# CW2 Report

Repository: https://github.com/Karmakze/3010---CW2-Splunk-Joseph-Troughton

## 1 - Introduction

#### 1.1 - Context

This report presents a review of the **Boss of the SOC Version 3 (BOTSv3)** dataset. The dataset is designed to introduce users to the operation of **Security Operations Centres (SOCs)** and highlight their critical role in monitoring, detecting, analysing, and responding to cyber threats within modern enterprise environments. Organisations rely on SOC analysts to interpret large volumes of log data, identify anomalous behaviour, and apply incident response methodologies to mitigate risk.

BOTSv3 provides a realistic example of how security professionals develop these skills by offering a highly detailed dataset that records a simulated cyber incident. The dataset was created by Splunk as a public, Splunk-based Capture The Flag (CTF) exercise designed to test and improve the analytical abilities of cybersecurity professionals.

The scenario simulates a complex attack against a fictional brewing company named **Frothly**, which operates cloud-based services alongside on-premises systems and web infrastructure. The dataset includes a wide range of log sources such as AWS CloudTrail logs, S3 access logs, endpoint monitoring data, and system hardware information, reflecting the diverse telemetry sources commonly analysed in a real SOC environment.

The aim of this report is to investigate selected AWS and endpoint-focused BOTSv3 questions using Splunk, demonstrating how SOC analysts identify identity misuse, insecure cloud configurations, and anomalous endpoint behaviour. The scope of this investigation is limited to answering the assigned question set and reflecting on SOC roles and incident handling practices, rather than conducting a full end-to-end kill chain analysis.

## 2 - SOC Roles & Incident Handling Reflection

### 2.1 - Roles

#### 2.1.1 - Tier 1: Triage Specialist

#### Tier 1 analysts are responsible for the initial collection and review of raw security data, including alerts and alarms, to identify suspicious activity. In the context of the BOTSv3 dataset, this role is reflected through the detection of anomalous AWS activity using CloudTrail logs. Tier 1 analysts are required to prioritise alerts based on severity while remaining vigilant against alert fatigue.

#### An example from this investigation is the detection of the PutBucketAcl API call that granted public access to an S3 bucket. This event represents a high-severity alert due to the risk of data exposure and would be prioritised for escalation. Identifying the IAM user responsible for this action also falls within Tier 1 responsibilities when determining whether further investigation is required. This demonstrates how Tier 1 analysts act as the first line of defence within a SOC.

#### 2.1.2 - Tier 2: Incident Responder

Tier 2 analysts are responsible for conducting in-depth investigations into high-priority alerts escalated by Tier 1. This involves correlating multiple data sources to determine the scope, impact, and intent of an incident. Tier 2 analysts confirm whether a true security incident has occurred and assess the associated risk to the organisation.

Within the BOTSv3 investigation, Tier 2 responsibilities are demonstrated through the correlation of AWS CloudTrail events with S3 access logs to assess the impact of the public bucket misconfiguration. The discovery of the file OPEN\_BUCKET\_PLEASE\_FIX.txt being uploaded into the bucket confirms external interaction following the misconfiguration, escalating the incident severity. Additional analysis, such as confirming the IAM user involved and identifying endpoint systems with unusual configurations, further reflects Tier 2 investigative workflows.

#### 2.1.3 - Tier 3: Threat Hunter

### Tier 3 analysts represent the highest level of technical escalation within a SOC and focus on proactive threat hunting and the development of detection mechanisms to prevent recurring incidents. Their responsibilities include identifying security gaps, improving detection coverage, and responding to advanced or previously unknown threats.

### In the context of BOTSv3, Tier 3 activity would involve developing Splunk alerts to detect S3 bucket ACL changes that grant public access and monitoring AWS API activity performed without multi-factor authentication (MFA). The use of the userIdentity.sessionContext.attributes.mfaAuthenticated field enables the detection of insecure identity behaviour. Implementing such detections would significantly reduce real-world detection times and improve overall SOC response capability.

### 

### 2.2 Incident Handling Reflection

## This investigation aligns with established incident handling frameworks such as NIST. The **preparation phase** is represented by the installation and configuration of Splunk and the ingestion of BOTSv3 data. **Detection and analysis** are demonstrated through the identification of cloud misconfigurations, IAM misuse, and endpoint anomalies using targeted SPL queries. While BOTSv3 does not simulate remediation actions, in a real SOC environment the **containment, eradication, and recovery** phases would involve removing public access permissions, enforcing MFA, and validating affected systems. The investigation highlights the importance of continuous monitoring and proactive detection in reducing organisational risk.

