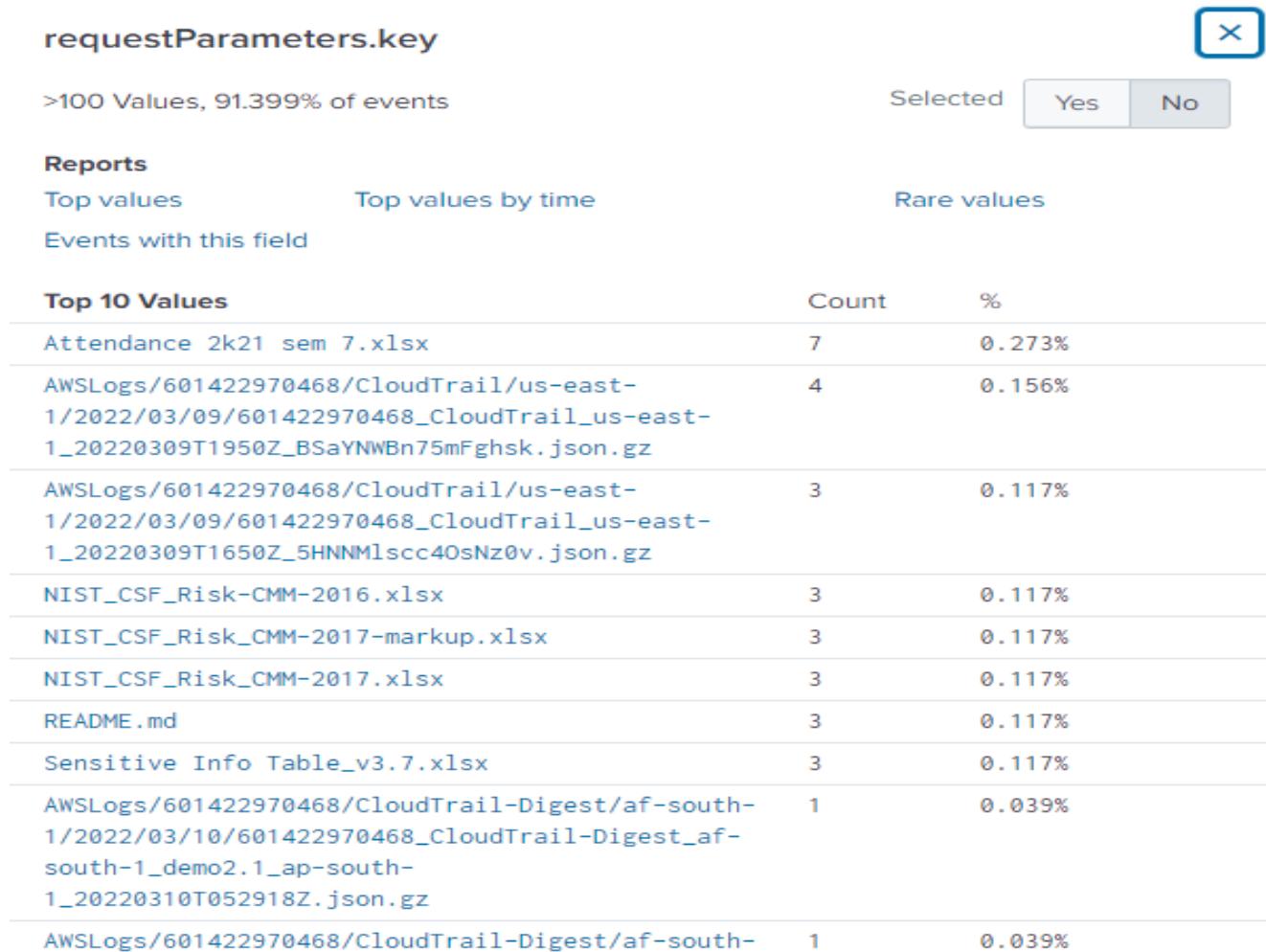


Figure 6.33 RequestParameterKey Field

6.4 Forensic Analysis in IaaS Cloud

Making use of cloud forensics to help organization in strengthening their incident response and threat detection capabilities, organizations must have proper forensics investigation tools to keep their cloud infrastructure secured in case of an attack, recognizing the signs of vulnerability as well as quickly locate an infection and its objectives before they have an impact on the organizations' important data.

If there is a case of a hacked virtual machine, most users terminate the virtual machine (VM), erasing all proof in the process. It would become very challenging to perform forensic analysis in such cases. Until now, there have been few tools and applications which monitor the system properly and gather data. When it comes to gathering and analyzing evidence, must look for the following:

- Network packet captures for forensics.
- Memory usage for a particular instance.
- Events and data logs

In order to provision a machine for forensic analysis, installing necessary forensic investigation tools is necessary in order to get insights. In order to implement this a package called SIFT has been utilized which provides access to most of the forensics tools from one executable package. The forensic machine for this mentioned scenario has been prepared in the following manner.

An EC2 instance called “cloudresearch-instance” is created and then after logging into it by doing SSH SIFT investigation tools are downloaded with the following commands.

```
ubuntu@ip-172-31-5-38:~$ sudo curl -Lo /usr/local/bin/sift https://github.com/sans-dfir/sift-cli/releases/download/v1.14.0-rc1/sift-cli-linux
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload Total Spent   Left Speed
100  147  100  147    0     0  630      0 --::-- --::-- --::--  630
100  651  100  651    0     0 1299      0 --::-- --::-- --::-- 1299
100 55.0M 100 55.0M    0     0 9211k      0 0:00:06 0:00:06 --::-- 9942k
ubuntu@ip-172-31-5-38:~$ sudo chmod 755 /usr/local/bin/sift
ubuntu@ip-172-31-5-38:~$ sudo sift install
> sift-cli@1.14.0-rc1+0-g0582d2b
> sift-version: notinstalled
```

Figure 6.34 SIFT Installation

After installing SIFT tools a snapshot is created for the instance to perform forensic analysis on it. After creating snapshot, a volume is created from that snapshot and then attached to the earlier created EC2 instance.

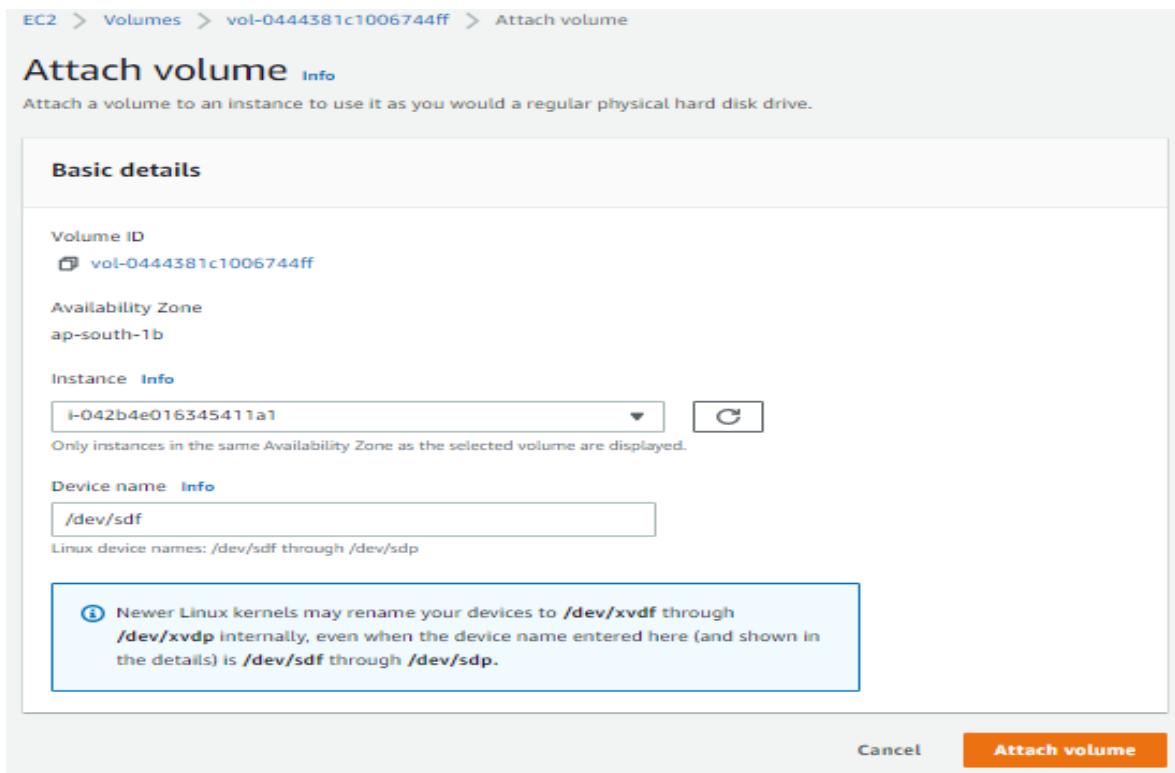


Figure 6.35 Attaching evidence volume to SIFT Workstation

Verifying evidence attached to a device using lsblk command.

```
ubuntu@ip-172-31-5-38:~$ sudo lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0      0 42.2M  1 loop /snap/snapd/14066
loop1    7:1      0 55.5M  1 loop /snap/core18/2253
loop2    7:2      0   25M  1 loop /snap/amazon-ssm-agent/4046
xvda   202:0     0   30G  0 disk 
└─xvda1 202:1     0   30G  0 part /
xvdf   202:80    0   30G  0 disk 
└─xvdf1 202:81    0   30G  0 part
```

Figure 6.36 Evidence Attached Verification

Using the file command to determine the format of the partition as shown below and also a directory has been made to mount the evidentiary Linux file system as read-only:

```
ubuntu@ip-172-31-5-38:~$ sudo file -s /dev/xvdf1
/dev/xvdf1: Linux rev 1.0 ext4 filesystem data, UUID=c1ce24a2-4987-4450-ae15-62eb028ff1cd, volume name "cloudimg-rootfs" (needs journal recovery)
extents) (64bit) (large files) (huge files)
ubuntu@ip-172-31-5-38:~$ sudo mkdir /mnt/linux_mount
ubuntu@ip-172-31-5-38:~$ mount -o ro /dev/xvdf1 /mnt/linux_mount/
mount: only root can use "--options" option
ubuntu@ip-172-31-5-38:~$ sudo mount -o ro /dev/xvdf1 /mnt/linux_mount/
ubuntu@ip-172-31-5-38:~$ sudo mount | grep "/mnt"
/dev/xvdf1 on /mnt/linux_mount type ext4 (ro,relatime)
```

Figure 6.37 Mounting evidentiary file on the system

Verifying the mounted data.

```
ubuntu@ip-172-31-5-38:~$ sudo ls -als /mnt/linux_mount/
total 124
4 drwxr-xr-x  24 root  root  4096 Apr  4 06:52 .
4 drwxr-xr-x  18 root  root  4096 Apr  4 07:11 ..
4 drwxr-xr-x   2 root  root  4096 Apr  4 06:41 bin
4 drwxr-xr-x   3 root  root  4096 Apr  4 06:49 boot
4 drwxrwxr-x   2 ubuntu root  4096 Apr  4 06:52 cases
4 drwxr-xr-x   4 root  root  4096 Nov 29 17:32 dev
12 drwxr-xr-x 155 root  root 12288 Apr  4 06:54 etc
4 drwxr-xr-x   3 root  root  4096 Apr  4 06:10 home
0 lrwxrwxrwx   1 root  root   30 Nov 29 17:39 initrd.img -> boot/initrd.img-5.4.0-1060-aws
0 lrwxrwxrwx   1 root  root   30 Nov 29 17:39 initrd.img.old -> boot/initrd.img-5.4.0-1060-aws
4 drwxr-xr-x  22 root  root  4096 Apr  4 06:16 lib
4 drwxr-xr-x   2 root  root  4096 Apr  4 06:16 lib64
16 drwx-----  2 root  root 16384 Nov 29 17:34 lost+found
4 drwxr-xr-x   2 root  root  4096 Nov 29 17:27 media
4 drwxr-xr-x  17 root  root  4096 Apr  4 06:52 mnt
4 drwxr-xr-x   4 root  root  4096 Apr  4 06:36 opt
4 drwxr-xr-x   2 root  root  4096 Apr 24 2018 proc
4 drwx-----  5 root  root  4096 Apr  4 06:52 root
4 drwxr-xr-x   5 root  root  4096 Nov 29 17:39 run
12 drwxr-xr-x   2 root  root 12288 Apr  4 06:48 sbin
4 drwxr-xr-x   6 root  root  4096 Apr  4 06:10 snap
4 drwxr-xr-x   2 root  root  4096 Nov 29 17:27 srv
4 drwxr-xr-x   2 root  root  4096 Apr 24 2018 sys
4 drwxrwxrwt  18 root  root  4096 Apr  4 06:56 tmp
4 drwxr-xr-x  12 root  root  4096 Apr  4 06:18 usr
4 drwxr-xr-x  14 root  root  4096 Apr  4 06:14 var
0 lrwxrwxrwx   1 root  root   27 Nov 29 17:39 vmlinuz -> boot/vmlinuz-5.4.0-1060-aws
0 lrwxrwxrwx   1 root  root   27 Nov 29 17:39 vmlinuz.old -> boot/vmlinuz-5.4.0-1060-aws
```

Figure 6.38 Listing data of mounted directory

After the evidence is attached to the SIFT Workstation, the initial step is to chisel data from the unallocated space and segregate out the files that are known to be good.

Another EC2 Instance is launched and based on the AMI and another snapshot is created and a volume is attached from the snapshot in the same availability zone as the SIFT Workstation. A different name called “HASH-BASELINE” for both the snapshot and the volume is assigned so that it is easy to differentiate these objects and the SIFT Workstation itself. Using the same steps as above the volume is attached and mounted as the 3rd volume on the SIFT Workstation which is named as /mnt/linux_base.

The screenshot shows the AWS EC2 Volumes page. A green success message at the top says "Successfully attached volume vol-02c60de7e3a2b28b2 to instance i-042b4e016345411a1.". The main table lists four volumes:

| Name | Volume ID | Type | Size | IOPS | Throughput | Snapshot | Created |
|-------------------|-----------------------|------|--------|------|------------|-----------------|----------|
| cloudresearch-... | vol-07ae0dc02e80314d7 | gp2 | 30 GiB | 100 | - | snap-01559e0... | 2022/04/ |
| - | vol-0444381c1006744ff | gp2 | 30 GiB | 100 | - | snap-045fa65... | 2022/04/ |
| HASH-BASELINE | vol-02c60de7e3a2b28b2 | gp2 | 30 GiB | 100 | - | snap-06220b3... | 2022/04/ |
| cloudresearch-... | vol-0d456a51127c71684 | gp2 | 30 GiB | 100 | - | snap-01559e0... | 2022/04/ |

Figure 6.39 Newly Attached Volume to the instance

```
ubuntu@ip-172-31-5-38:~$ sudo mkdir /mnt/linux_base
ubuntu@ip-172-31-5-38:~$ sudo mount -o ro /dev/xvdg1 /mnt/linux_base/
ubuntu@ip-172-31-5-38:~$ sudo lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 42.2M  1 loop /snap/snapd/14066
loop1    7:1    0 55.5M  1 loop /snap/core18/2253
loop2    7:2    0   25M  1 loop /snap/amazon-ssm-agent/4046
xvda   202:0    0   30G  0 disk 
└─xvda1 202:1    0   30G  0 part /
xvdf   202:80   0   30G  0 disk 
└─xvdf1 202:81   0   30G  0 part /mnt/linux_mount
xvdg   202:96   0   30G  0 disk 
└─xvdg1 202:97   0   30G  0 part /mnt/linux_base
ubuntu@ip-172-31-5-38:~$ |
```

Figure 6.40 Attaching and verifying additional mounted volume on SIFT Workstation

A hash database of all files on the reference volume is created using hfind which is called "known_files.md5" and in order to identify which files are new or modified an another hast list of files is created for the volume under investigation, this is called "investigate_files.md5" and with that list an names additional list of files that are new or changed are stored in "changed_files.txt".

Then in order to search for known indicators of compromise for the server instance an IOC scanner called “Loki” is installed which detects indicators of compromise Detection is based on four detection methods:

- File Names of IOC- which matches regular expressions of file names
- Yara Rule Check-matches signature of data and processes memory
- Hash Check – it compares harmful hashes like MD5, SHA1, SHA256 with scanned files

The following results were obtained when Loki detected indicators of compromise.

```
[NOTICE]
FILE: /mnt/linux_mount/usr/sbin/dsniff SCORE: 55 TYPE: ELF SIZE: 76496
FIRST_BYTES: 7f454c460201010000000000000000003003e00 / <filter object at 0x7f156823e160>
MD5: 7bae63a8ad173e4388c71aa9dddb170f
SHA1: a59fcfa06ee629203bd70c7999208c50a14e3762
SHA256: cfe3941ce302f6dcfb13ee5c2780f900bdef009a068c65a5bbcfc19ac9f15be CREATED: Mon Apr 4 06:19:12 2022 MODIFIED: Fri Jul 21 18:05:34 2017 ACCES
SED: Mon Apr 4 06:19:12 2022
REASON_1: Yara Rule MATCH: HkTL_Dsniff SUBSCORE: 55
DESCRIPTION: Detects Dsniff hack tool REF: https://goo.gl/eFoP4A AUTHOR: Florian Roth
MATCHES: Str1: .*account.*|.*acct.*|.*domain.*|.*login.*|.*member.*

[NOTICE]
FILE: /mnt/linux_mount/usr/sbin/webmitr SCORE: 55 TYPE: ELF SIZE: 30720
FIRST_BYTES: 7f454c460201010000000000000000003003e00 / <filter object at 0x7f156823e128>
MD5: 53332776a1abaccef056e5f8abf22762
SHA1: 97730171ad257fcaa7a83bf118d2b294bcd16205
SHA256: 777b361598119307cf92d6ddb6f272d8df7ccb8ca927352e4581246af81a6aff CREATED: Mon Apr 4 06:19:12 2022 MODIFIED: Fri Jul 21 18:05:34 2017 ACCES
SED: Mon Apr 4 06:19:12 2022
REASON_1: Yara Rule MATCH: HKTL_Dsniff SUBSCORE: 55
DESCRIPTION: Detects Dsniff hack tool REF: https://goo.gl/eFoP4A AUTHOR: Florian Roth
MATCHES: Str1: .*account.*|.*acct.*|.*domain.*|.*login.*|.*member.*
```

Figure 6.41 Notice for indicators of compromise

```
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/mscoree.dll SCORE: 70 TYPE: EXE SIZE: 5236
FIRST_BYTES: 4d5a400001000000600000fffff0000b8000000 / <filter object at 0x7f1565d38668>
MD5: 8cb5ae3dab7578de39fa36cbe260f21f
SHA1: b726aab0531eacc130809a5e9bfd94ebad03c1e8
SHA256: a14c0edb326fd24c220036f806730c0db359adcf5a7e41d9f5a0b7faab8aa8 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCES
SED: Mon Apr 4 06:43:36 2022
REASON_1: File Name IOC matched PATTERN: /mscoree\.dll SUBSCORE: 70 DESC: Unattributed Shadowpad Activity in Exchange Exploitation IOC https://www.welivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/rundll32.exe SCORE: 60 TYPE: EXE SIZE: 1032
FIRST_BYTES: 4d5a400001000000600000fffff0000b8000000 / <filter object at 0x7f1565d38630>
MD5: 35f92c16dcc3beb49f3142bcd2874d1
SHA1: b0732939192a4e9ac448886896f3d7abf50c4a6
SHA256: b90af2992f8ef634ac07041695b5d790b167c15de737914aeeffe69c3a4ddebf3f CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_rundll32_exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of rundll32.exe REF: - AUTHOR: Florian Roth
[ALERT]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/svchost.exe SCORE: 115 TYPE: EXE SIZE: 1032
FIRST_BYTES: 4d5a400001000000600000fffff0000b8000000 / <filter object at 0x7f1565d38550>
MD5: 7c20774d170cc400a78de22fad2d59ce
SHA1: a294a9e485c37d89bf68b9571808b0994ea260d
SHA256: 556d962a414c1abaf3b7b6a64017e08e69fb6efb447ddb71b38fad755cdb3b68 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_svchost_exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of svchost.exe REF: - AUTHOR: Florian Roth
REASON_2: Yara Rule MATCH: svchost_ANOMALY SUBSCORE: 55
DESCRIPTION: Abnormal svchost.exe - typical strings not found in file REF: - AUTHOR: Florian Roth
```

Figure 6.42 Warnings and Alerts for Compromise of Indicators 1

```

WARNING] FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/mscoree.dll SCORE: 70 TYPE: ELF SIZE: 254160
FIRST_BYTES: 7f454c46020101000000000000000000003003e0 / <filter object at 0x7f1567173b38>
MD5: b5f52a3df13f58d5279996f3dcda71
SHA1: 7ca4a01ff1df04fe92dc4929c95ca52555917e5
SHA256: 994ae08459d849d5e7ada92d47613e148721cb25ae1ad1b7b785de083785393 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: File Name IOC matched PATTERN: /mscoree.dll SUBSCORE: 70 DESC: Unattributed Shadowpad Activity in Exchange Exploitation IOC https://www.welivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/spoolsv.exe SCORE: 60 TYPE: EXE SIZE: 1032
FIRST_BYTES: 4d5a40000100000006000000ffff0000b8000000 / <filter object at 0x7f1565d38550>
MD5: c20774d170c400a78de22fad5d9ce
SHA1: a294a9e48537d89bf68b9571808b0994ea260d
SHA256: 556d962a414c1abaf3b7b6a64017e08e69fb6efb447ddb71b38fad755cd3b68 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_spoolsv_exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of spoolsv.exe REF: - AUTHOR: Florian Roth
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/conhost.exe SCORE: 70 TYPE: EXE SIZE: 2484
FIRST_BYTES: 4d5a40000100000006000000ffff0000b8000000 / <filter object at 0x7f1565d385c0>
MD5: d50ecfc13fba9fb825bde1f32d25403a
SHA1: 6a52d45fe24897300d94a2554cfef94d769aed2
SHA256: a437953c7fe8af92491f940bfba78cc0b71eb451cc6b74b71014698a0204 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: conhost_ANOMALY SUBSCORE: 70
DESCRIPTION: Anomaly rule looking for certain strings in a system file (maybe false positive on certain systems) - file conhost.exe REF: not set AU
THOR: Florian Roth
[ALERT]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/explorer.exe SCORE: 115 TYPE: EXE SIZE: 6616
FIRST_BYTES: 4d5a40000100000006000000ffff0000b8000000 / <filter object at 0x7f1565d38518>
MD5: a3fd0188117097609497863450d44f6
SHA1: 9dc49ed8a9ca0ffed5e55d4028915aaa37c
SHA256: e6b05533c6e315d29673f9373d450df856e7dd9f837a426fa7fad597e49 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_explorer_exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of explorer.exe REF: - AUTHOR: Florian Roth
REASON_2: Yara Rule MATCH: explorer_ANOMALY SUBSCORE: 55
DESCRIPTION: Abnormal explorer.exe - typical strings not found in file REF: - AUTHOR: Florian Roth
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/Fakedlls/mscoree.dll SCORE: 70 TYPE: EXE SIZE: 5236
FIRST_BYTES: 4d5a40000100000006000000ffff0000b8000000 / <filter object at 0x7f1565d38668>
MD5: b726aab053140cc3980945e90fd94ebad03c1e8
SHA1: b10d53dcece179109aff61b6e6ccaa65be816f3c4
SHA256: d1a473a0dd813bd3565b810dch8ff8bc7907478a994c564d5200925894e0d32 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: File Name IOC matched PATTERN: /mscoree.dll SUBSCORE: 70 DESC: Unattributed Shadowpad Activity in Exchange Exploitation IOC https://www.welivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/Fakedlls/rundll32.exe SCORE: 60 TYPE: EXE SIZE: 1032

```

Figure 6.43 Warnings and Alerts for Compromise of Indicators 2

```

NOTICE] FILE: /mnt/linux_mount/usr/local/lib/python3.6/dist-packages/pip/_vendor/distlib/w64.exe SCORE: 50 TYPE: EXE SIZE: 99840
FIRST_BYTES: 4d5a400003000000400000ffff0000b8000000 / <filter object at 0x7f1567173c50>
MD5: 0655a0af4a2ff9bf591f614ba8f5721f
SHA1: b10d53dcece179109aff61b6e6ccaa65be816f3c4
SHA256: d1a473a0dd813bd3565b810dch8ff8bc7907478a994c564d5200925894e0d32 CREATED: Mon Apr 4 06:48:14 2022 MODIFIED: Mon Apr 4 06:48:13 2022 ACCESSED: Mon Apr 4 06:48:14 2022
REASON_1: File Name IOC matched PATTERN: /w64.exe SUBSCORE: 50 DESC: Cred Dumping
NOTICE] Results: 2 alerts, 10 warnings, 11 notices
RESULTS] Indicators detected
RESULTS] Loki recommends checking the elements on virustotal.com or Google and triage with a professional tool like THOR https://nextron-systems.com/thor in corporate networks.
INFO] Please report false positives via https://github.com/Neo23x0/signature-base
NOTICE] Finished LOKI Scan SYSTEM: swiftworkstation TIME: 20220404T10:16:13Z
ubuntu@ip-172-31-5-38:/tmp/Loki-0.44.25 |

```

Figure 6.44 Final Results of Loki

6.4.1 Additional Forensic Analysis

Some malware or anomaly makes use of the start-up scripts that are initiated when the system is started and runs at boot time. On some distributions, these are found in /etc/init.d, but on Amazon Linux and Red Hat variants, the scripts will be in /etc/rc*.d.

```
*ubuntu@ip-172-31-5-38:~$ ls -als -t /mnt/linux_mount/etc/rc*.d/
/mnt/linux_mount/etc/rc0.d:
total 16
12 drwxr-xr-x 155 root root 12288 Apr  4 06:54 ..
4 drwxr-xr-x  2 root root  4096 Apr  4 06:43 .
0 lwxrwxrwxrwx  1 root root   17 Apr  4 06:43 K01winbind -> ../init.d/winbind
0 lwxrwxrwxrwx  1 root root   15 Apr  4 06:42 K01saned -> ../init.d/saned
0 lwxrwxrwxrwx  1 root root   22 Apr  4 06:42 K01avahi-daemon -> ../init.d/avahi-daemon
0 lwxrwxrwxrwx  1 root root   19 Apr  4 06:42 K01bluetooth -> ../init.d/bluetooth
0 lwxrwxrwxrwx  1 root root   18 Apr  4 06:40 K01stunnel4 -> ../init.d/stunnel4
0 lwxrwxrwxrwx  1 root root   21 Apr  4 06:39 K01samba-ad-dc -> ../init.d/samba-ad-dc
0 lwxrwxrwxrwx  1 root root   14 Apr  4 06:39 K01nmbd -> ../init.d/nmbd
0 lwxrwxrwxrwx  1 root root   14 Apr  4 06:39 K01smbd -> ../init.d/smbd
0 lwxrwxrwxrwx  1 root root   27 Apr  4 06:35 K01speech-dispatcher -> ../init.d/speech-dispatcher
0 lwxrwxrwxrwx  1 root root   16 Apr  4 06:32 K01nfdump -> ../init.d/nfdump
0 lwxrwxrwxrwx  1 root root   20 Apr  4 06:32 K01nbd-client -> ../init.d/nbd-client
0 lwxrwxrwxrwx  1 root root   16 Apr  4 06:18 K01docker -> ../init.d/docker
0 lwxrwxrwxrwx  1 root root   26 Apr  4 06:17 K01clamav-freshclam -> ../init.d/clamav-freshclam
0 lwxrwxrwxrwx  1 root root   29 Apr  4 06:14 K01apache-htcacheclean -> ../init.d/apache-htcacheclean
0 lwxrwxrwxrwx  1 root root   17 Apr  4 06:14 K01apache2 -> ../init.d/apache2
0 lwxrwxrwxrwx  1 root root   23 Nov 29 17:31 K01lvm2-lvmpolld -> ../init.d/lvm2-lvmpolld
0 lwxrwxrwxrwx  1 root root   22 Nov 29 17:31 K01lvm2-lvmetad -> ../init.d/lvm2-lvmetad
0 lwxrwxrwxrwx  1 root root   13 Nov 29 17:31 K01lxd -> ../init.d/lxd
0 lwxrwxrwxrwx  1 root root   23 Nov 29 17:31 K01open-vm-tools -> ../init.d/open-vm-tools
0 lwxrwxrwxrwx  1 root root   18 Nov 29 17:31 K01plymouth -> ../init.d/plymouth
0 lwxrwxrwxrwx  1 root root   20 Nov 29 17:31 K01cryptdisks -> ../init.d/cryptdisks
0 lwxrwxrwxrwx  1 root root   26 Nov 29 17:31 K01cryptdisks-early -> ../init.d/cryptdisks-early
0 lwxrwxrwxrwx  1 root root   20 Nov 29 17:31 K01irqbalance -> ../init.d/irqbalance
0 lwxrwxrwxrwx  1 root root   15 Nov 29 17:31 K01lxcsfs -> ../init.d/lxcsfs
0 lwxrwxrwxrwx  1 root root   29 Nov 29 17:31 K01unattended-upgrades -> ../init.d/unattended-upgrades
0 lwxrwxrwxrwx  1 root root   18 Nov 29 17:31 K01ebtables -> ../init.d/ebtables
0 lwxrwxrwxrwx  1 root root   15 Nov 29 17:31 K01uuid -> ../init.d/uuid
0 lwxrwxrwxrwx  1 root root   15 Nov 29 17:31 K01mdadm -> ../init.d/mdadm
0 lwxrwxrwxrwx  1 root root   24 Nov 29 17:31 K01mdadm-waitidle -> ../init.d/mdadm-waitidle
0 lwxrwxrwxrwx  1 root root   20 Nov 29 17:31 K01open-iscsi -> ../init.d/open-iscsi
0 lwxrwxrwxrwx  1 root root   16 Nov 29 17:31 K01iscsid -> ../init.d/iscsid
0 lwxrwxrwxrwx  1 root root   13 Nov 29 17:31 K01atd -> ../init.d/atd
0 lwxrwxrwxrwx  1 root root   17 Nov 29 17:27 K01rsyslog -> ../init.d/rsyslog
```

Figure 6.45 Startup Scripts

Looking for unusual files can be a hectic task, so in order to make it easy a security expert looks for SUID and SGID files (SUID Files - SUID is a special file permission for executable files which enables other users to run the file with effective permissions of the file owner while SGID Files - SGID is a special file permission that also applies to executable files and enables other users to inherit the effective GID of file group owner).The following commands perform the comparison on mounted volume for evidence capturing.

```
ubuntu@ip-172-31-5-38:~$ sudo find /mnt/linux_mount/ -uid 0 -perm -4000 -print > suid_evidence
ubuntu@ip-172-31-5-38:~$ sudo
usage: sudo -h | -K | -k | -v
usage: sudo [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknP$] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...
ubuntu@ip-172-31-5-38:~$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknP$] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...
ubuntu@ip-172-31-5-38:~$ sudo find /mnt/linux_base/ -uid 0 -perm -4000 -print > suid_base
ubuntu@ip-172-31-5-38:~$ sudo cut suid_base -d"/" -f4- > suid_base_relative
ubuntu@ip-172-31-5-38:~$ sudo cut suid_base -d"/" -f4- > suid_evidence_relative
ubuntu@ip-172-31-5-38:~$ sudo diff suid_base_relative suid_evidence_relative
ubuntu@ip-172-31-5-38:~$ ls
Desktop  changed_files.txt  investigate_files.md5-md5.idx  known_files.md5          known_files.md5-md5.idx2  output      suid_base_relative  suid_evidence_relative
changed.md5  investigate_files.mds  investigate_files.md5-md5.idx2  known_files.md5-md5.idx  1oki_0.44.2.zip      suid_base  suid_evidence
ubuntu@ip-172-31-5-38:~$ echo suid_base_relative
suid_base_relative
ubuntu@ip-172-31-5-38:~$ vim suid_base_relative
ubuntu@ip-172-31-5-38:~$ vim suid_base_relative
ubuntu@ip-172-31-5-38:~$
```

Figure 6.46 Commands to look for unusual files

```
ubuntu@ip-172-31-5-38: ~
```

```
bin/mount
bin/fusermount
bin/umount
bin/ping
bin/su
usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
usr/lib/eject/dmcrypt-get-device
usr/lib/snapd/snap-confine
usr/lib/polkit-1/polkit-agent-helper-1
usr/lib/dbus-1.0/dbus-daemon-launch-helper
usr/lib/openssh/ssh-keysign
usr/bin/chsh
usr/bin/chfn
usr/bin/sudo
usr/bin/newgrp
usr/bin/traceroute6.iputils
usr/bin/newuidmap
usr/bin/passwd
usr/bin/gpasswd
usr/bin/pkexec
usr/bin/newgidmap
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

Figure 6.47 List of Unusual files

In order to look for files with high entropy there is a tool in SIFT called DensityScout which detects packing, compression, and encrypted files that exceed a “density” threshold. The following commands are implemented in order to find such files which exceed the threshold.

```
ubuntu@ip-172-31-5-38:~$ sudo densityscout -r -p 0.1 -l 0.1 -o high_density_evidence.txt /mnt/linux_mount/
DensityScout (Build 45)
by Christian Wojner

Calculating density for file ...
(0.08665) | /mnt/linux_mount/usr/share/man/man1/gawk.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/x86_64-linux-gnu-gcc-7.1.gz
(0.07464) | /mnt/linux_mount/usr/share/man/man1/wget.1.gz
(0.08366) | /mnt/linux_mount/usr/share/man/man1/socat.1.gz
(0.05668) | /mnt/linux_mount/usr/share/man/man1/xterm.1.gz
(0.09947) | /mnt/linux_mount/usr/share/man/man1/less.1.gz
(0.09091) | /mnt/linux_mount/usr/share/man/man1/sh.distrib.1.gz
(0.09165) | /mnt/linux_mount/usr/share/man/man1/keytool.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/g++-7.1.gz
(0.09927) | /mnt/linux_mount/usr/share/man/man1/git-fast-import.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/g++.1.gz
(0.07959) | /mnt/linux_mount/usr/share/man/man1/x86_64-linux-gnu-ld.bfd.1.gz
(0.07728) | /mnt/linux_mount/usr/share/man/man1/gpg.1.gz
(0.09091) | /mnt/linux_mount/usr/share/man/man1/sh.1.gz
(0.07959) | /mnt/linux_mount/usr/share/man/man1/ld.1.gz
(0.08236) | /mnt/linux_mount/usr/share/man/man1/git-log.1.gz
(0.05229) | /mnt/linux_mount/usr/share/man/man1/bash.1.gz
(0.08055) | /mnt/linux_mount/usr/share/man/man1/cli.1.gz
(0.09974) | /mnt/linux_mount/usr/share/man/man1/find.1.gz
```

Figure 6.48 High Entropy files in /mnt/linux_mount (Volume where SIFT is installed)

```
ubuntu@ip-172-31-5-38:~$ sudo densityscout -r -p 0.1 -l 0.1 -o high_density_base.txt /mnt/linux_base/
DensityScout (Build 45)
by Christian Wojner

Calculating density for file ...
(0.08665) | /mnt/linux_base/usr/share/man/man1/gawk.1.gz
(0.07464) | /mnt/linux_base/usr/share/man/man1/wget.1.gz
(0.09947) | /mnt/linux_base/usr/share/man/man1/less.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/sh.distrib.1.gz
(0.09927) | /mnt/linux_base/usr/share/man/man1/git-fast-import.1.gz
(0.07728) | /mnt/linux_base/usr/share/man/man1/gpg.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/sh.1.gz
(0.08236) | /mnt/linux_base/usr/share/man/man1/git-log.1.gz
(0.05229) | /mnt/linux_base/usr/share/man/man1/bash.1.gz
(0.09974) | /mnt/linux_base/usr/share/man/man1/find.1.gz
(0.09947) | /mnt/linux_base/usr/share/man/man1/pager.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/dash.1.gz
(0.08665) | /mnt/linux_base/usr/share/man/man1/nawk.1.gz
(0.07303) | /mnt/linux_base/usr/share/man/man1/rsync.1.gz
(0.07669) | /mnt/linux_base/usr/share/man/man1/top.1.gz
(0.08718) | /mnt/linux_base/usr/share/man/man1/tmux.1.gz
(0.08665) | /mnt/linux_base/usr/share/man/man1/awk.1.gz
(0.06611) | /mnt/linux_base/usr/share/man/man1/screen.1.gz
(0.07075) | /mnt/linux_base/usr/share/man/man1/git-config.1.gz
(0.08041) | /mnt/linux_base/usr/share/man/man1/curl.1.gz
(0.07633) | /mnt/linux_base/usr/share/man/man1/busybox.1.gz
(0.07022) | /mnt/linux_base/usr/share/man/man3/pcrepattern.3.gz
(0.08504) | /mnt/linux_base/usr/share/man/es/man8/dnsMasq.8.gz
(0.09558) | /mnt/linux_base/usr/share/man/man7/systemd.directives.7.gz
(0.09402) | /mnt/linux_base/usr/share/man/man7/mdoc.samples.7.gz
```

Figure 6.49 High Entropy files in /mnt/linux_base (Additional mounted volume containing forensic evidence)

Clamscan is a malware scanner that comes loaded when we install SIFT Workstation on our instances, it is used to scan all data and infected files present on the system.