## 3 Installation & Data Preparation

### 3.1 Environment Overview

Splunk was downloaded and deployed within an ubuntu 24.04 virtual machine using Virt-Manager with KVM as shown in figure 1 Below.

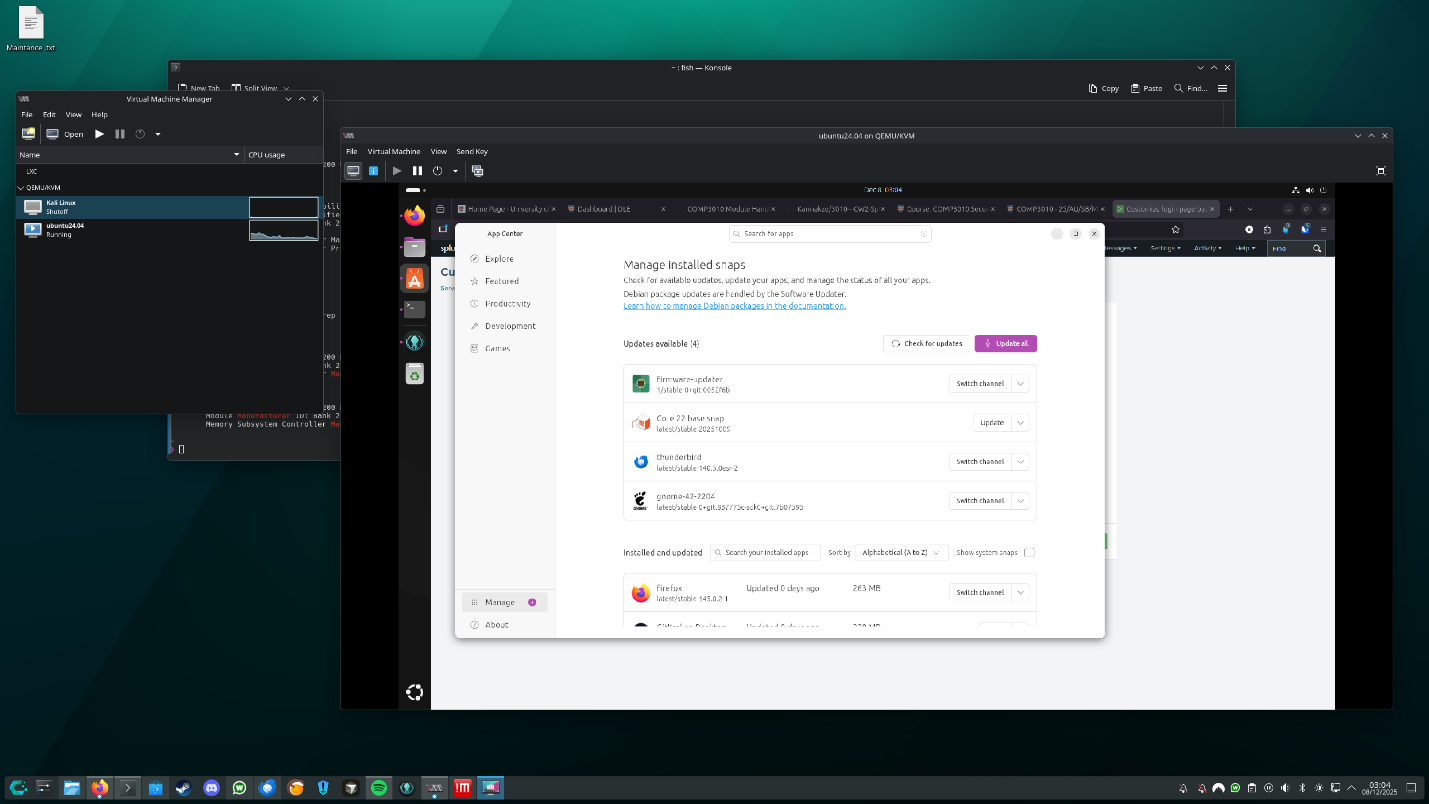


Figure 1

This machine was configured with 8192MB of memory, 6 virtual CPUs, a single 65gb disk, internet and was accessed with the Spice display manager, these specification exceed what is required to support large-scale log ingestion an analysis but contributes to a smoother experience. I chose this method over VMWare as even though I am familiar with both and had VMWare installed, I found VMWare an to run less consistently than Virt-Manager on my system which runs Linux where Virt-Manager is far more supported.

### 3.2 Splunk Installation

The Splunk Package was installed using the official Linux .tgz package obtained from the link given in by my lecturer. This installation was performed though CLI instead of GUI shown in figure 2 below which is important practice for real world administrative processes where a GUI may not be available.