```
ubuntu@siftworkstation: ~
$ clamscan -i -r --log=/cases/clam.log /mnt/linux_mount/
/mnt/linux_mount/usr/bin/upx-ucl: Unix.Trojan.Mirai-9936831-0 FOUND

----- SCAN SUMMARY -----
Known viruses: 8612220
Engine version: 0.103.5
Scanned directories: 18949
Scanned files: 152078
Infected files: 1
Total errors: 124
Data scanned: 7669.29 MB
Data read: 7304.66 MB (ratio 1.05 :1)
Time: 2091.245 sec (34 m 51 s)
Start Date: 2022:04:20 12:29:25
End Date: 2022:04:20 13:04:16
ubuntu@siftworkstation: ~
$ |
```

Figure 6.50 Clamscan Results

At this stage, the forensic investigator must have realized that there must be multiple files which are infected, in order to check which files are infected third party hash hook-ups from online websites like virustotal.com can provide more clearer results to the investigator. After executing the below mentioned commands on SIFT workstation multiple csv files will be generated containing links of virustotal which can be pasted on the website to get better view of infected files present.

- virustotal-search.py investigate_files.md5 > virustotal-results.txt
- virustotal-submit.py virustotal-search.pkl

```
ubuntu@siftworkstation: ~
$ ls
Desktop           investigate_files.md5-md5.idx  known_files.md5-md5.idx2          virustotal-search-20220420-165827.csv
changed.md5       investigate_files.md5-md5.idx2  signature-base                  virustotal-search.pkl
changed_files.txt known_files.md5                virustotal-results.txt            virustotal-submit-20220420-122040.csv
investigate_files.md5 known_files.md5-md5.idx      virustotal-search-20220420-121606.csv  virustotal-submit-20220420-133819.csv
ubuntu@siftworkstation: ~
$ cat virustotal-submit-20220420-133819.csv
Filename;Response;Message;md5;sha256;Scan ID;Permalink
virustotal-search.pkl;1;Scan request successfully queued, come back later for the report;1efff85cccd2bb7758438e2477ff7c6a67;05cdbd7c9bad8c73a64f273ea241da0beaf380dac9ad8beb1a6b1226bb6d12d;05cd9d7c9bad8c73a64f273ea241daf0beaf380dac9ad8beb1a6b1226bb6d12d-1650461901;https://www.virustotal.com/gui/file/05cd9d7c9bad8c73a64f273ea241daf0beaf380dac9ad8beb1a6b1226bb6d12d/detection/f-05cd9d7c9bad8c73a64f273ea241daf0beaf380dac9ad8beb1a6b1226bb6d12d-1650461901
1
ubuntu@siftworkstation: ~
$ |
```

Figure 6.51 Linking Virustotal.com

The link generated from multiple .csv files as shown above will give details about the infected files present on the EC2 instance.

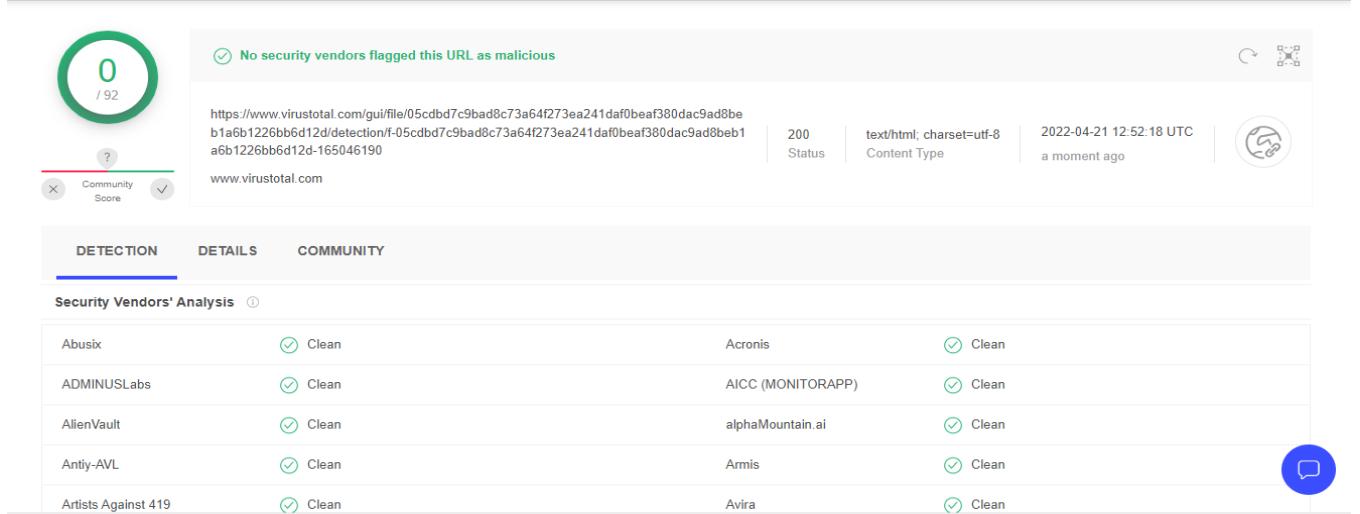


Figure 6.52 Third Party Hash Hook-ups 1

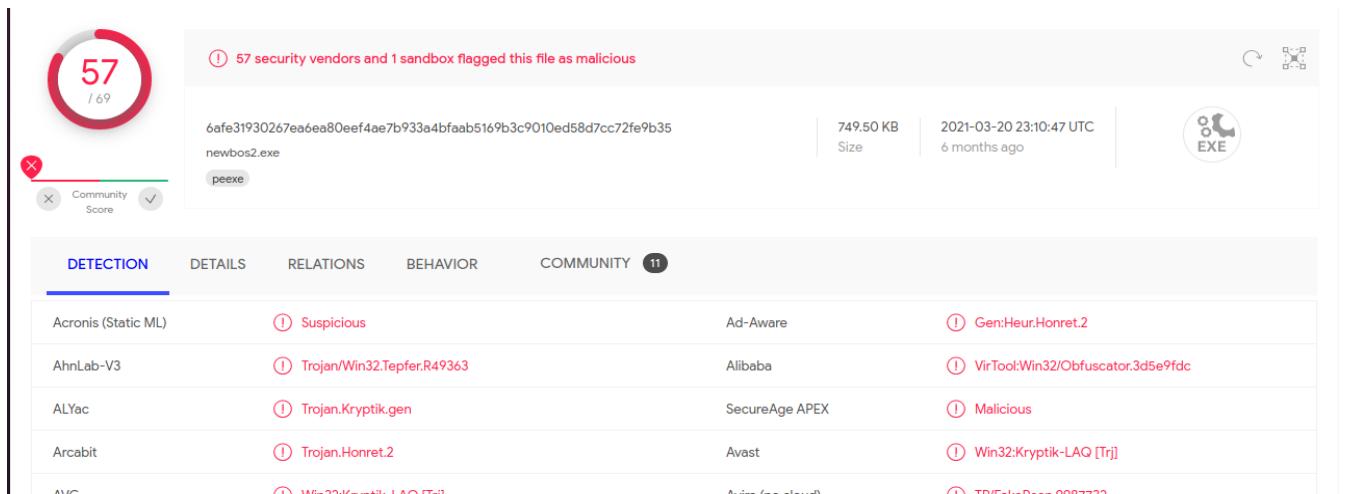


Figure 6.53 Third Party Hash Hook-ups 2

CHAPTER: 7 CONCLUSION AND FUTURE WORK

CHAPTER 7 CONCLUSION AND FUTURE WORK

Conclusion

In conclusion, the number of cases and the severity, refinement of malware attacks and expense of malware infect is increasing at an alarming rate. Malware should be detected as early as possible and mitigated. In this project, cybersecurity and security in-depth principles are applied to the cloud environment. These propositions indicate that defense controls of the cloud environment will not succeed at numerous cases and an attack might prevail so the companies must have response mechanisms to put off these attacks. Log monitoring and digital forensics gathering are the main trait for investigators for tracking and detecting active malware attacks. In this project 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. After that, we validated the applicability and limitation of deploying this baseline by doing a malware attack Any type of malicious activity which might takes place on the cloud account can be mitigated if the data is monitored properly. A baseline is built on AWS using a service called AWS CloudTrail which generated logs of activities taking place within S3. and then they were integrated with Splunk which is a SIEM Tool to perform investigation and analysis and take some steps regarding attack decision. Splunk provided data correlation, enrichment, integration with other security events, and long-term storage. Lastly in order to investigate the vulnerability of VMs Investigations were performed on the compromised IaaS VMs which displayed how a user should be careful and alert of the vulnerability of the system and take necessary steps to prevent it in future.

Future Work

As there are ample number of malware attacks happening day by day which are very difficult to track whether it is on-premises or on cloud environment, security management and investigation techniques should be given more value as the data uploaded on these environments is very important leading to changes to world economy at some stages. For future work, we believe that cloud services sources should provide the necessary tools for executing volatile memory analysis for their VMs. Also, develop a new automated tool for incident response and forensics investigation on the IaaS.

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_Tool_S
- [7] <https://docs.splunk.com/Documentation>
- [8] J. Dykstra , "Digital forensics for infrastructure-as-a-service cloud computing," Ph.D dissertation, Faculty of the Graduate School of the University of Maryland, Baltimore County, 2013.
- [9] A. Pichan, M. Lazarescu and S. T. Soh, "Cloud forensics: Technical challenges, solutions and comparative analysis," Digital Investigation ,Elsevier, Vol.13, pp. 38-57, 23 March 2015.

G20 IBM PROJECT REPORT

by Jainam Shah

Submission date: 26-Apr-2022 05:58PM (UTC+0530)

Submission ID: 1815266059

File name: IBM.pdf (2.81M)

Word count: 2542

Character count: 13147

CHAPTER 1 INTRODUCTION

The availability to store data on cloud is a modern technology. In recent years, utilization of services on cloud platform is skyrocketing at an alarming pace; now it has turned more famous after the emergence of the 4th Economic Industrial Revolution. In the year of 2020, nearly 83 percentage of workloads in business are functional on the cloud platform, and around 94 percentage of companies in today's market utilize a cloud service in one of their services. There are approximately 3 most used cloud services including Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Monitoring the current market, (IaaS) is the most rapidly growing service in Cloud Computing.

However due to ample number of exquisite features being available on the IaaS cloud, it is turning into a playground to many attacks of malware for the following reasons:

- 1) Companies which are providing cloud services unremittingly offer off the chart's performance with more computation power for their clients. These virtual machines on cloud are primary targets for cryptocurrency mining malware.
- 2) The rise of work from home era and internationally separated manpower and resources accessibility after the Sars-Cov2 (corona virus) has provided the attackers more ways to conceal their detrimental traffic to take over the cloud-hosted virtual machines, and utilize them for their malicious activities.
- 3) The definite rise in IoT applications that use cloud-hosted data and services to analyze the gigantic quantity of data created by these applications to construct 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 directs its goal to utilize the best ways to attain non-stop monitoring of malware attacks on the cloud.
- The techniques of logging data and performing forensics have always been the foundations of accomplishing non-stop monitoring and detection of malware attacks.
- To arrogate the perfect methods to bring the concept of forensics and logging to the cloud and integrate them with on-premises visibility.
- Attaining the proper monitoring considering the whole security standpoint of the organization assets whether they are stored on an on-premises system or on the cloud platform.

2

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.

2 **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

Minimum Hardware Requirements

| | |
|------------------|--------------|
| Processor | 2.0 GHz |
| RAM | 8GB |
| HDD | Minimum 30GB |

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

| | |
|-------------------------------|---|
| OS | Any operating system which can support an internet browser. |
| Programming languages | - |
| Tools and Technologies | AWS, Splunk, kali Linux |

Table 3.2 Minimum Software Requirements

CHAPTER 4 PROCESS MODEL

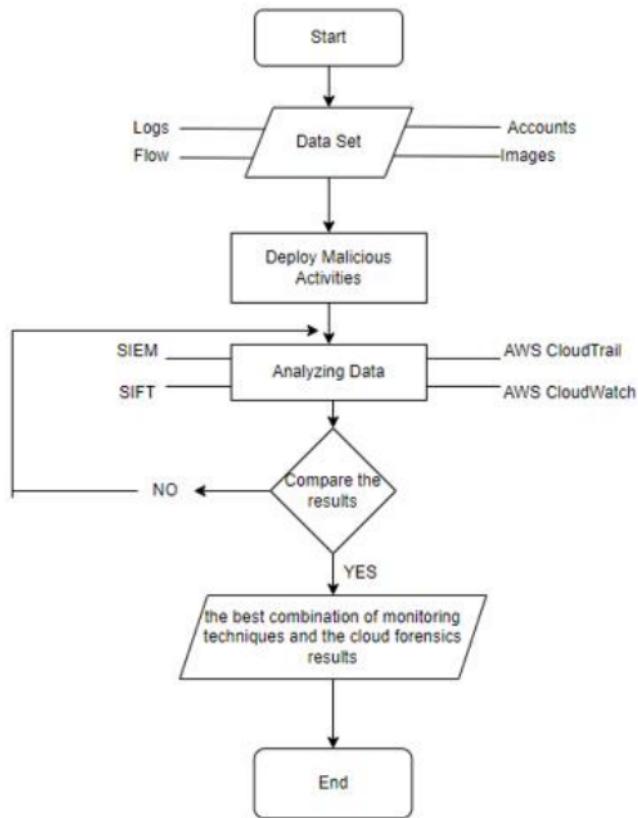


Figure 4.1 Process Model of Project

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

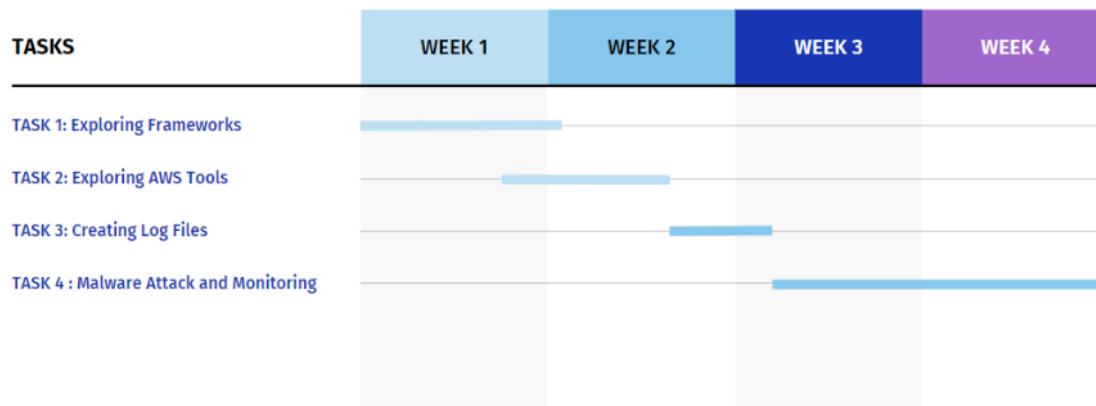


Figure 5.1 Task Completion Time Duration in First Phase
Missing "", ETS

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

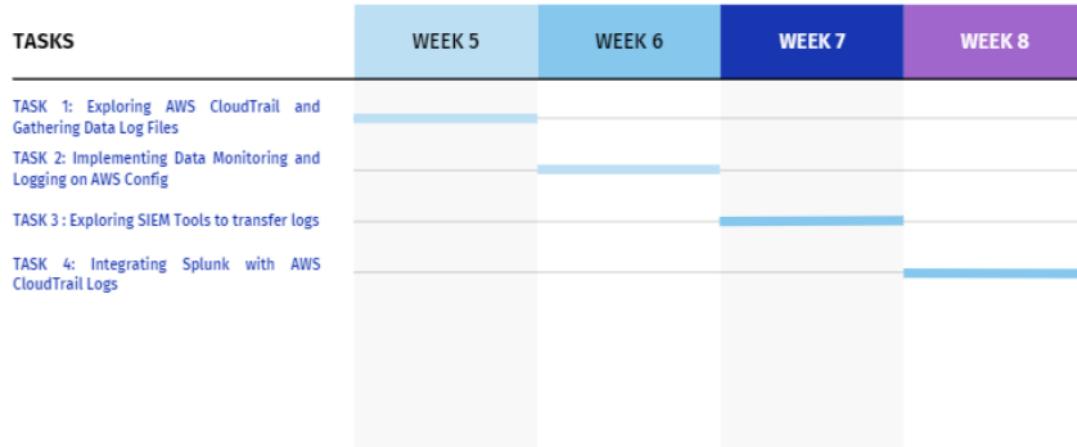


Figure 5.2 Task Completion Time Duration in Second Phase

5.1.5 Tasks for Evidence Capturing and Forensic Analysis in Third Phase

- Task: - 1 SIFT Exploration
- Task: - 2 AWS EC2 Exploration and setting up investigation tools on EC2
- Task: - 3 Cloud Forensic Analysis and Evidence Capturing
- Task: - 4 Additional Cloud Forensics

5.1.6 Time Duration to Complete Third Phase



2

Figure 5.3 Task Completion Time Duration in Third Phase

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 can be considered as the most important service, which provides basic computational services such as servers, networking, and storage. This service supplements the availability of the system on lower costs and providing a more pliable system.

2. Malware Attacks - In layman terms the term malware means dangerous and harmful, it has similar effect on software, networks, OS, or other components. The most major challenge while using IaaS cloud world is its vulnerability and the possibility of malware attacks; it is a vital concern to devices present at home as well as the devices used for business in a corporate and also on cloud VMs.

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.: - Techniques based on Signature/Behavior for malware detection, Machine Learning Based malware detection methods etc.

Missing "," 

4. Cloud Forensics – It can be defined as techniques that are utilized in order to perform collecting and storing incidents happening around and their visibility, remodeling events, recognizing when an incident is happening, how an incident is happening, and where an incident is happening, and implementing information/data regarding that.

Run-on 

6.2 Methodology

The methodology of the mentioned project has been segregated into two parts:

The First: Whenever or wherever a malware attack takes place, the investigator should make cloud analysis for that particular malware detection.

The Second: is to perform forensics analysis in the IaaS Cloud after the attack happens.

6.2.1 Gathering Data

In today's, there are two initiatives that define and segregate in an orderly manner how each cloud attack technique is witnessed; they are 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

5 1
6.3.1. Test Environment - The tests for this mentioned project were performed on Amazon Web services (AWS), the main reason for choosing AWS for this project was because it is currently the market leader and provides so many public cloud services and possesses a varied range of service catalog and thus in turn making it a more suitable choice for the mentioned problem.

Missing "", (ETS)

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.

1 Non-Stop monitoring on IaaS can be attained by collecting and processing the following

- Monitoring API Calls (CloudTrail's logs in AWS).
- Hosting logs and logs of deployed HIDs.
- VPC flows.
- Logs of numerous cloud resources (CloudWatch Logs in AWS)
- Validation and Integrity of images and instances.

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

AWS CloudWatch – to gather and monitor metrics, gather and monitor log files, setting 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

As per the MITRE ATT&CK framework, most majorly used attack vectors for attacks on cloud and malware attacks aiming cloud-hosted environments is the cloud account takeover. There are multiple ways to detect that, the most promising way is detecting changes in the billings on AWS. Most public cloud providers provide features to allow their customers to make billing tags and then send them emails whenever alarms are triggered.

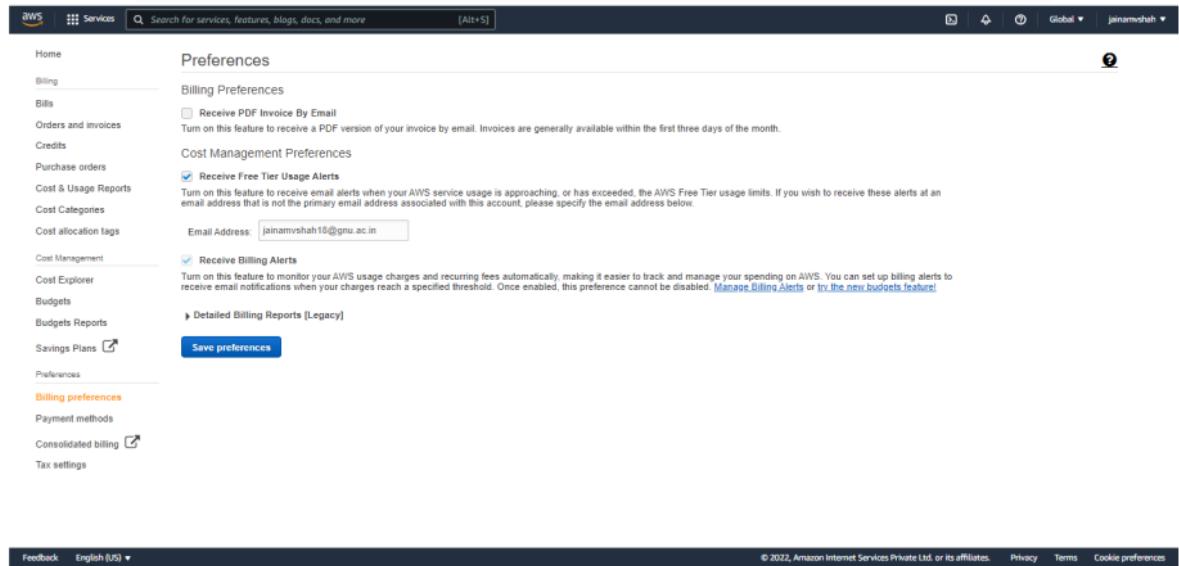


Figure 6.1 AWS Billing Preferences

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 Cloud platform provides a service called AWS Config, this service enables monitoring AWS resource configurations and keep tracks on resource inventory and varied changes that might take place within it, which can be utilized to detect any malicious configuration changes that an attacker might try to make in order to gain control over the compromised account's resources. The monitoring information gathered from this then can be absorbed using AWS CloudWatch and SNS Notifications can be created based on them.

Malware attacks earmark and make changes to the data stored and any misconfigured cloud storage leading to data leak. By making use of AWS Config, the rules like proper storage versioning is kept on for AWS storage (S3). By enabling the s3-bucketversioning- enabled rule, another activity audited by attackers is to hide their vindictive API calls by disabling API calls monitoring, another rule that can be configured is to check if CloudTrail service is enabled or not and yet another rule to detect whether the volumes which are utilized are having encryption or not.

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

The screenshot shows the AWS S3 Buckets page. On the left, there's a sidebar with options like Buckets, Storage Lens, and Feature spotlight. The main area has a heading 'Account snapshot' with a link to 'View Storage Lens dashboard'. Below it, a table lists three buckets:

| Name | AWS Region | Access | Creation date |
|---|----------------------------------|-----------------------|--|
| aws-cloudtrail-logs-601422970468-722e7fc6 | Asia Pacific (Mumbai) ap-south-1 | Public | February 1, 2022, 17:02:31 (UTC+05:30) |
| jainamvshah18 | Asia Pacific (Mumbai) ap-south-1 | Objects can be public | February 1, 2022, 09:18:07 (UTC+05:30) |
| malwaredemo | Asia Pacific (Mumbai) ap-south-1 | Public | February 2, 2022, 19:11:52 (UTC+05:30) |

Figure 6.2 S3 Buckets

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

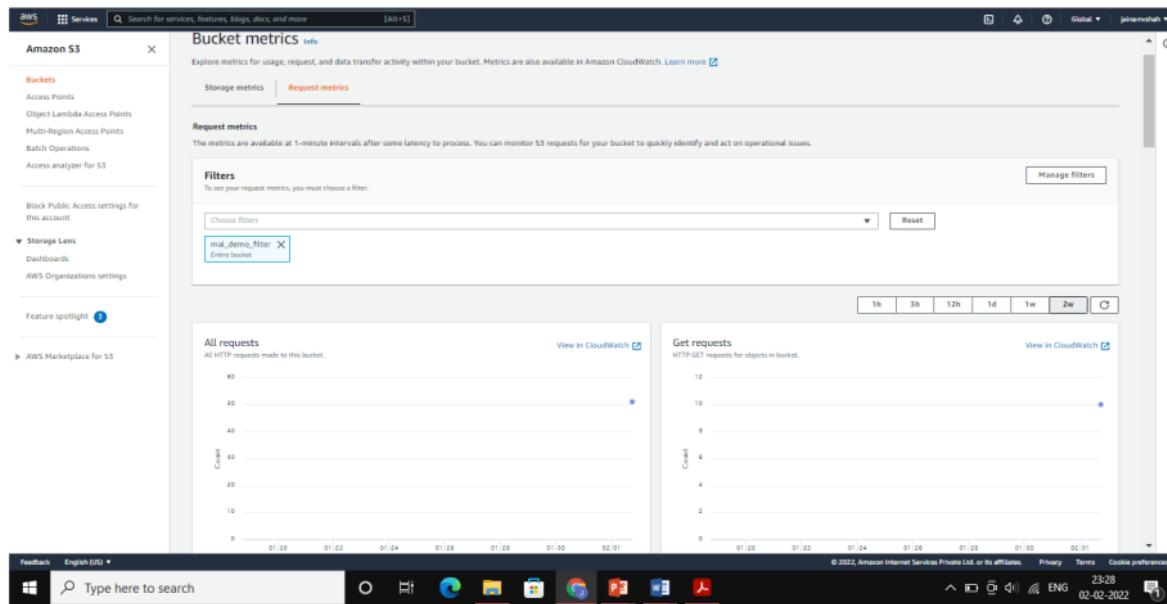


Figure 6.3 Creating Metrics for bucket

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.

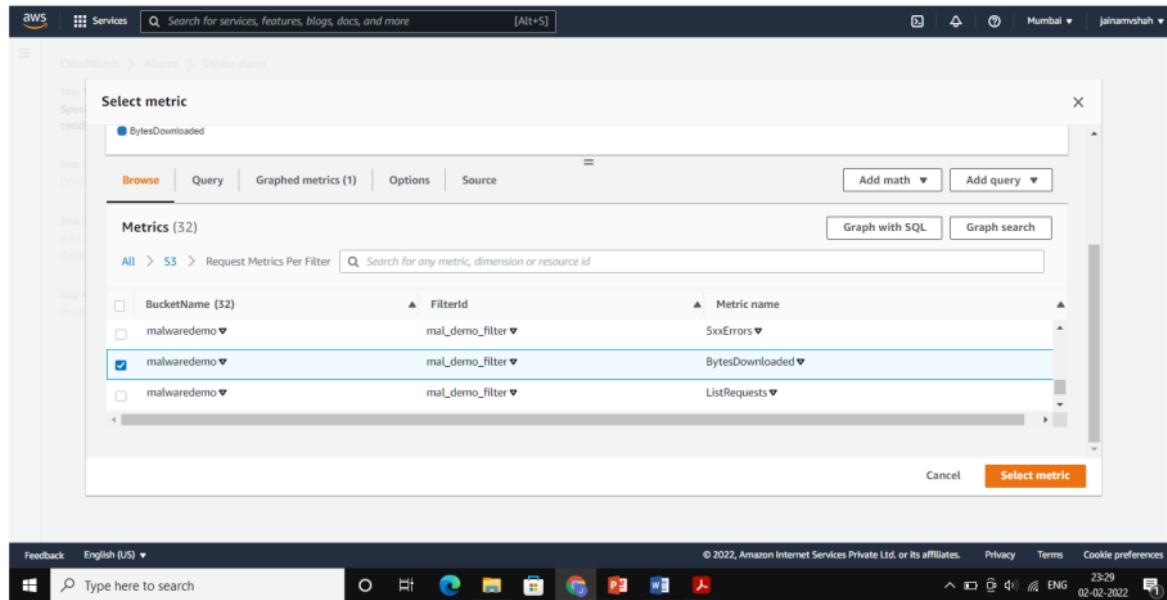


Figure 6.4 Setting up Alarm

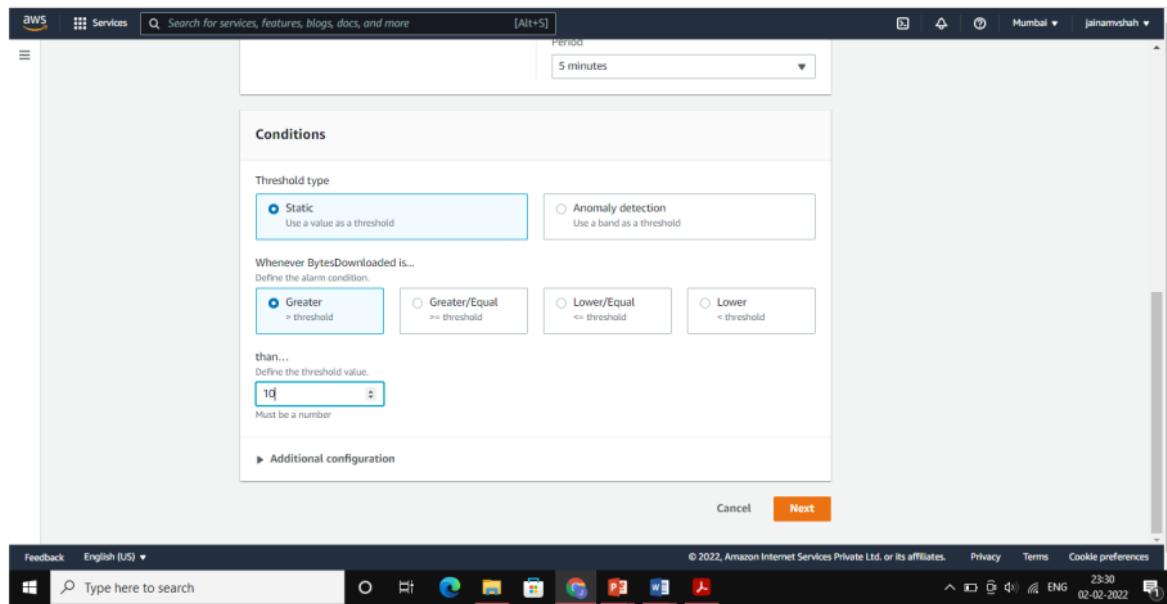


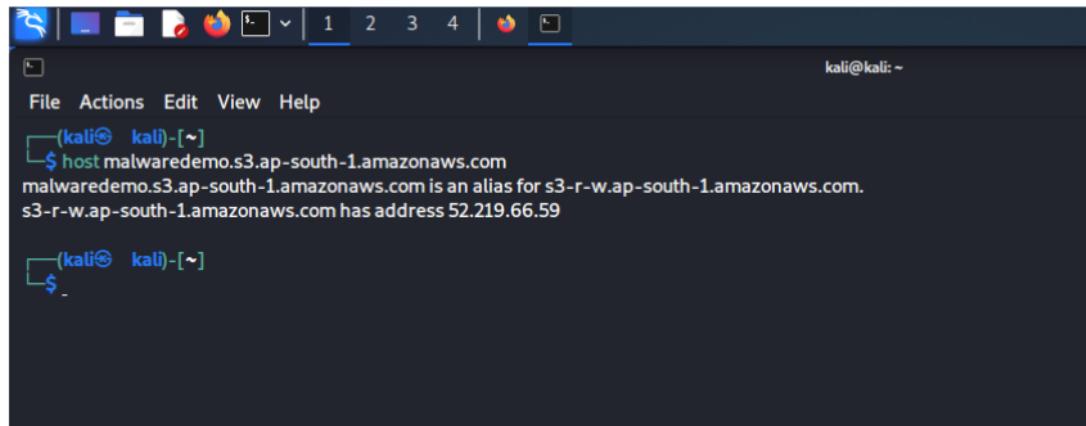
Figure 6.5 Defining threshold value for a definite amount of size

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

Article Error (ETS)

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



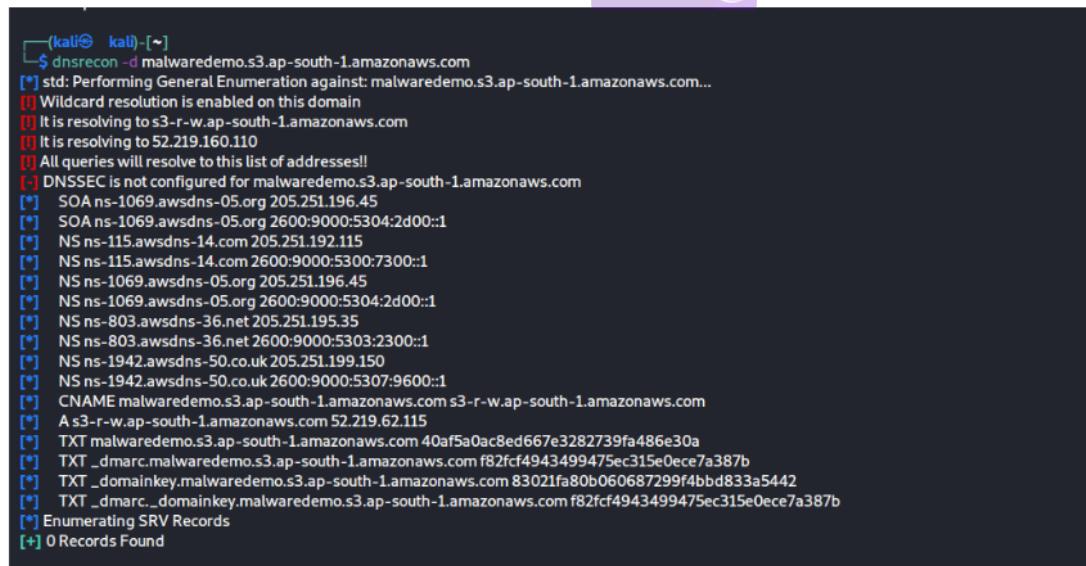
```
(kali㉿ kali) [~]
$ host malwaredemo.s3.ap-south-1.amazonaws.com
malwaredemo.s3.ap-south-1.amazonaws.com is an alias for s3-r-w.ap-south-1.amazonaws.com.
s3-r-w.ap-south-1.amazonaws.com has address 52.219.66.59

(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

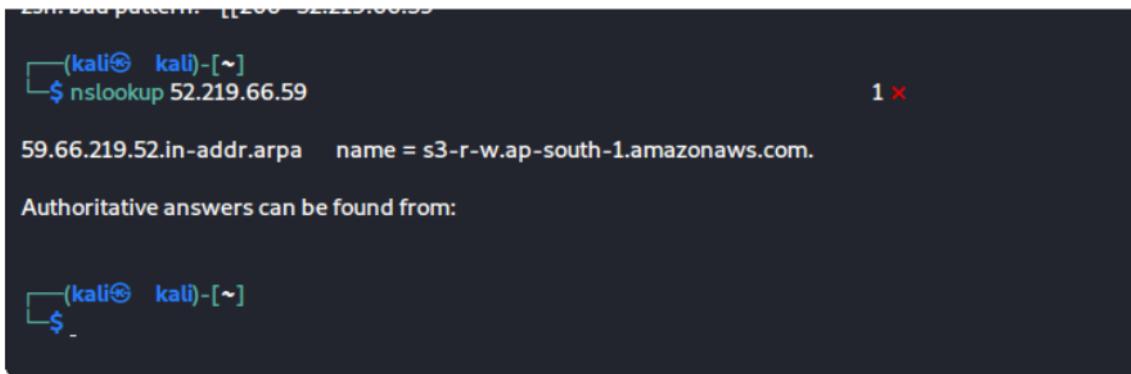
Article Error (ETS)



```
(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㉿ kali] -[~]
$ nslookup 52.219.66.59
      1 ×

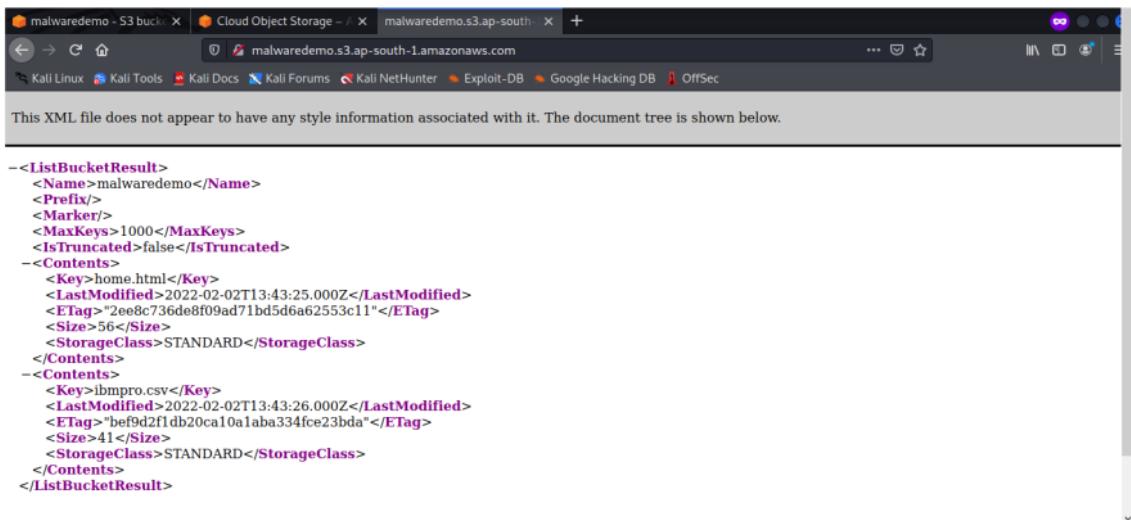
59.66.219.52.in-addr.arpa  name = s3-r-w.ap-south-1.amazonaws.com.

Authoritative answers can be found from:

└─[kali㉿ 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.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

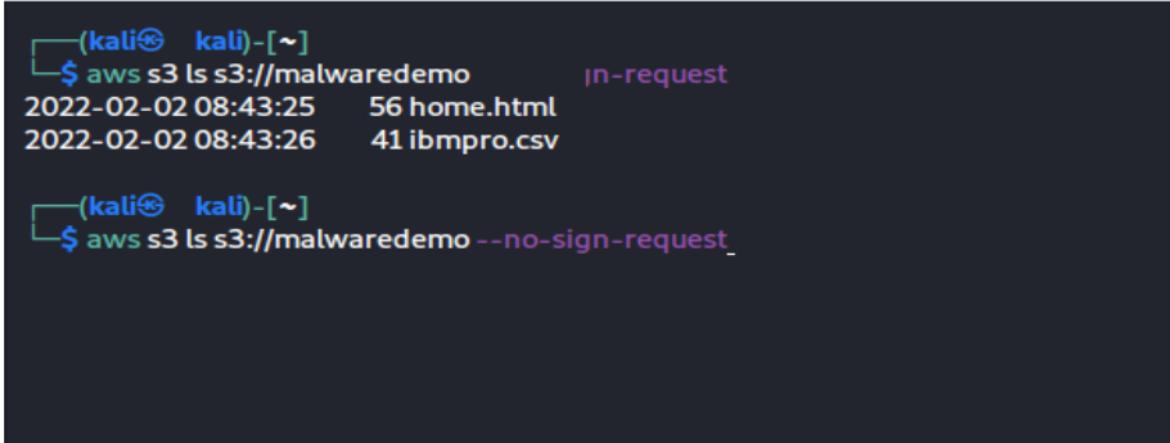
```
<ListBucketResult>
<Name>malwaredemo</Name>
<Prefix/>
<Marker/>
<MaxKeys>1000</MaxKeys>
<IsTruncated>false</IsTruncated>
<Contents>
<Key>home.html</Key>
<LastModified>2022-02-02T13:43:25.000Z</LastModified>
<ETag>"2ee8c736de8f09ad71bd5d6a62553c11"</ETag>
<Size>56</Size>
<StorageClass>STANDARD</StorageClass>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