Figure 2

Navigated to its location in Gnome file manager to see that it was correctly installed as shown in figure 3. Command to install Splunk was then run successfully and Splunk was installed, configured and connected to as show in figure 4 & 5.

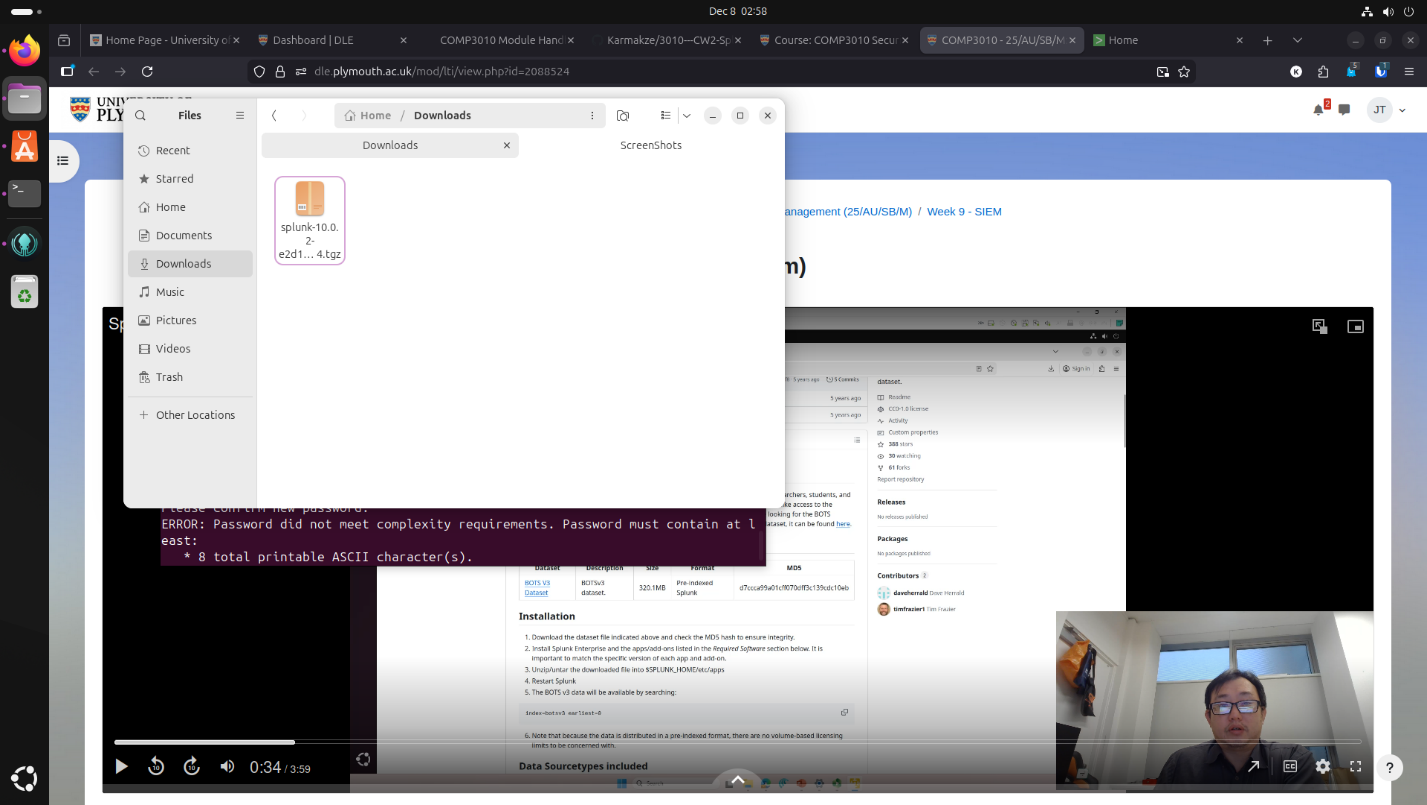


Figure 3

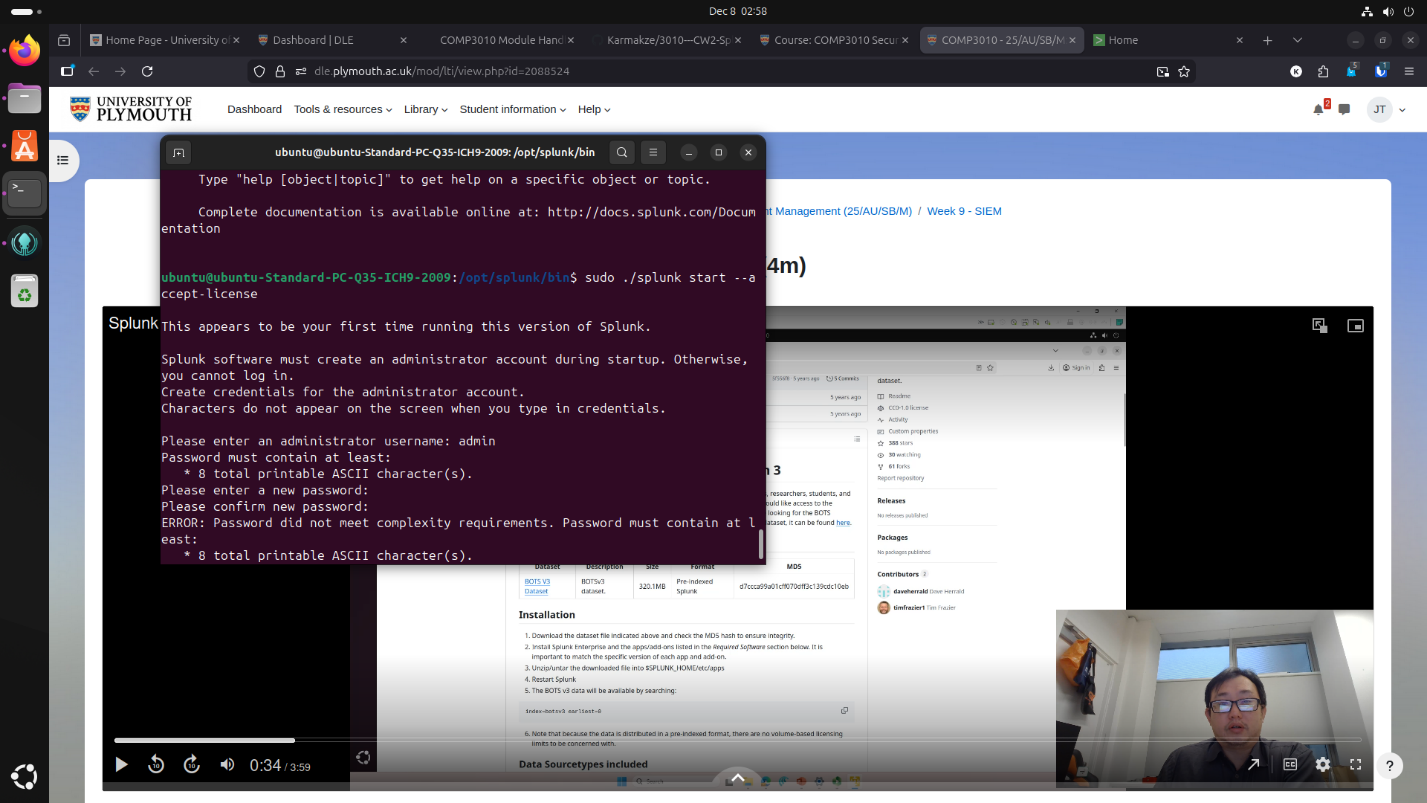


Figure 4

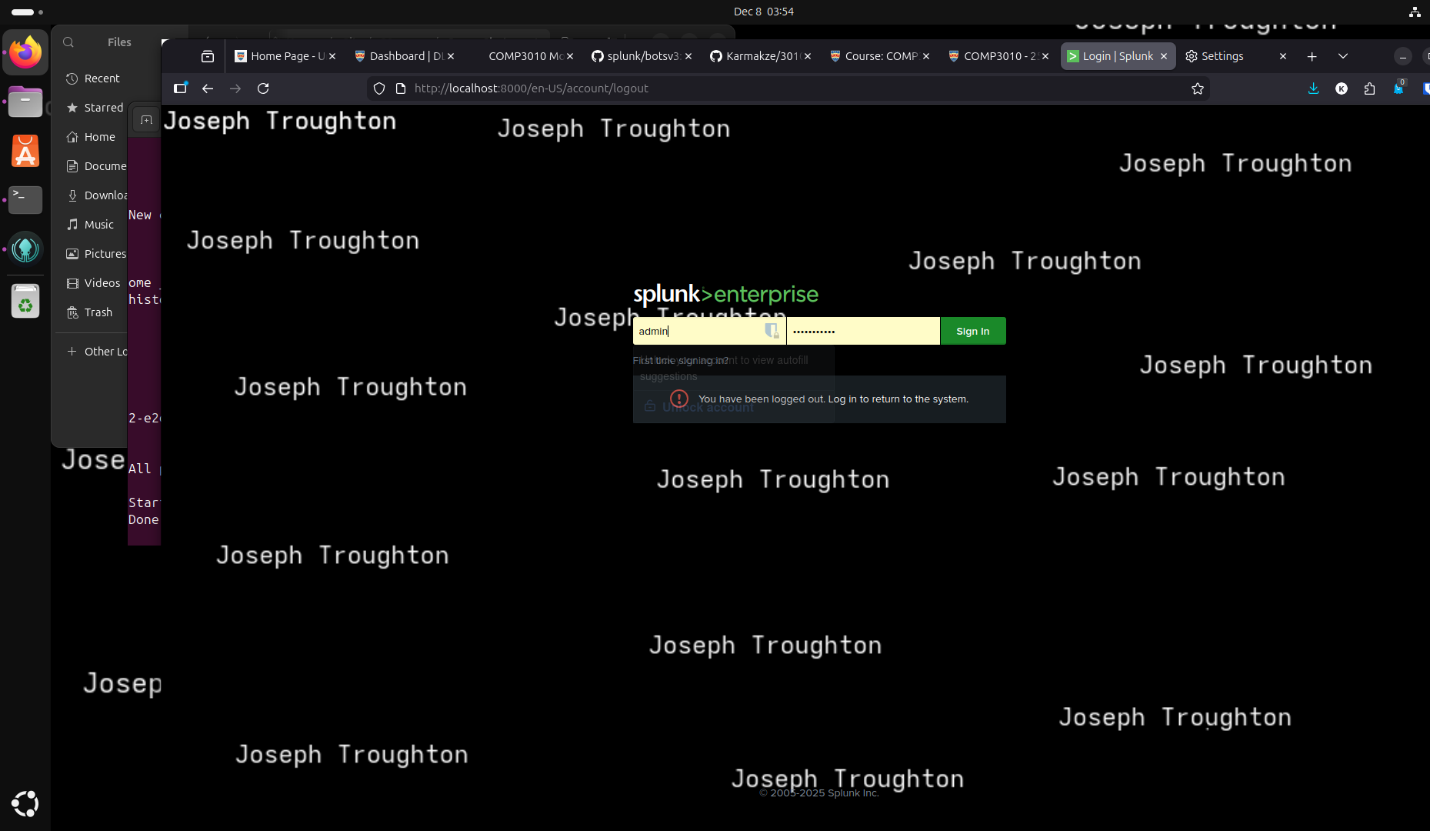


Figure 5

### 3.3 BOTSv3 Dataset Acquisition

The BOTSv3 was then downloaded from the official repository, then navigated using Gnome’s file manager as shown in Figure 7 I move it using the file manager (ctrl + x) as well instead of the “mv” command by adding “admin://“ to the root of the file name shown in Figure 4 and took a picture of it in the new location at /opt/splunk/etc/apps as shown in figure 7.