Article Error 

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

S/V (ETS)



```
(kali㉿ kali)-[~]
└─$ aws s3 ls s3://malwaredemo      in-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

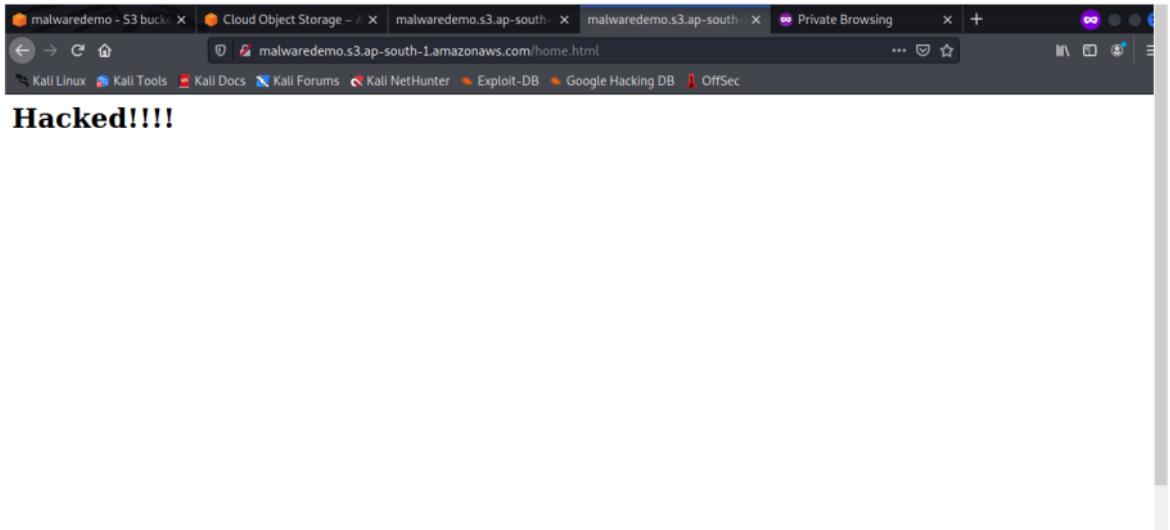


Figure 6.11 Home.html file

Article Error (ETS)

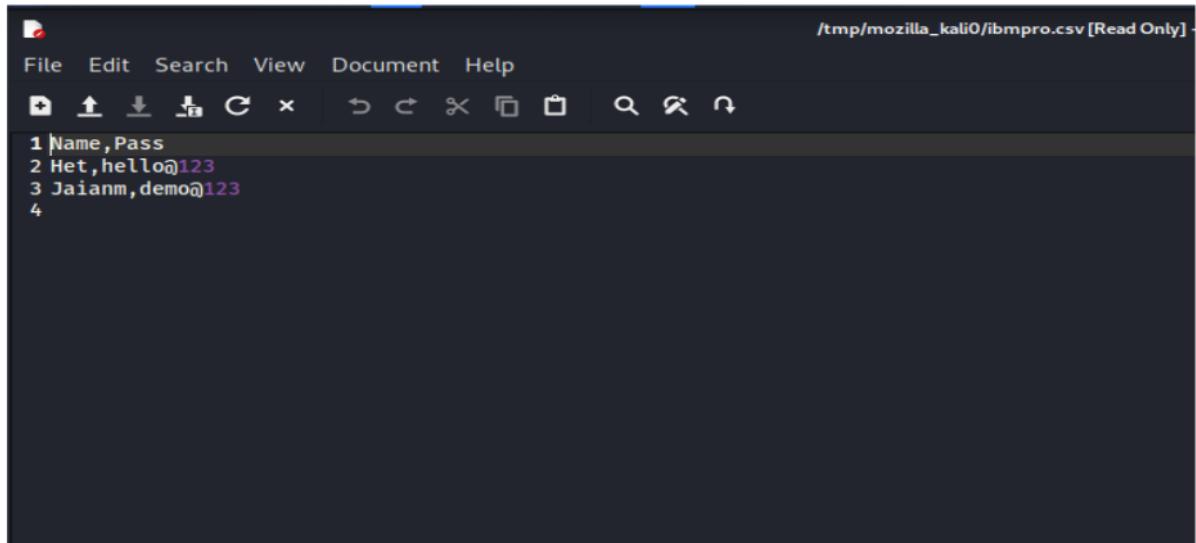


Figure 6.12 ibmpro.csv

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

```
(kali㉿ kali)~]$ python3 -m S3Scanner scan --bucket malwaredemo.s3.ap-south-1.amazonaws.com
Warning: AWS credentials not configured - functionality will be limited. Run: `aws configure` to fix this.

malwaredemo | bucket_exists | AuthUsers: [], AllUsers: [Read, ReadACP]

(kali㉿ kali)~]$
```

Figure 6.13 Listing Users

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

```
(kali㉿ kali)~]$ python3 -m S3Scanner dump --bucket malwaredemo.s3.ap-south-1.amazonaws.com --dump-dir ~/Desktop/S3Dump/
Warning: AWS credentials not configured - functionality will be limited. Run: `aws configure` to fix this.

malwaredemo | Enumerating bucket objects...
malwaredemo | Total Objects: 2, Total Size: 97.0B
malwaredemo | Dumping contents using 4 threads...
malwaredemo | Dumping completed

(kali㉿ kali)~]$
```

Figure 6.14 Dump status

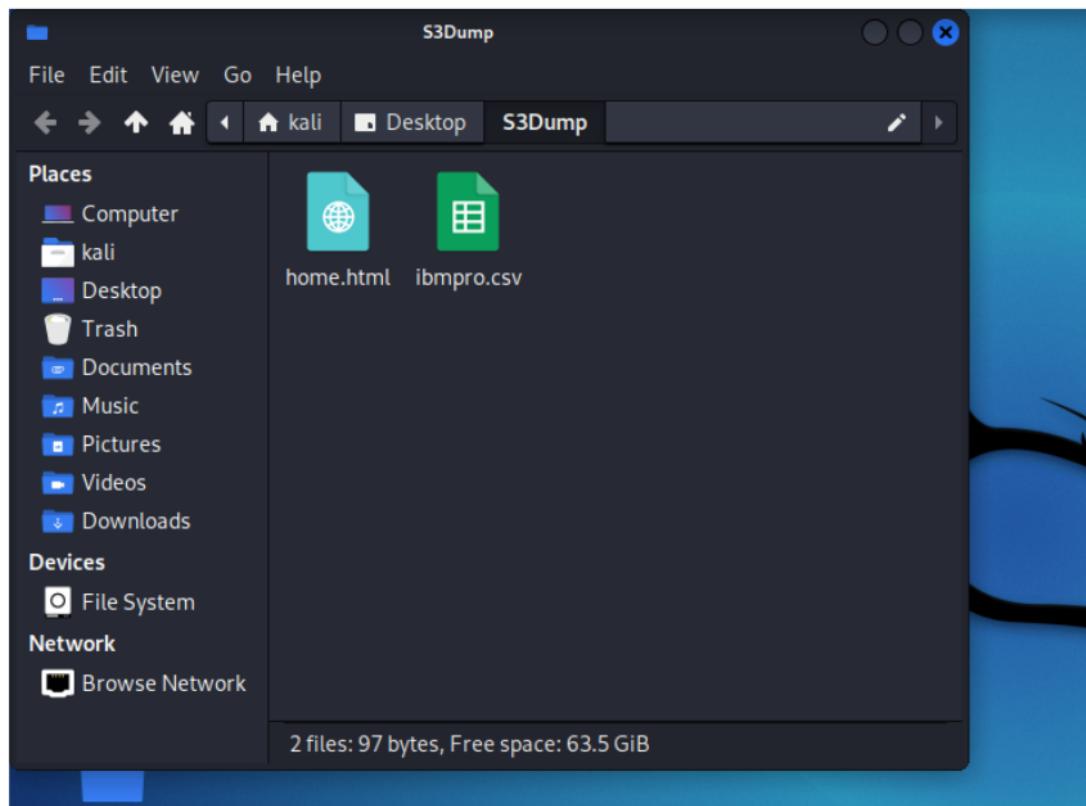


Figure 6.15 Files downloaded on local machine

Article Error 

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.

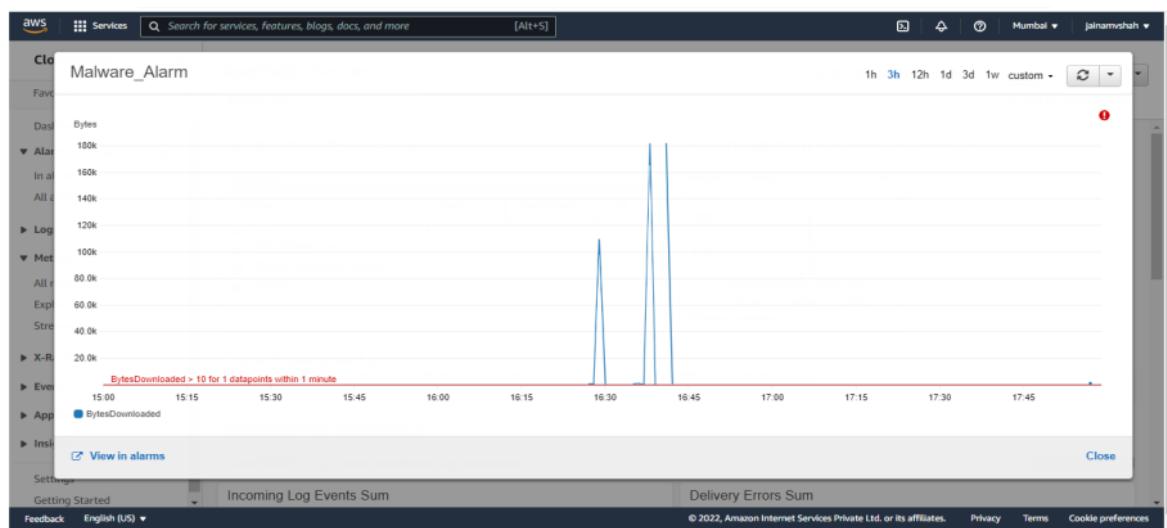


Figure 6.16 Malware Alarm

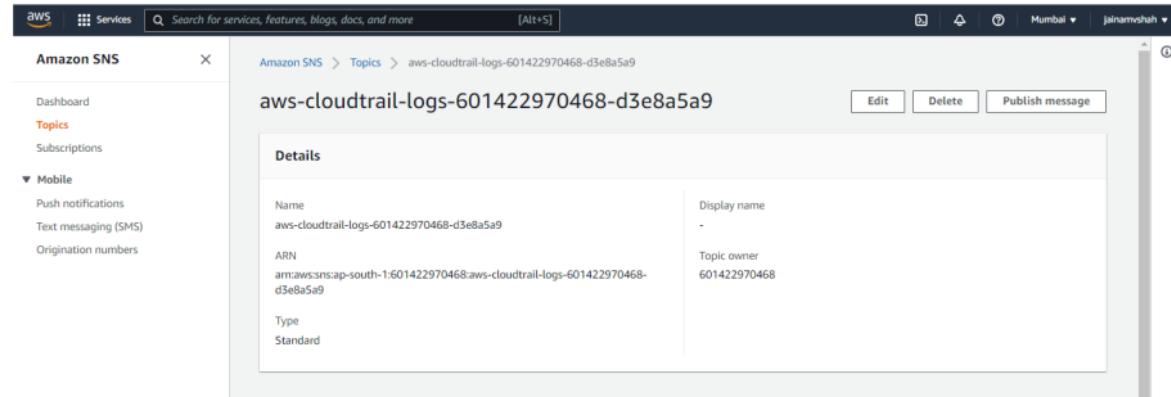
4. Implementing Data Monitoring using SNS

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 AWS account.

Article Error 

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 

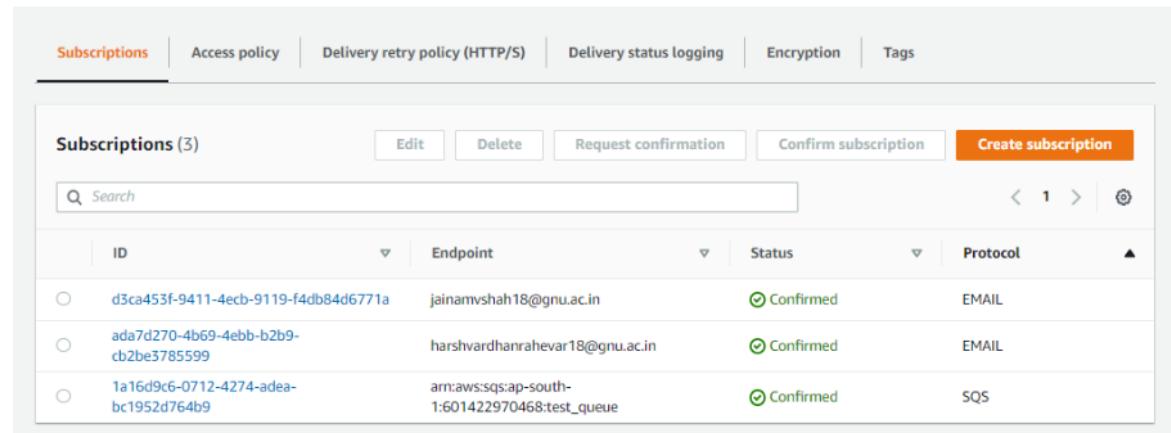


The screenshot shows the AWS SNS Topics page. On the left, a sidebar lists 'Dashboard', 'Topics' (which is selected and highlighted in orange), and 'Subscriptions'. Below that is a 'Mobile' section with 'Push notifications', 'Text messaging (SMS)', and 'Origination numbers'. The main content area shows a single topic named 'aws-cloudtrail-logs-601422970468-d3e8a5a9'. The topic details are listed in a table:

| | |
|--------------|---|
| Name | aws-cloudtrail-logs-601422970468-d3e8a5a9 |
| ARN | arn:aws:sns:ap-south-1:601422970468:aws-cloudtrail-logs-601422970468-d3e8a5a9 |
| Type | Standard |
| Display name | - |
| Topic owner | 601422970468 |

Buttons for 'Edit', 'Delete', and 'Publish message' are located at the top right of the topic card. The URL in the address bar is 'Amazon SNS > Topics > aws-cloudtrail-logs-601422970468-d3e8a5a9'.

Figure 6.17 SNS Topic



The screenshot shows the AWS SNS Subscriptions page. At the top, there are tabs for 'Subscriptions' (which is selected and highlighted in orange), 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. Below the tabs, there is a search bar and a table titled 'Subscriptions (3)'. The table has columns: ID, Endpoint, Status, and Protocol. The data is as follows:

| ID | Endpoint | Status | Protocol |
|--------------------------------------|--|-----------|----------|
| d3ca453f-9411-4ecb-9119-f4db84d6771a | jainamvshah18@gnu.ac.in | Confirmed | EMAIL |
| ada7d270-4b69-4ebb-b2b9-cb2be3785599 | harshvardhanrahevar18@gnu.ac.in | Confirmed | EMAIL |
| 1a16d9c6-0712-4274-adea-bc1952d764b9 | arn:aws:sqs:ap-south-1:601422970468:test_queue | Confirmed | SQS |

Figure 6.18 SNS Notification Subscriptions

| Subscription: ada7d270-4b69-4ebb-b2b9-cb2be3785599 | |
|---|--|
| Edit Delete | |
| 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 | |
|---|--|
| Edit Delete | |
| 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.

Missing "", ETS

```
{
  "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 utilized to track account activity and API calls, as most cloud providers offer their services via APIs this is a very important service. These feeds from CloudTrail can also be integrated with AWS CloudWatch in order to create metrics for employing alarms for any suspicious account's behavior or any miscellaneous 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.

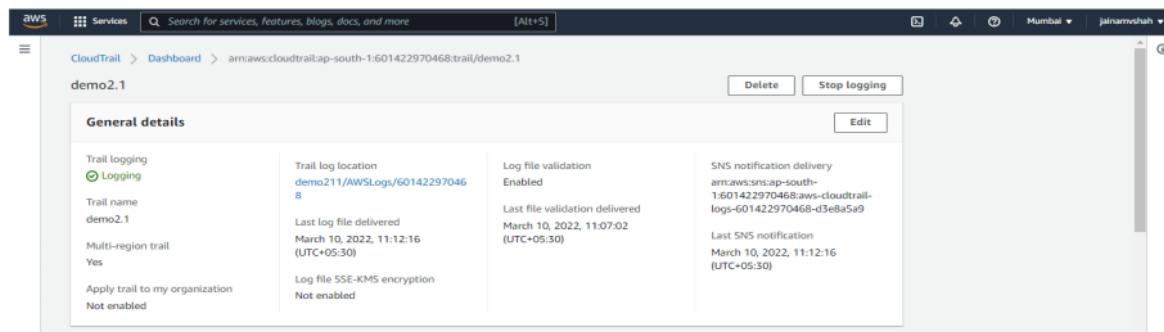


Figure 6.20 CloudTrail Details

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.

Sentence Cap. (ETS)

Amazon S3 > demo211

demo211 info

Publicly accessible

Objects Properties Permissions Metrics Management Access Points

Objects (4)

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

| <input type="checkbox"/> | Name | Type | Last modified | Size | Storage class |
|--------------------------|---------------|--------|-------------------------------------|----------|---------------|
| <input type="checkbox"/> | AWSLogs/ | Folder | - | - | - |
| <input type="checkbox"/> | IBM_Demo.docx | docx | March 9, 2022, 21:39:35 (UTC+05:30) | 689.1 KB | Standard |
| <input type="checkbox"/> | ibmpro.csv | csv | March 9, 2022, 21:39:36 (UTC+05:30) | 41.0 B | Standard |
| <input type="checkbox"/> | notes.txt | txt | March 9, 2022, 21:39:37 (UTC+05:30) | 163.0 B | Standard |

Figure 6.21 S3 Bucket with CloudTrail Logs

Amazon S3 > demo211 > AWSLogs/ > 601422970468/ > CloudTrail/ > us-west-2/ > 2022/ > 03/ > 09/

09/ Copy S3 URI

Objects Properties

Objects (13)

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

| <input type="checkbox"/> | Name | Type | Last modified | Size | Storage class |
|--------------------------|---|------|--------------------------------------|---------|---------------|
| <input type="checkbox"/> | 601422970468_CloudTrail_us-west-2_20220309T1950Z_0rP6edVHmlANBkxv.json.gz | gz | March 10, 2022, 01:28:20 (UTC+05:30) | 902.0 B | Standard |
| <input type="checkbox"/> | 601422970468_CloudTrail_us-west-2_20220309T1950Z_IWcOQ1tcAk/iee6Z.json.gz | gz | March 10, 2022, 01:20:40 (UTC+05:30) | 903.0 B | Standard |
| <input type="checkbox"/> | 601422970468_CloudTrail_us-west-2_20220309T1945Z_KVU3M711ZpYpbkp.json.gz | gz | March 10, 2022, 01:13:59 (UTC+05:30) | 1.0 KB | Standard |
| <input type="checkbox"/> | 601422970468_CloudTrail_us-west-2_20220309T1930Z_q3Hw64lwQ2HaDrX.json.gz | gz | March 10, 2022, 00:57:49 (UTC+05:30) | 1.0 KB | Standard |
| <input type="checkbox"/> | 601422970468_CloudTrail_us-west- | | March 10, 2022, 01:04:19 | | |

Figure 6.22 Data Log Files

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

- Detect any third-party AWS console logins from unknown places or countries.
- In most cases, companies/organizations do transfer logs to their own data center for long term storage, it's crucial to generate the logs in a text-like format such as JSON, for better understanding of complex data, from a test, notice AWS uses this concept in their generated logs and flows.
- Anchorage the storage API i.e. s3 API to import cloud trails to a search and arranging platform or security management systems like (SIEM solution) for creating more secured and robust use cases monitoring.

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 escorting cloud logs into a single point where they can be combined with on premises security events, thus enabling the investigator to have a single view of screen from which he/she can monitor the whole security strata.

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.

| Name | Key ID | IAM Role | Region Category | Inputs | Actions |
|-------|----------------------|-------------------------|-----------------|--------|-----------------------|
| user1 | AKIAYYB42HZSEAT5WT50 | Autodiscovered IAM Role | Global | 0 | Edit Clone Delete |

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:-

AWS Input Configuration

| | |
|-----------------------|--------------------------|
| Name | Test_input |
| AWS Account | user1 |
| AWS Region | Asia Pacific (Mumbai) |
| Use Private Endpoints | <input type="checkbox"/> |
| SNS Queue | test_queue |
| Source Type | aws.cloudtrail |
| Index | main |

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.



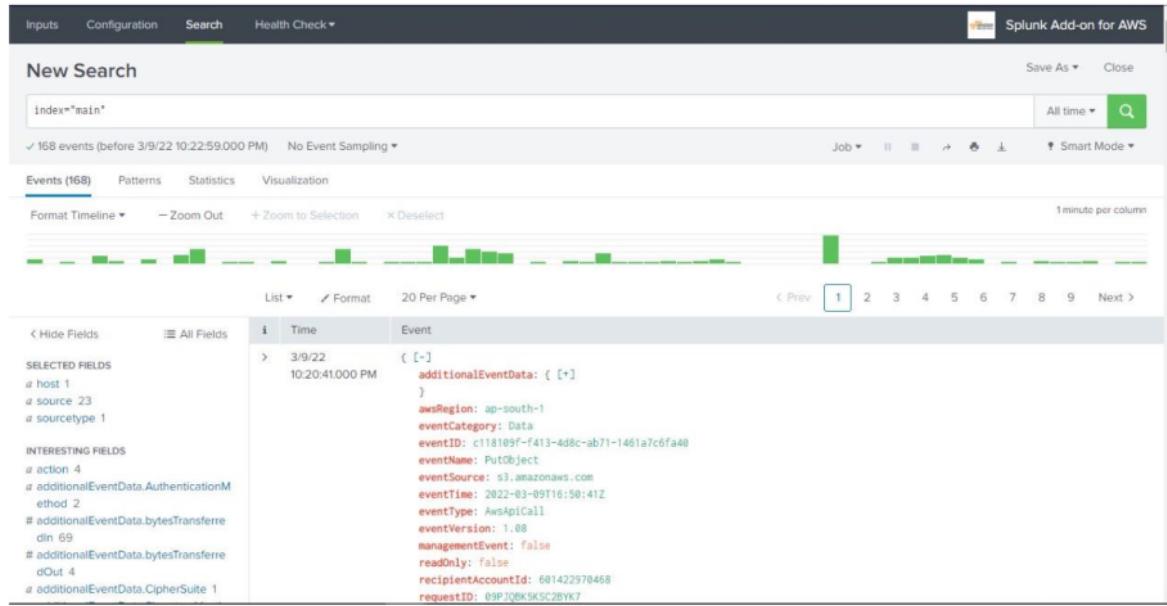


Figure 6.25 CloudTrail Logs Analysis

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, a glimpse of the created JSON Script is as follows: -

Article Error

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "config:GetComplianceSummaryByConfigRule",
        "sns:DeleteMessage",
        "iam:GetAccountPasswordPolicy",
        "s3:GetLifecycleConfiguration",
        "s3:GetBucketTagging",
        "ec2:DescribeInstances",
        "ec2:DescribeAddresses",
        "s3:GetBucketLogging",
        "s3:PutBucketOwnershipControls",
        "ec2:DescribeRegions",
        "sns:ReceiveMessage",
        "s3:GetAccelerateConfiguration",
        "ec2:DescribeSnapshots",
        "elasticloadbalancing:DescribeLoadBalancers",
        "lambda:InvokeFunction"
      ],
      "Resource": "*"
    }
  ]
}
```

Figure 6.26 JSON Script

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.

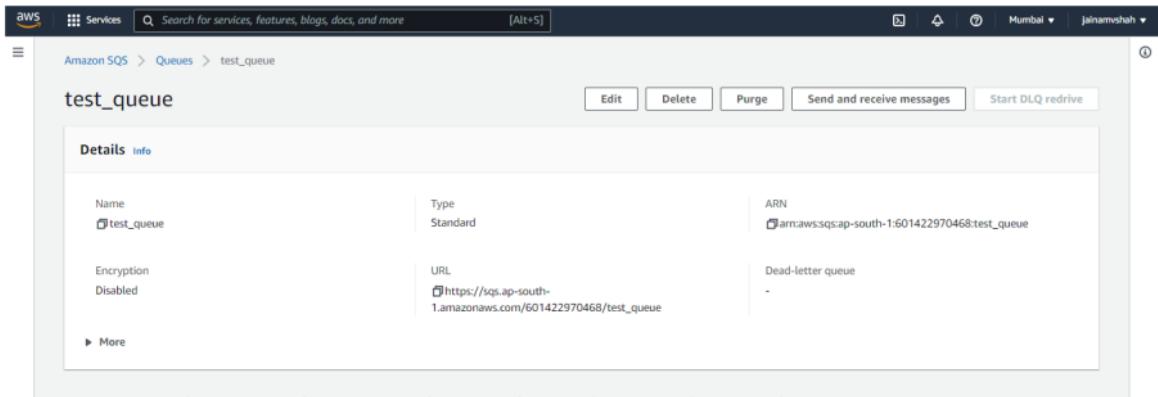


Figure 6.27 SQS Service

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.

A screenshot of an Excel spreadsheet titled 'demo - Excel'. The data is organized into columns labeled A through U. The first few rows show the schema:

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |
|---|----------|---------------------|----------|-----------------|-------|------|---|--------------------|-----------|----------|---------|-----------|----|---|----|-------|---|---|---|---|
| 1 | raw | | | | | | | | | | | | | | | | | | | |
| 2 | "eventV4 | 2022-03-07-modified | AuthHead | EC2HE-RS.SSE.S3 | SigV4 | 2371 | 0 | 2hwSEAV-AwsApiCall | ap-south- | 6.01E+11 | storage | PutObject | 16 | 9 | 50 | march | | | | |

Subsequent rows contain detailed log entries for various AWS API calls, such as PutObject, GetObject, and DeleteObject, across different regions and services like S3 and CloudWatch Metrics.

Figure 6.28 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.

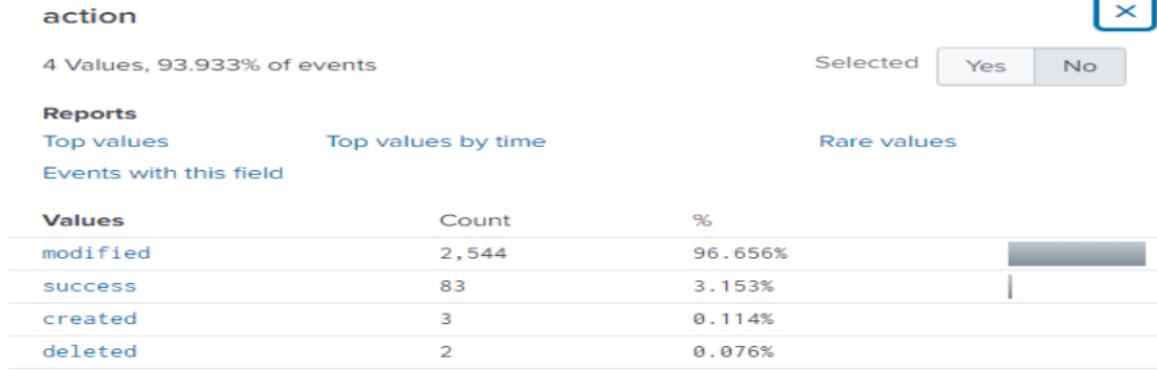


Figure 6.29 Action Field

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

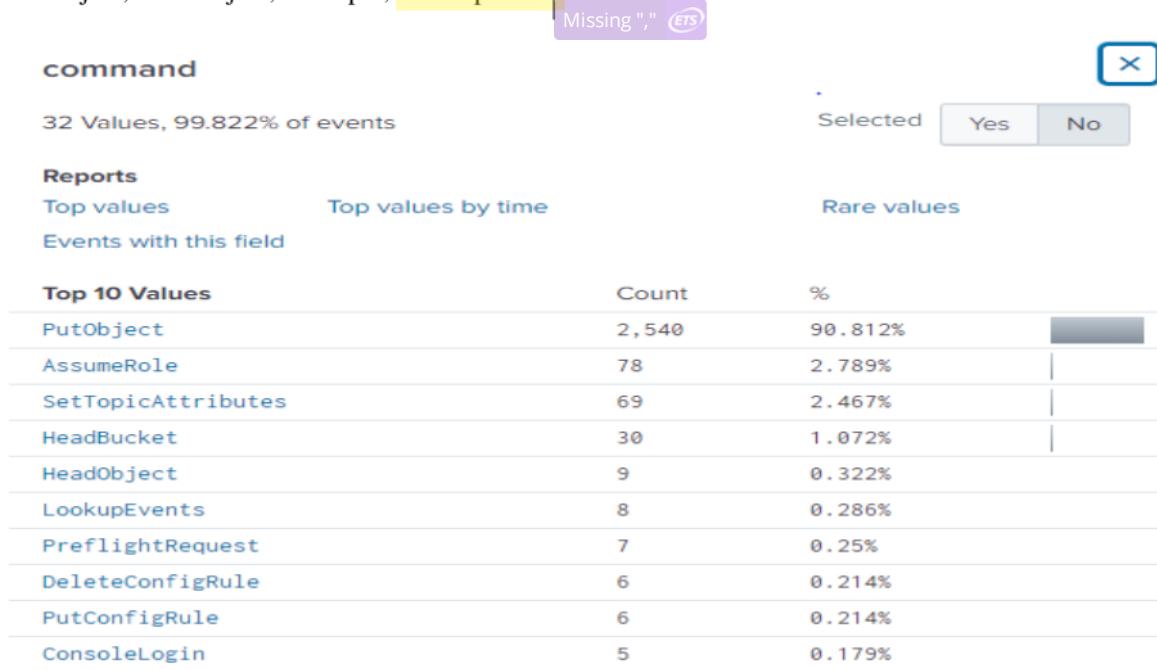


Figure 6.30 Command Field

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



Figure 6.31 ErrorCode Field

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



Figure 6.32 Host Field

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

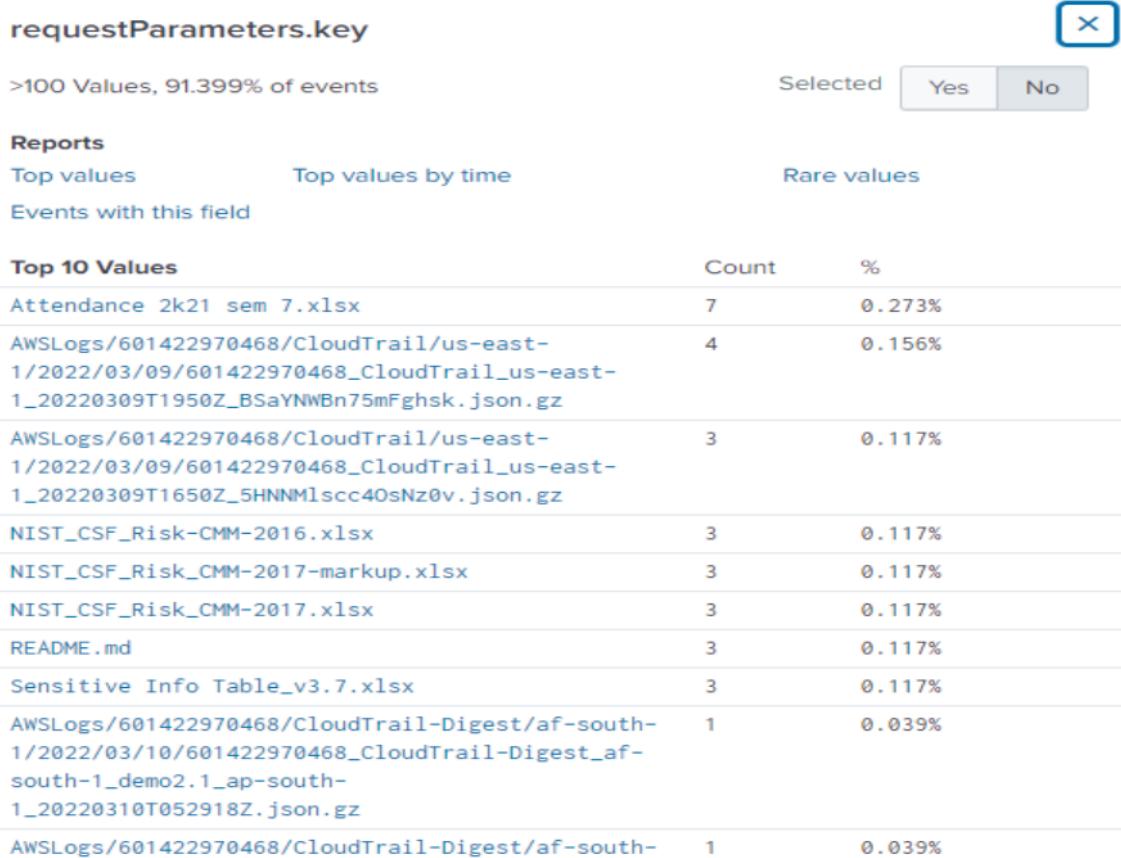


Figure 6.33 RequestParameterKey Field

6.4 Forensic Analysis in IaaS Cloud

Making use of cloud forensics to help organization in strengthening their incident response and threat detection capabilities, organizations must have proper forensics investigation tools to keep their cloud infrastructure secured in case of an attack, recognizing the signs of vulnerability as well as quickly locate an infection and its objectives before they have an impact on the organizations' important data.

If there is a case of a hacked virtual machine, most users terminate the virtual machine (VM), erasing all proof in the process. It would become very challenging to perform forensic analysis in such cases. Until now, there have been few tools and applications which monitor the system properly and gather data. When it comes to gathering and analyzing evidence, must look for the following:

- Network packet captures for forensics.
- Memory usage for a particular instance.
- Events and data logs

In order to provision a machine for forensic analysis, installing necessary forensic investigation tools is necessary in order to get insights. In order to implement this a package called SIFT has been utilized which provides access to most of the forensics tools from one executable package. The forensic machine for this mentioned scenario has been prepared in the following manner.

An EC2 instance called “cloudreasearch-instance” is created and then after logging into it by doing SSH SIFT investigation tools are downloaded with the following commands.

```
ubuntu@ip-172-31-5-38:~$ sudo curl -Lo /usr/local/bin/sift https://github.com/sans-dfir/sift-cli/releases/download/v1.14.0-rc1/sift-cli-linux
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload Total Spent   Left Speed
100  147  100  147    0     0  630      0 --:--:-- --:--:-- --:--:-- 630
100  651  100  651    0     0 1299      0 --:--:-- --:--:-- --:--:-- 1299
100 55.0M 100 55.0M    0     0 9211k    0 0:00:06 0:00:06 --:--:-- 9942k
ubuntu@ip-172-31-5-38:~$ sudo chmod 755 /usr/local/bin/sift
ubuntu@ip-172-31-5-38:~$ sudo sift install
> sift-cl@1.14.0-rc1+0-g0582d2b
> sift-version: notinstalled
```

Figure 6.34 SIFT Installation

After installing SIFT tools a snapshot is created for the instance to perform forensic analysis on it. After creating snapshot, a volume is created from that snapshot and then attached to the earlier created EC2 instance.

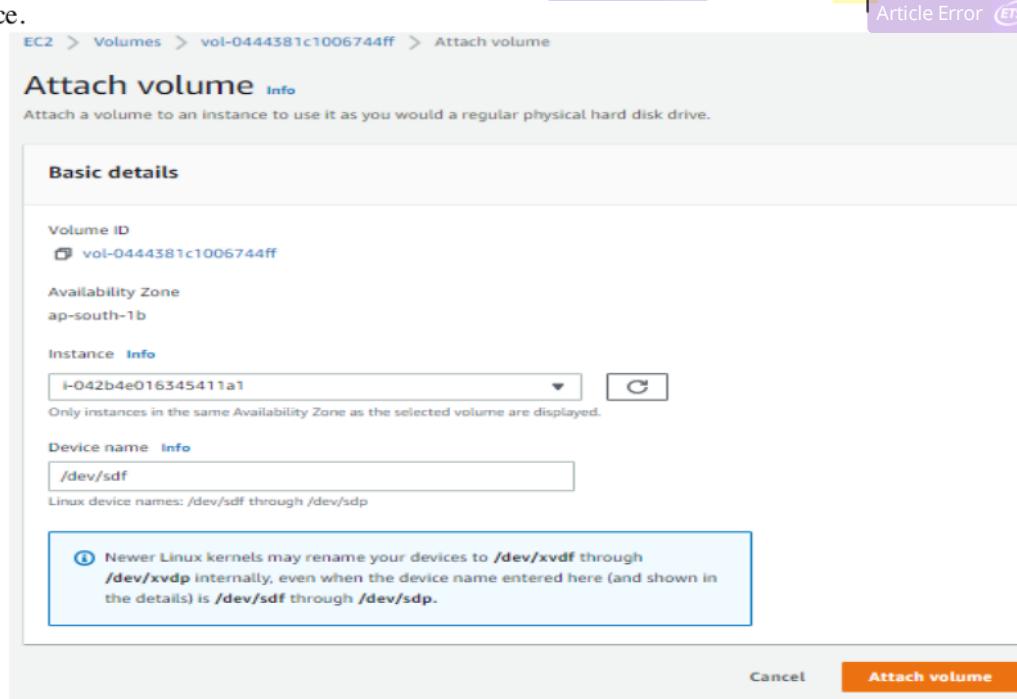


Figure 6.35 Attaching evidence volume to SIFT Workstation

Verifying evidence attached to a device using `lsblk` command.

```
ubuntu@ip-172-31-5-38:~$ sudo lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 42.2M  1 loop /snap/snapd/14066
loop1    7:1    0 55.5M  1 loop /snap/core18/2253
loop2    7:2    0  25M  1 loop /snap/amazon-ssm-agent/4046
xvda   202:0    0  30G  0 disk 
└─xvda1 202:1    0  30G  0 part /
xvdf   202:80   0  30G  0 disk 
└─xvdf1 202:81   0  30G  0 part
```

Figure 6.36 Evidence Attached Verification

Using the `file` command to determine the format of the partition as shown below and also a directory has been made to mount the evidentiary Linux file system as read-only:

```
ubuntu@ip-172-31-5-38:~$ sudo file -s /dev/xvdf1
/dev/xvdf1: Linux rev 1.0 ext4 filesystem data, UUID=c1ce24a2-4987-4450-ae15-62eb028ff1cd, volume name "cloudimg-rootfs" (needs journal recovery)
extents) (64bit) (large files) (huge files)
ubuntu@ip-172-31-5-38:~$ sudo mkdir /mnt/linux_mount
ubuntu@ip-172-31-5-38:~$ mount -o ro /dev/xvdf1 /mnt/linux_mount
mount: only root can use "--options" option
ubuntu@ip-172-31-5-38:~$ sudo mount -o ro /dev/xvdf1 /mnt/linux_mount/
ubuntu@ip-172-31-5-38:~$ sudo mount | grep "/mnt"
/mnt on /mnt/linux_mount type ext4 (ro,relatime)
```

Figure 6.37 Mounting evidentiary file on the system

Verifying the mounted data.