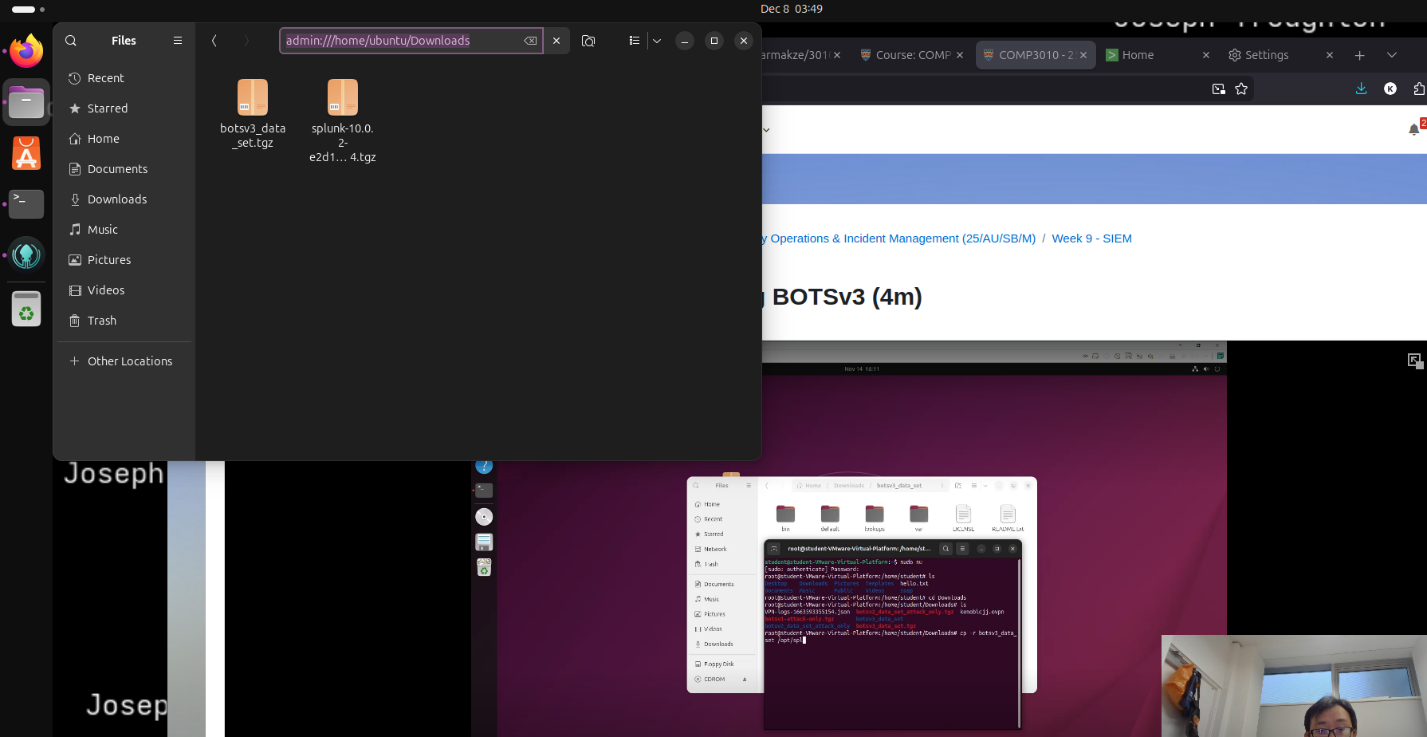


Figure 6

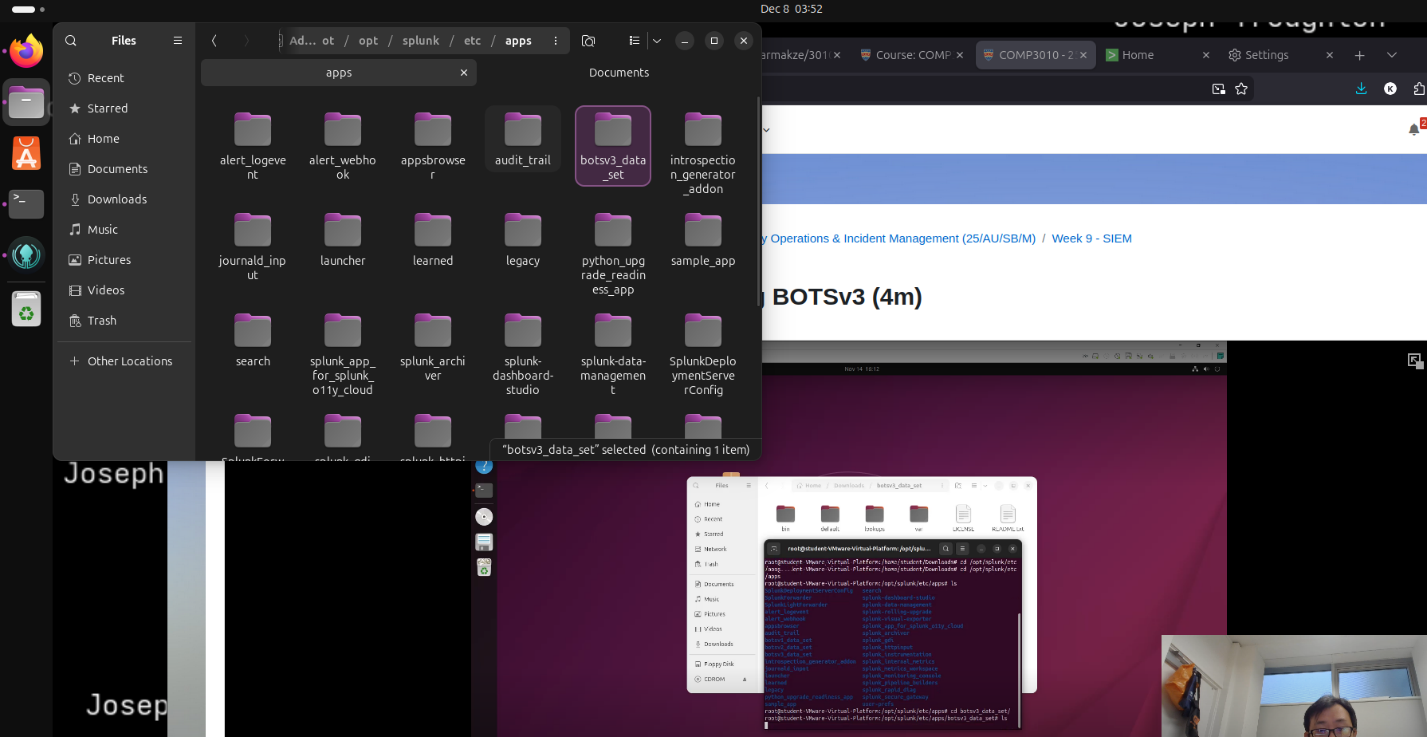


Figure 7

Now the BOTSv3 Dataset was ingested it could be accessed via indexing with the command ‘index=”botsv3”’ if this command successfully searches and bring up the dataset and crosscheck the amount of event to the requirement “2,083,056” I could confirm I had gotten the right dataset and added it successfully as shown below in figure 8.