```
ubuntu@ip-172-31-5-38:~$ sudo ls -als /mnt/linux_mount/
total 124
4 drwxr-xr-x  24 root  root  4096 Apr  4 06:52 .
4 drwxr-xr-x  18 root  root  4096 Apr  4 07:11 ..
4 drwxr-xr-x  2 root  root  4096 Apr  4 06:41 bin
4 drwxr-xr-x  3 root  root  4096 Apr  4 06:49 boot
4 drwxrwxr-x  2 ubuntu root  4096 Apr  4 06:52 cases
4 drwxr-xr-x  4 root  root  4096 Nov 29 17:32 dev
12 drwxr-xr-x 155 root  root 12288 Apr  4 06:54 etc
4 drwxr-xr-x  3 root  root  4096 Apr  4 06:10 home
0 lrwxrwxrwx  1 root  root   30 Nov 29 17:39 initrd.img -> boot/initrd.img-5.4.0-1060-aws
0 lrwxrwxrwx  1 root  root   30 Nov 29 17:39 initrd.img.old -> boot/initrd.img-5.4.0-1060-aws
4 drwxr-xr-x  22 root  root  4096 Apr  4 06:16 lib
4 drwxr-xr-x  2 root  root  4096 Apr  4 06:16 lib64
16 drwx----- 2 root  root 16384 Nov 29 17:34 lost+found
4 drwxr-xr-x  2 root  root  4096 Nov 29 17:27 media
4 drwxr-xr-x  17 root  root  4096 Apr  4 06:52 mnt
4 drwxr-xr-x  4 root  root  4096 Apr  4 06:36 opt
4 drwxr-xr-x  2 root  root  4096 Apr 24 2018 proc
4 drwx-----  5 root  root  4096 Apr  4 06:52 root
4 drwxr-xr-x  5 root  root  4096 Nov 29 17:39 run
12 drwxr-xr-x  2 root  root 12288 Apr  4 06:48 sbin
4 drwxr-xr-x  6 root  root  4096 Apr  4 06:10 snap
4 drwxr-xr-x  2 root  root  4096 Nov 29 17:27 srv
4 drwxr-xr-x  2 root  root  4096 Apr 24 2018 sys
4 drwxrwxrwt 18 root  root  4096 Apr  4 06:56 tmp
4 drwxr-xr-x 12 root  root  4096 Apr  4 06:18 usr
4 drwxr-xr-x 14 root  root  4096 Apr  4 06:14 var
0 lrwxrwxrwx  1 root  root   27 Nov 29 17:39 vmlinuz -> boot/vmlinuz-5.4.0-1060-aws
0 lrwxrwxrwx  1 root  root   27 Nov 29 17:39 vmlinuz.old -> boot/vmlinuz-5.4.0-1060-aws
```

Figure 6.38 Listing data of mounted directory

After the evidence is attached to the SIFT Workstation, the initial step is to chisel data from the unallocated space and segregate out the files that are known to be good.

Another EC2 Instance is launched and based on the AMI and another snapshot is created and a volume is attached from the snapshot in the same availability zone as the SIFT Workstation. A different name called "HASH-BASELINE" for both the snapshot and the volume is assigned so that it is easy to differentiate these objects and the SIFT Workstation itself. Using the same steps as above the volume is attached and mounted as the 3rd volume on the SIFT Workstation which is named as /mnt/linux_base.

The screenshot shows the AWS EC2 Volumes page. At the top, there is a success message: "Successfully attached volume vol-02c60de7e3a2b28b2 to instance i-042b4e016345411a1". Below this, the "Volumes (4)" table is displayed. The columns are: Name, Volume ID, Type, Size, IOPS, Throughput, Snapshot, and Created. The data in the table is:

| Name | Volume ID | Type | Size | IOPS | Throughput | Snapshot | Created |
|------------------------|-----------------------|------|--------|------|------------|-----------------|-------------|
| clouddata-research-... | vol-07ae0dc02e80314d7 | gp2 | 30 GiB | 100 | - | snap-01559e0... | 2022/04/... |
| - | vol-0444381c1006744ff | gp2 | 30 GiB | 100 | - | snap-045fa65... | 2022/04/... |
| HASH-BASELINE | vol-02c60de7e3a2b28b2 | gp2 | 30 GiB | 100 | - | snap-06220b3... | 2022/04/... |
| clouddata-research-... | vol-0d456a51127c71684 | gp2 | 30 GiB | 100 | - | snap-01559e0... | 2022/04/... |

Figure 6.39 Newly Attached Volume to the instance

```
ubuntu@ip-172-31-5-38:~$ sudo mkdir /mnt/linux_base
ubuntu@ip-172-31-5-38:~$ sudo mount -o ro /dev/xvdg1 /mnt/linux_base/
ubuntu@ip-172-31-5-38:~$ sudo lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 42.2M  1 loop /snap/snapd/14066
loop1    7:1    0 55.5M  1 loop /snap/core18/2253
loop2    7:2    0   25M  1 loop /snap/amazon-ssm-agent/4046
xvda   202:0    0   30G  0 disk
└─xvda1 202:1    0   30G  0 part /
xvdf   202:80   0   30G  0 disk
└─xvdf1 202:81   0   30G  0 part /mnt/linux_mount
xvdg   202:96   0   30G  0 disk
└─xvdg1 202:97   0   30G  0 part /mnt/linux_base
ubuntu@ip-172-31-5-38:~$ |
```

Figure 6.40 Attaching and verifying additional mounted volume on SIFT Workstation

A hash database of all files on the reference volume is created using hfind which is called "known_files.md5" and in order to identify which files are new or modified an another hast list of files is created for the volume under investigation, this is called "investigate_files.md5" and with that list an names additional list of files that are new or changed are stored in "changed_files.txt".

Then in order to search for known indicators of compromise for the server instance an IOC scanner called “Loki” is installed which detects indicators of compromise Detection is based on four detection methods:

- File Names of IOC- which matches regular expressions of file names
- Yara Rule Check-matches signature of data and processes memory
- Hash Check – it compares harmful hashes like MD5, SHA1, SHA256 with scanned files

The following results were obtained when Loki detected indicators of compromise.

```
[NOTICE]
FILE: /mnt/linux_mount/usr/sbin/dsniff SCORE: 55 TYPE: ELF SIZE: 76496
FIRST_BYTES: 7f454c460201010000000000000000003003e00 / <filter object at 0x7f156823e160>
MDS: 7bae83a8ad173e438871aa9ddbd170f
SHA1: a59fcfa06ee62923bd70c7999208c50a14e3762
SHA256: cfe3941ce302f6dc7b13ee5c2780f800bdef009a068c65a5bbcfb19ac9f15be CREATED: Mon Apr 4 06:19:12 2022 MODIFIED: Fri Jul 21 18:05:34 2017 ACCESED: Mon Apr 4 06:19:12 2022
REASON_1: Yara Rule MATCH: HKTLDsniff SUBSCORE: 55
DESCRIPTION: Detects Dsniff hack tool REF: https://goo.gl/eFoP4A AUTHOR: Florian Roth
MATCHES: Str1: ".account.|.|.acct.|.|.domain.|.|.login.|.|.member.|."
[NOTICE]
FILE: /mnt/linux_mount/usr/sbin/webmit SCORE: 55 TYPE: ELF SIZE: 30720
FIRST_BYTES: 7f454c460201000000000000000000003003e00 / <filter object at 0x7f156823e128>
MDS: 53332776a1babadcef056e5f8abf22762
SHA1: b726aab0531eacc130809a5e9bfd94ebad03c1e8
SHA256: 777b361598119307cf92d6dbf272d8df7ccb8ca927352e4581246af81a6aff CREATED: Mon Apr 4 06:19:12 2022 MODIFIED: Fri Jul 21 18:05:34 2017 ACCESED: Mon Apr 4 06:19:12 2022
REASON_1: Yara Rule MATCH: HKTLDsniff SUBSCORE: 55
DESCRIPTION: Detects Dsniff hack tool REF: https://goo.gl/eFoP4A AUTHOR: Florian Roth
MATCHES: Str1: ".account.|.|.acct.|.|.domain.|.|.login.|.|.member.|."
```

Figure 6.41 Notice for indicators of compromise

```
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/mscoree.dll SCORE: 70 TYPE: EXE SIZE: 5236
FIRST_BYTES: 4d5a4000010000006000000fffff0000b800000 / <filter object at 0x7f1565d38668>
MDS: 8cb5ae3dab7578de39fa36cbe260f21f
SHA1: b726aab0531eacc130809a5e9bfd94ebad03c1e8
SHA256: a14c0ededb326fd24c220036fb06730c0db359adcf5a7e41d9f5a0b7faab8aa8 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCESED: Mon Apr 4 06:43:39 2022
REASON_1: File Name IOC matched PATTERN: /mscoree.dll SUBSCORE: 70 DESC: Unattributed ShadowPad Activity in Exchange Exploitation IOC https://www.welivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/rundl32.exe SCORE: 60 TYPE: EXE SIZE: 1032
FIRST_BYTES: 4d5a4000010000006000000fffff0000b800000 / <filter object at 0x7f1565d38630>
MDS: 35f92c16dcc3beb49f3142bcdff2874d1
SHA1: b0732939192a4e9a4c48886896f3d7abf50c4a6
SHA256: b90af2992fe8bf634ac07041695b5d790b167c15de737914aeeff69c3a4ddeb3f CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCESED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_rundl32.exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of rundl32.exe REF: - AUTHOR: Florian Roth
[ALERT]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/svchost.exe SCORE: 115 TYPE: EXE SIZE: 1032
FIRST_BYTES: 4d5a4000010000006000000fffff0000b800000 / <filter object at 0x7f1565d38550>
MDS: 7c20774d170cc400a78de2f2fd2d59ce
SHA1: a294a9e485c17d80bf68bb9571808b0994ea260d
SHA256: 556d962a414c1abaf3b7b6a64017e08e69fb6efb447fdb71b38fad755cdb3b68 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACCESED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_svchost.exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of svchost.exe REF: - AUTHOR: Florian Roth
REASON_2: Yara Rule MATCH: svchost_ANOMALY SUBSCORE: 55
DESCRIPTION: Abnormal svchost.exe - typical strings not found in file REF: - AUTHOR: Florian Roth
|-
```

Figure 6.42 Warnings and Alerts for Compromise of Indicators 1

Article Error 

```

WARNING] FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/mscoree.dll.so SCORE: 70 TYPE: ELF SIZE: 254160
FIRST_BYTES: 7f454c46020101000000000000000000000000003003e00 / <filter object at 0x7f1567173b38>
MD5: d95f52d43d3f73d5279998673cd3a71
SHA1: a2944e0459d4849e7eada82d467813e4872cb25maebdb7b785de083785393 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: File Name IOC matched PATTERN: /mscoree.dll SUBSCORE: 70 DESC: Unattributed Shadowpad Activity in Exchange Exploitation IOC https://www.olivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/spoolsv.exe SCORE: 60 TYPE: EXE SIZE: 1032
FIRST_BYTES: 465400000100000000000000000000000000000000000000 / <filter object at 0x7f1565d38590>
MD5: Tc20774d170c400a7da2d2f2ad2d59ce
SHA1: a2944e04537d89bf68b8080994ea260d
SHA256: 556d962a14c1abaf3b7b6a601e08e9fbefb447db1b38fad75cd8b68 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_spoolsv.exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of spoolsv.exe REF: - AUTHOR: Florian Roth
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/conhost.exe SCORE: 70 TYPE: EXE SIZE: 2484
FIRST_BYTES: 465400000100000000000000000000000000000000000000 / <filter object at 0x7f1565d385c0>
MD5: d50ecfc13fbaf9b825bdef12d2d3403a
SHA1: 6a52dc45f2e489730094a2554c2f9d4769ed2
SHA256: 1c7437953c17918ef931f940fbfa8cc0b7e451cc6b74b7104698a80204 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: conhost_ANOMALY SUBSCORE: 70
DESCRIPTION: Anomaly rule looking for certain strings in a system file (maybe false positive on certain systems) - file conhost.exe REF: not set AU
THOR: Florian Roth
[ALERT]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/explorer.exe SCORE: 115 TYPE: EXE SIZE: 6616
FIRST_BYTES: 465400000100000000000000000000000000000000000000 / <filter object at 0x7f1565d38618>
MD5: 9d4c9e1d89ca0ffefdd1e5d40789115aa37c
SHA1: e6b05313cc0e151d129673f937df450df856e7dd9f837a426fa7fadc197e49 CREATED: Mon Apr 4 06:43:39 2022 MODIFIED: Wed Jan 24 10:24:52 2018 ACES
SED: Mon Apr 4 06:43:36 2022
REASON_1: Yara Rule MATCH: Suspicious_Size_explorer.exe SUBSCORE: 60
DESCRIPTION: Detects uncommon file size of explorer.exe REF: - AUTHOR: Florian Roth
REASON_2: Yara Rule MATCH: explorer_ANOMALY SUBSCORE: 35
DESCRIPTION: Abnormal explorer.exe - typical strings not found in file REF: - AUTHOR: Florian Roth
[WARNING]
FILE: /mnt/linux_mount/usr/lib/x86_64-linux-gnu/wine/fakedlls/rundll32.exe SCORE: 60 TYPE: EXE SIZE: 1032

```

Figure 6.43 Warnings and Alerts for Compromise of Indicators 2

Article Error (ETS)

```

[NOTICE] FILE: /mnt/linux_mount/usr/local/lib/python3.6/dist-packages/pip/_vendor/distlib/_w64.exe SCORE: 50 TYPE: EXE SIZE: 99840
FIRST_BYTES: 4d5400000100000000000000000000000000000000000000 / <filter object at 0x7f1567173c50>
MD5: 0655a0ef42ff9bf591f614ba8f5721f
SHA1: b10653dce173109af61b686ccca65be816f3c4
SHA256: diaf73aa0dd31bd35650810dcbbff78bc790747a994c564d5520925894e0d32 CREATED: Mon Apr 4 06:48:14 2022 MODIFIED: Mon Apr 4 06:48:13 2022 ACCESSED: Mon Apr 4 06:48:14 2022
REASON_1: File Name IOC matched PATTERN: /w64.exe SUBSCORE: 50 DESC: Cred Dumping
[NOTICE] [!] 11 alerts detected, after filtering warnings, 11 notices
[NOTICE] [!] 11 notices detected
[RESULT] Loki recommends checking the elements on virustotal.com or Google and triage with a professional tool like THOR https://nextron-systems.com/thor in corporate networks.
[VIDEO] Please report false positives via https://github.com/Neo23x0/signature-base
[NOTICE] Finished LOKI Scan SYSTEM: siftworkstation TIME: 20220404T10:16:13Z
ubuntu@ip-172-31-5-38:/tmp/Loki-0.44.2 $

```

Figure 6.44 Final Results of Loki

6.4.1 Additional Forensic Analysis

Some malware or anomaly makes use of the start-up scripts that are initiated when the system is started and runs at boot time. On some distributions, these are found in /etc/init.d, but on Amazon Linux and Red Hat variants, the scripts will be in /etc/rc*.d.

```
ubuntu@ip-172-31-5-38:~$ ls -als -t /mnt/linux_mount/etc/rc*.d/
/mnt/linux_mount/etc/rc0.d/:
total 16
12 drwxr-xr-x 155 root root 12288 Apr  4 06:54 ..
4 drwxr-xr-x  2 root root  4096 Apr  4 06:43 .
0 Trwxrwxrwx  1 root root   17 Apr  4 06:43 K01winbind -> ../init.d/winbind
0 Trwxrwxrwx  1 root root   15 Apr  4 06:42 K01saned -> ../init.d/saned
0 Trwxrwxrwx  1 root root   22 Apr  4 06:42 K01avahi-daemon -> ../init.d/avahi-daemon
0 Trwxrwxrwx  1 root root   19 Apr  4 06:42 K01bluetooth -> ../init.d/bluetooth
0 Trwxrwxrwx  1 root root   18 Apr  4 06:40 K01stunnel4 -> ../init.d/stunnel4
0 Trwxrwxrwx  1 root root   21 Apr  4 06:39 K01samba-ad-dc -> ../init.d/samba-ad-dc
0 Trwxrwxrwx  1 root root   14 Apr  4 06:39 K01nmbd -> ../init.d/nmbd
0 Trwxrwxrwx  1 root root   14 Apr  4 06:39 K01smbd -> ../init.d/smbd
0 Trwxrwxrwx  1 root root   27 Apr  4 06:35 K01speech-dispatcher -> ../init.d/speech-dispatcher
0 Trwxrwxrwx  1 root root   16 Apr  4 06:32 K01fdump -> ../init.d/ndfdump
0 Trwxrwxrwx  1 root root   20 Apr  4 06:32 K01nbd-client -> ../init.d/nbd-client
0 Trwxrwxrwx  1 root root   16 Apr  4 06:18 K01docker -> ../init.d/docker
0 Trwxrwxrwx  1 root root   26 Apr  4 06:17 K01clamav-freshclam -> ../init.d/clamav-freshclam
0 Trwxrwxrwx  1 root root   29 Apr  4 06:14 K01apache-htcacheload -> ../init.d/apache-htcacheload
0 Trwxrwxrwx  1 root root   17 Apr  4 06:14 K01apache2 -> ../init.d/apache2
0 Trwxrwxrwx  1 root root   23 Nov 29 17:31 K01lvm2-lvmpolld -> ../init.d/lvm2-lvmpolld
0 Trwxrwxrwx  1 root root   22 Nov 29 17:31 K01lvm2-lvmetad -> ../init.d/lvm2-lvmetad
0 Trwxrwxrwx  1 root root   13 Nov 29 17:31 K01lxd -> ../init.d/lxd
0 Trwxrwxrwx  1 root root   23 Nov 29 17:31 K01open-vm-tools -> ../init.d/open-vm-tools
0 Trwxrwxrwx  1 root root   18 Nov 29 17:31 K01plymouth -> ../init.d/plymouth
0 Trwxrwxrwx  1 root root   20 Nov 29 17:31 K01cryptdisks -> ../init.d/cryptdisks
0 Trwxrwxrwx  1 root root   26 Nov 29 17:31 K01cryptdisks-early -> ../init.d/cryptdisks-early
0 Trwxrwxrwx  1 root root   20 Nov 29 17:31 K01irqbalance -> ../init.d/irqbalance
0 Trwxrwxrwx  1 root root   15 Nov 29 17:31 K01lxcsf -> ../init.d/lxcsf
0 Trwxrwxrwx  1 root root   29 Nov 29 17:31 K01unattended-upgrades -> ../init.d/unattended-upgrades
0 Trwxrwxrwx  1 root root   18 Nov 29 17:31 K01ebscripts -> ../init.d/ebscripts
0 Trwxrwxrwx  1 root root   15 Nov 29 17:31 K01uidd -> ../init.d/uidd
0 Trwxrwxrwx  1 root root   15 Nov 29 17:31 K01mdadm -> ../init.d/mdadm
0 Trwxrwxrwx  1 root root   24 Nov 29 17:31 K01mdadm-waitidle -> ../init.d/mdadm-waitidle
0 Trwxrwxrwx  1 root root   20 Nov 29 17:31 K01open-iscsi -> ../init.d/open-iscsi
0 Trwxrwxrwx  1 root root   16 Nov 29 17:31 K01iscsid -> ../init.d/iscsid
0 Trwxrwxrwx  1 root root   13 Nov 29 17:31 K01atd -> ../init.d/atd
0 Trwxrwxrwx  1 root root   17 Nov 29 17:27 K01rsyslog -> ../init.d/rsyslog
```

Figure 6.45 Startup Scripts

Looking for unusual files can be a hectic task, so in order to make it easy a security expert looks for SUID and SGID files (SUID Files - SUID is a special file permission for executable files which enables other users to run the file with effective permissions of the file owner while SGID Files - SGID is a special file permission that also applies to executable files and enables other users to inherit the effective GID of file group owner).The following commands perform the comparison on mounted volume for evidence capturing.

```

ubuntu@ip-172-31-5-38:~$ sudo find /mnt/linux_mount/ -uid 0 -perm -4000 -print > suid_evidence
ubuntu@ip-172-31-5-38:~$ sudo
usage: sudo [-h | -K | -k | -V
usage: sudo [-e [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo [-Y [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-ADEHknPs] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo [-e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...
ubuntu@ip-172-31-5-38:~$ sudo
usage: sudo [-h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-ADEHknPs] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...
ubuntu@ip-172-31-5-38:~$ sudo find /mnt/linux_base/ -uid 0 -perm -4000 -print > suid_base
ubuntu@ip-172-31-5-38:~$ sudo cut suid_base -d"/" -f4 -> suid_base_relative
ubuntu@ip-172-31-5-38:~$ sudo cut suid_base -d"/" -f4 -> suid_evidence_relative
ubuntu@ip-172-31-5-38:~$ sudo diff suid_base_relative suid_evidence_relative
ubuntu@ip-172-31-5-38:~$ ls
Desktop  changed_files.txt  investigate_files.md5-md5.idx  known_files.md5-md5.idx2  output  suid_base_relative  suid_evidence_relative
changed_md5  investigate_files.md5  known_files.md5-md5.idx2  known_files.md5-md5.idx  Toki_0.44.2.zip  suid_base  suid_evidence
suid_base_relative
ubuntu@ip-172-31-5-38:~$ vim suid_base_relative
ubuntu@ip-172-31-5-38:~$ vim suid_base_relative
ubuntu@ip-172-31-5-38:~$ vim suid_evidence_relative
ubuntu@ip-172-31-5-38:~|
```

Figure 6.46 Commands to look for unusual files

```

ubuntu@ip-172-31-5-38: ~
bin/mount
bin/fusermount
bin/umount
bin/ping
bin/su
usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
usr/lib/eject/decrypt-get-device
usr/lib/snapd/snap-confine
usr/lib/polkit-1/polkit-agent-helper-1
usr/lib/dbus-1.0/dbus-daemon-launch-helper
usr/lib/openssh/ssh-keysign
usr/bin/chsh
usr/bin/chfn
usr/bin/sudo
usr/bin/newgrp
usr/bin/traceroute6.iputils
usr/bin/newuidmap
usr/bin/passwd
usr/bin/gpasswd
usr/bin/pkexec
usr/bin/newgidmap
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

Figure 6.47 List of Unusual files

In order to look for files with high entropy there is a tool in SIFT called DensityScout which detects packing, compression, and encrypted files that exceed a “density” threshold. The following commands are implemented in order to find such files which exceed the threshold.

```
ubuntu@ip-172-31-5-38:~$ sudo densityscout -r -p 0.1 -l 0.1 -o high_density_evidence.txt /mnt/linux_mount/
DensityScout (Build 45)
by Christian Wojner

Calculating density for file ...
(0.08665) | /mnt/linux_mount/usr/share/man/man1/gawk.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/x86_64-linux-gnu-gcc-7.1.gz
(0.07464) | /mnt/linux_mount/usr/share/man/man1/wget.1.gz
(0.08366) | /mnt/linux_mount/usr/share/man/man1/socat.1.gz
(0.05668) | /mnt/linux_mount/usr/share/man/man1/xterm.1.gz
(0.09947) | /mnt/linux_mount/usr/share/man/man1/less.1.gz
(0.09091) | /mnt/linux_mount/usr/share/man/man1/sh.distrib.1.gz
(0.09165) | /mnt/linux_mount/usr/share/man/man1/keytool.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/g++-7.1.gz
(0.09927) | /mnt/linux_mount/usr/share/man/man1/git-fast-import.1.gz
(0.04133) | /mnt/linux_mount/usr/share/man/man1/g++.1.gz
(0.07959) | /mnt/linux_mount/usr/share/man/man1/x86_64-linux-gnu-ld.bfd.1.gz
(0.07728) | /mnt/linux_mount/usr/share/man/man1/gpg.1.gz
(0.09091) | /mnt/linux_mount/usr/share/man/man1/sh.1.gz
(0.07959) | /mnt/linux_mount/usr/share/man/man1/ld.1.gz
(0.08236) | /mnt/linux_mount/usr/share/man/man1/git-log.1.gz
(0.05229) | /mnt/linux_mount/usr/share/man/man1/bash.1.gz
(0.08055) | /mnt/linux_mount/usr/share/man/man1/cli.1.gz
(0.09974) | /mnt/linux_mount/usr/share/man/man1/find.1.gz
```

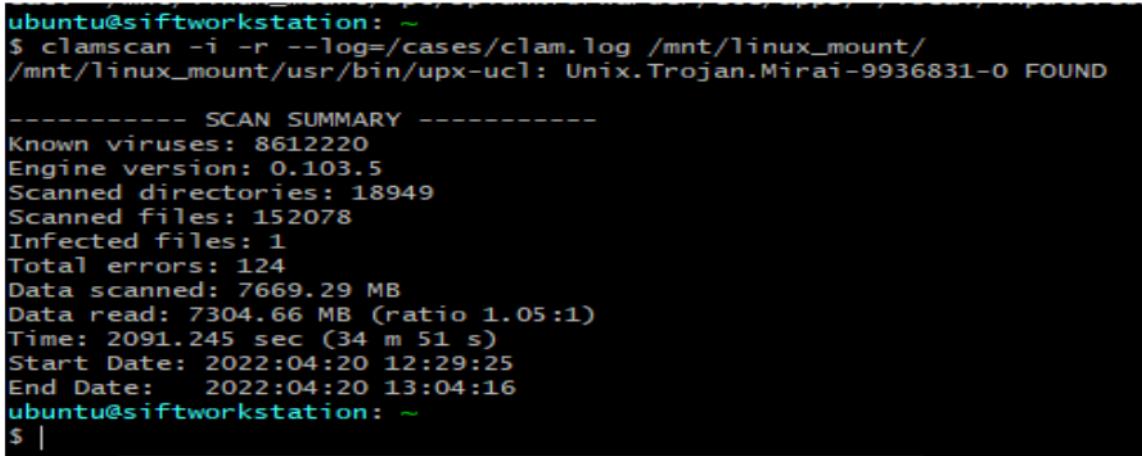
Figure 6.48 High Entropy files in /mnt/linux_mount (Volume where SIFT is installed)

```
ubuntu@ip-172-31-5-38:~$ sudo densityscout -r -p 0.1 -l 0.1 -o high_density_base.txt /mnt/linux_base/
DensityScout (Build 45)
by Christian Wojner

Calculating density for file ...
(0.08665) | /mnt/linux_base/usr/share/man/man1/gawk.1.gz
(0.07464) | /mnt/linux_base/usr/share/man/man1/wget.1.gz
(0.09947) | /mnt/linux_base/usr/share/man/man1/less.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/sh.distrib.1.gz
(0.09927) | /mnt/linux_base/usr/share/man/man1/git-fast-import.1.gz
(0.07728) | /mnt/linux_base/usr/share/man/man1/gpg.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/sh.1.gz
(0.08236) | /mnt/linux_base/usr/share/man/man1/git-log.1.gz
(0.05229) | /mnt/linux_base/usr/share/man/man1/bash.1.gz
(0.09974) | /mnt/linux_base/usr/share/man/man1/find.1.gz
(0.09947) | /mnt/linux_base/usr/share/man/man1/pager.1.gz
(0.09091) | /mnt/linux_base/usr/share/man/man1/dash.1.gz
(0.08665) | /mnt/linux_base/usr/share/man/man1/nawk.1.gz
(0.07303) | /mnt/linux_base/usr/share/man/man1/rsync.1.gz
(0.07669) | /mnt/linux_base/usr/share/man/man1/top.1.gz
(0.08718) | /mnt/linux_base/usr/share/man/man1/tmux.1.gz
(0.08665) | /mnt/linux_base/usr/share/man/man1/awk.1.gz
(0.06611) | /mnt/linux_base/usr/share/man/man1/screen.1.gz
(0.07075) | /mnt/linux_base/usr/share/man/man1/git-config.1.gz
(0.08041) | /mnt/linux_base/usr/share/man/man1/curl.1.gz
(0.07633) | /mnt/linux_base/usr/share/man/man1/busybox.1.gz
(0.07022) | /mnt/linux_base/usr/share/man/man3/pcrepattern.3.gz
(0.08504) | /mnt/linux_base/usr/share/man/es/man8/dnsmasq.8.gz
(0.09558) | /mnt/linux_base/usr/share/man/man7/systemd.directives.7.gz
(0.09402) | /mnt/linux_base/usr/share/man/man7/mdoc.samples.7.gz
```

Figure 6.49 High Entropy files in /mnt/linux_base (Additional mounted volume containing forensic evidence)

Clamscan is a malware scanner that comes loaded when we install SIFT Workstation on our instances, it is used to scan all data and infected files present on the system.



```
ubuntu@siftworkstation: ~
$ clamscan -i -r --log=/cases/clam.log /mnt/linux_mount/
/mnt/linux_mount/usr/bin/upx-ucl: Unix.Trojan.Mirai-9936831-0 FOUND

----- SCAN SUMMARY -----
Known viruses: 8612220
Engine version: 0.103.5
Scanned directories: 18949
Scanned files: 152078
Infected files: 1
Total errors: 124
Data scanned: 7669.29 MB
Data read: 7304.66 MB (ratio 1.05:1)
Time: 2091.245 sec (34 m 51 s)
Start Date: 2022:04:20 12:29:25
End Date: 2022:04:20 13:04:16
ubuntu@siftworkstation: ~
$ |
```

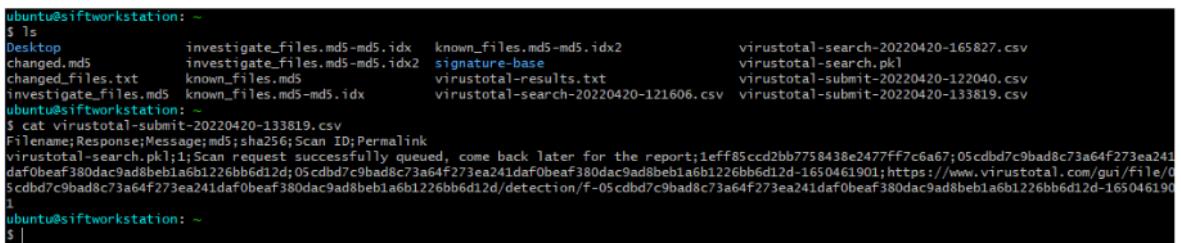
Figure 6.50 Clamscan Results

At this stage, the forensic investigator must have realized that there must be multiple files which are infected, in order to check which files are infected third party hash hook-ups from online websites like virustotal.com can provide more clearer results to the investigator. After executing the below mentioned commands on SIFT workstation multiple csv files will be generated containing links of virustotal which can be pasted on the website to get better view of infected files present.

Sp. ETS

- virustotal-search.py investigate_files.md5 > virustotal-results.txt
- virustotal-submit.py virustotal-search.pkl

Sent Specified Job. ETS



```
ubuntu@siftworkstation: ~
$ ls
Desktop      investigate_files.md5-md5.idx    known_files.md5-md5.idx2          virustotal-search-20220420-165827.csv
changed.md5   investigate_files.md5-md5.idx2  signature-base                  virustotal-search.pkl
changed_files.txt known_files.md5               virustotal-results.txt          virustotal-submit-20220420-122040.csv
investigate_files.md5 known_files.md5-md5.idx  virustotal-search-20220420-121606.csv virustotal-submit-20220420-133819.csv
ubuntu@siftworkstation: ~
$ cat virustotal-submit-20220420-133819.csv
Filename;Response;Message;md5;sha256;Scan ID;Permalink
virustotal-search.pkl;1;Scan request successfully queued, come back later for the report;1eff85cc2bb7758438e2477ff7c6a67;05cdbd7c9bad8c73a64f273ea241da0beaf380dac9ad8beb1a6b1226bb6d12d;05cdbd7c9bad8c73a64f273ea241da0beaf380dac9ad8beb1a6b1226bb6d12d;1650461901;https://www.virustotal.com/gui/file/05cdbd7c9bad8c73a64f273ea241da0beaf380dac9ad8beb1a6b1226bb6d12d/detection/f-05cdbd7c9bad8c73a64f273ea241da0beaf380dac9ad8beb1a6b1226bb6d12d-1650461901
1
ubuntu@siftworkstation: ~
$ |
```

Figure 6.51 Linking Virustotal.com

The link generated from multiple .csv files as shown above will give details about the infected files present on the EC2 instance.

Prep. (ETS)

Sentence Cap. (ETS)

This screenshot shows the VirusTotal analysis interface for a URL. The top bar indicates "0 / 92" detections as malicious. The URL analyzed is <https://www.virustotal.com/gui/file/05cd8d7c9bad8c73a64f273ea241da0beaf380dac9ad8be>. The analysis results show 200 Status, text/html; charset=utf-8 Content Type, and a timestamp of 2022-04-21 12:52:18 UTC, "a moment ago". Below this, there are tabs for DETECTION, DETAILS, and COMMUNITY. The DETECTION tab displays a table titled "Security Vendors' Analysis" with 10 rows of data. All entries are marked as "Clean".

| Vendor | Status | Vendor | Status |
|---------------------|--------|-------------------|--------|
| Abusix | Clean | Acronis | Clean |
| ADMINUSLabs | Clean | AIIC (MONITORAPP) | Clean |
| AlienVault | Clean | alphaMountain.ai | Clean |
| Antly-AVL | Clean | Armis | Clean |
| Artists Against 419 | Clean | Avira | Clean |

Figure 6.52 Third Party Hash Hook-ups 1

This screenshot shows the VirusTotal analysis interface for a file. The top bar indicates "57 / 69" detections as malicious. The file analyzed is `newbs2.exe`. The analysis results show 749.50 KB Size, and a timestamp of 2021-03-20 23:10:47 UTC, "6 months ago". Below this, there are tabs for DETECTION, DETAILS, RELATIONS, BEHAVIOR, and COMMUNITY. The DETECTION tab displays a table with 10 rows of data, each with a vendor name, detection status, and a detailed description. Most detections are marked as "Suspicious" or "Malicious".

| Vendor | Status | Description |
|---------------------|----------------------------|----------------------|
| Acronis (Static ML) | Suspicious | Ad-Aware |
| AhnLab-V3 | Trojan/Win32.Tepfer.R49363 | Alibaba |
| ALYac | Trojan.Kryptik.gen | SecureAge APEX |
| Arcabit | Trojan.Honret.2 | Avast |
| AVG | Win32/Kryptik-LAO [Tr] | Avira (non-priority) |
| Baidu | | |
| Cynet | | |
| Cloudmark | | |
| Comodo | | |
| Comodo Positive | | |

Figure 6.53 Third Party Hash Hook-ups 2

CHAPTER 7 CONCLUSION AND FUTURE WORK

Conclusion

In conclusion, the number of cases and the severity, refinement of malware attacks and expense of malware infect is increasing at an alarming rate. Malware should be detected as early as possible and mitigated. In this project, cybersecurity and security in-depth principles are applied to the cloud environment. These propositions indicate that defense controls of the cloud environment will not succeed at numerous cases and an attack might prevail so the companies must have response mechanisms to put off these attacks. Log monitoring and digital forensics gathering are the main trait for investigators for tracking and detecting active malware attacks. In this project 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. After that, we validated the applicability and limitation of deploying this baseline by doing a malware attack Any type of malicious activity which might takes place on the cloud account can be mitigated if the data is monitored properly. A baseline is built on AWS using a service called AWS CloudTrail which generated logs of activities taking place within S3. and then they were integrated with Splunk which is a SIEM Tool to perform investigation and analysis and take some steps regarding attack decision. Splunk provided data correlation, enrichment, integration with other security events, and long-term storage. Lastly in order to investigate the vulnerability of VMs Investigations were performed on the compromised IaaS VMs which displayed how a user should be careful and alert of the vulnerability of the system and take necessary steps to prevent it in future.

Future Work

As there are ample number of malware attacks happening day by day which are very difficult to track whether it is on-premises or on cloud environment, security management and investigation techniques should be given more value as the data uploaded on these environments is very important leading to changes to world economy at some stages. For future work, we believe that cloud services sources should provide the necessary tools for executing volatile memory analysis for their VMs. Also, develop a new automated tool for incident response and forensics investigation on the IaaS.

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>
- [8] J. Dykstra , "Digital forensics for infrastructure-as-a-service cloud computing," Ph.D dissertation, Faculty of the Graduate School of the University of Maryland, Baltimore County, 2013.
- [9] A. Pichan, M. Lazarescu and S. T. Soh, "Cloud forensics: Technical challenges, solutions and comparative analysis," Digital Investigation ,Elsevier, Vol.13, pp. 38-57, 23 March 2015.

G20 IBM PROJECT REPORT

ORIGINALITY REPORT



PRIMARY SOURCES

| | | |
|---|--|---------------|
| 1 | thesai.org Internet Source | 3% |
| 2 | Submitted to Ganpat University Student Paper | 2% |
| 3 | www.coursehero.com Internet Source | 1% |
| 4 | Submitted to University of Western Australia Student Paper | 1% |
| 5 | Lama Almadhoor, A. A. bd El-Aziz, Hedi Hamdi. "Detecting Malware Infection on Infrastructure Hosted in IaaS Cloud using Cloud Visibility and Forensics", International Journal of Advanced Computer Science and Applications, 2021 Publication | <1% |
| 6 | 1library.net Internet Source | <1% |

Exclude quotes Off

Exclude matches Off

Exclude bibliography On

G20 IBM PROJECT REPORT

PAGE 1

 **Article Error** You may need to use an article before this word.

 **Article Error** You may need to remove this article.

 **Article Error** You may need to use an article before this word.

 **Missing ","** You may need to place a comma after this word.

 **Article Error** You may need to use an article before this word.

PAGE 2

PAGE 3

PAGE 4

PAGE 5

 **Missing ","** You may need to place a comma after this word.

PAGE 6

PAGE 7

 **Article Error** You may need to remove this article.

 **Article Error** You may need to use an article before this word. Consider using the article a.

 **Wrong Form** You may have used the wrong form of this word.

 **Missing ","** You may need to place a comma after this word.

 **Run-on** This sentence may be a run-on sentence. Proofread it to see if it contains too many independent clauses or contains independent clauses that have been combined without conjunctions or punctuation. Look at the "Writer's Handbook" for advice about correcting run-on sentences.



Article Error You may need to remove this article.

PAGE 8



Article Error You may need to remove this article.



Missing "," You may need to place a comma after this word.

PAGE 9



Verb This verb may be incorrect. Proofread the sentence to make sure you have used the correct form of the verb.

PAGE 10

PAGE 11

PAGE 12



Article Error You may need to use an article before this word. Consider using the article **the**.



Article Error You may need to use an article before this word.

PAGE 13



Article Error You may need to use an article before this word.

PAGE 14



S/V This subject and verb may not agree. Proofread the sentence to make sure the subject agrees with the verb.



Article Error You may need to use an article before this word.

PAGE 15



Sp. This word is misspelled. Use a dictionary or spellchecker when you proofread your work.



Article Error You may need to use an article before this word. Consider using the article **the**.



Missing "," You may need to place a comma after this word.



Wrong Form You may have used the wrong form of this word.



Article Error You may need to use an article before this word. Consider using the article **the**.



Article Error You may need to use an article before this word.

PAGE 16



Article Error You may need to use an article before this word.

PAGE 17



Article Error You may need to use an article before this word.



Confused You have a spelling mistake near the word **an** that makes **an** appear to be a confused-word error.

PAGE 18



Missing "," You have a spelling or typing mistake that makes the sentence appear to have a comma error.

PAGE 19



Sentence Cap. Remember to capitalize the first word of each sentence.

PAGE 20



Article Error You may need to use an article before this word.

PAGE 21



P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.

PAGE 22



Article Error You may need to use an article before this word. Consider using the article **the**.

PAGE 23



P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.



Confused You have a spelling mistake near the word **an** that makes **an** appear to be a confused-word error.



Hyph. You may need to add a hyphen between these two words.



Confused You have used **a** in this sentence. You may need to use **an** instead.



Confused You have a spelling mistake near the word **an** that makes **an** appear to be a confused-word error.



Confused You have used **of** in this sentence. You may need to use **off** instead.

PAGE 24



Wrong Article You may have used the wrong article or pronoun. Proofread the sentence to make sure that the article or pronoun agrees with the word it describes.



S/V This subject and verb may not agree. Proofread the sentence to make sure the subject agrees with the verb.



Missing "," You have a spelling or typing mistake that makes the sentence appear to have a comma error.

PAGE 25

PAGE 26



Sp. This word is misspelled. Use a dictionary or spellchecker when you proofread your work.

PAGE 27



P/V You have used the passive voice in this sentence. Depending upon what you wish to emphasize in the sentence, you may want to revise it using the active voice.



Article Error You may need to remove this article.



Article Error You may need to remove this article.

PAGE 28



Sp. This word is misspelled. Use a dictionary or spellchecker when you proofread your work.



Article Error You may need to remove this article.



Article Error You may need to remove this article.



Article Error You may need to use an article before this word.



Article Error You may need to use an article before this word.



Verb This verb may be incorrect. Proofread the sentence to make sure you have used the correct form of the verb.



Faulty Comparison You cannot use these two words together to form a comparison. You can delete "more" or "most" to correct this mistake.



Sp. This word is misspelled. Use a dictionary or spellchecker when you proofread your work.



Frag. This sentence may be a fragment or may have incorrect punctuation. Proofread the sentence to be sure that it has correct punctuation and that it has an independent clause with a complete subject and predicate.



Sentence Cap. Remember to capitalize the first word of each sentence.



Sp. This word is misspelled. Use a dictionary or spellchecker when you proofread your work.



Prep. You may be using the wrong preposition.



Sentence Cap. Remember to capitalize the first word of each sentence.

PAGE 37

PAGE 38