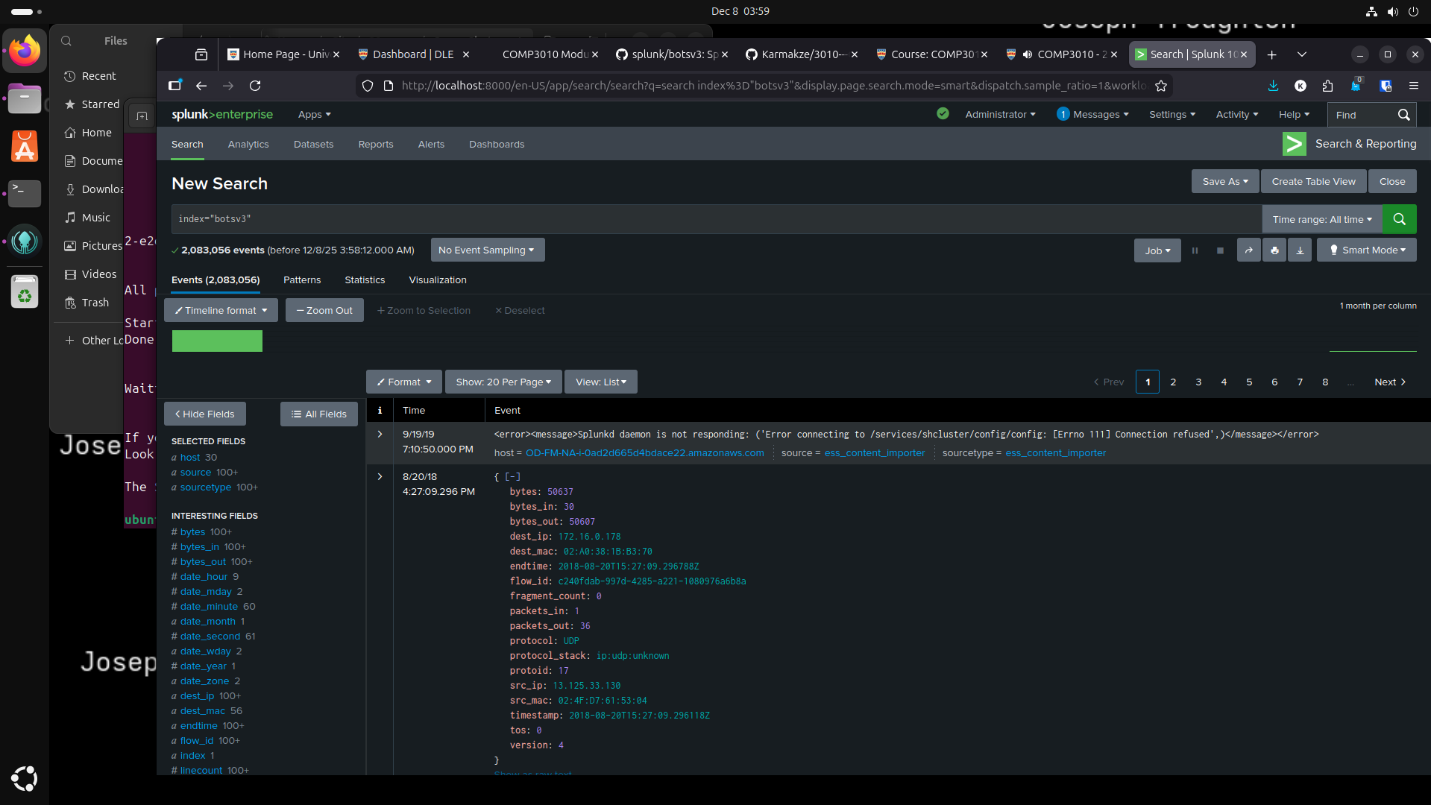


Figure 8

## 4 – Guided Questions: AWS and Endpoint Investigation

This section documents the investigative process used to answer the BOTSv3 guided questions. While the final answers were submitted via the DLE quiz, this section provides justification, methodology, and supporting evidence demonstrating correct data source selection, query logic, and SOC relevance.

### 4.1 Identification of IAM Users Accessing AWS Services

Objective:Identify IAM users that accessed AWS services within Frothly’s AWS environment.

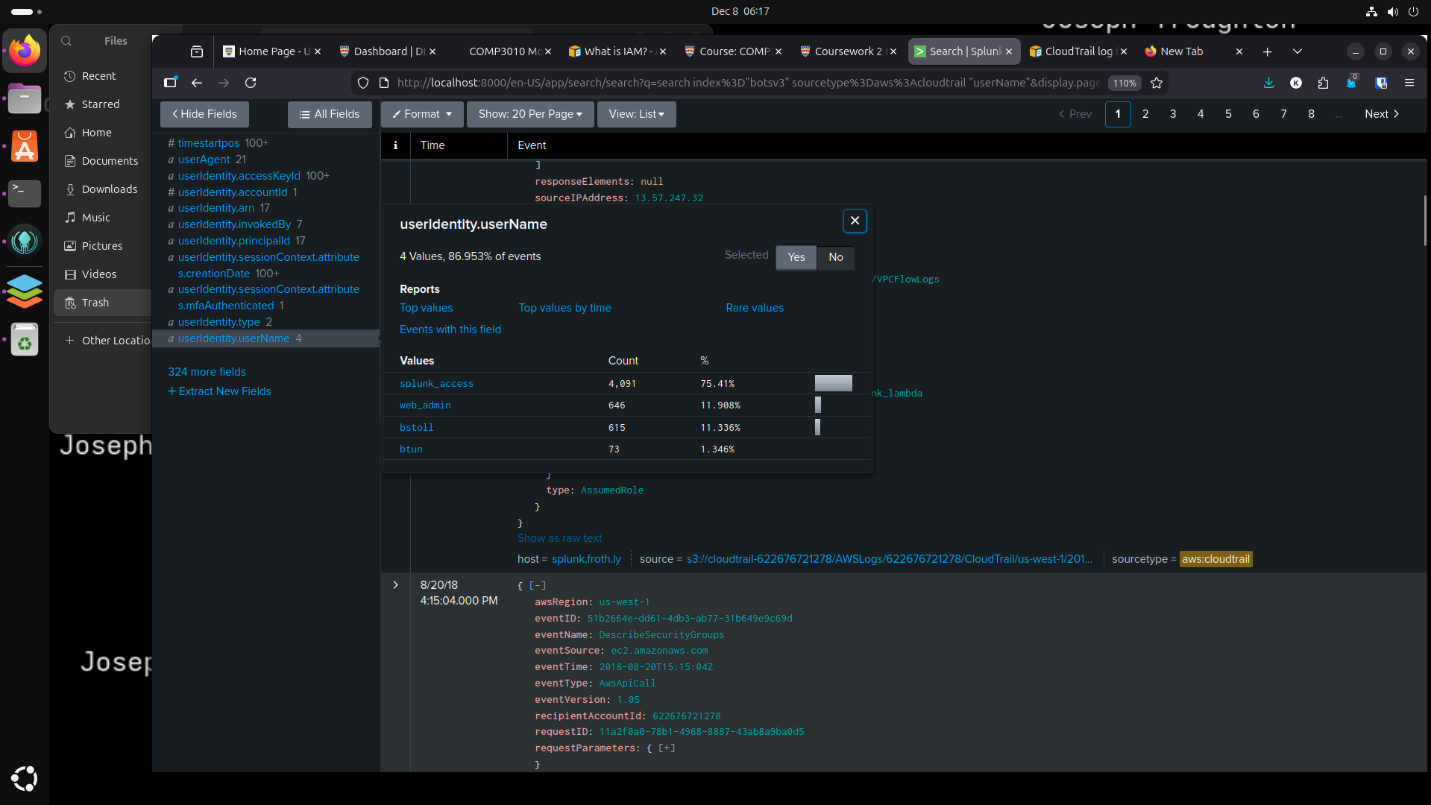
#### SPL:

**‘index="botsv3" sourcetype=aws:cloudtrail’**

Methodology and Justification:  
AWS CloudTrail logs record all API activity performed within an AWS account, including both successful and unsuccessful requests. The field userIdentity.userName identifies the IAM user responsible for each API call. By examining this field across all CloudTrail events, it is possible to enumerate every IAM identity that interacted with AWS services.

This approach aligns with SOC best practices, as identity-based analysis is a core method for detecting unauthorised access, compromised credentials, or misuse of privileges.

Answer:  
The following IAM users were observed accessing AWS services: bstoll, btun, splunk\_access, web\_admin



### 4.2 Detection of AWS API Activity Without MFA

Objective:  
Identify the field used to detect AWS API activity performed without multi-factor authentication (MFA).

#### SPL:

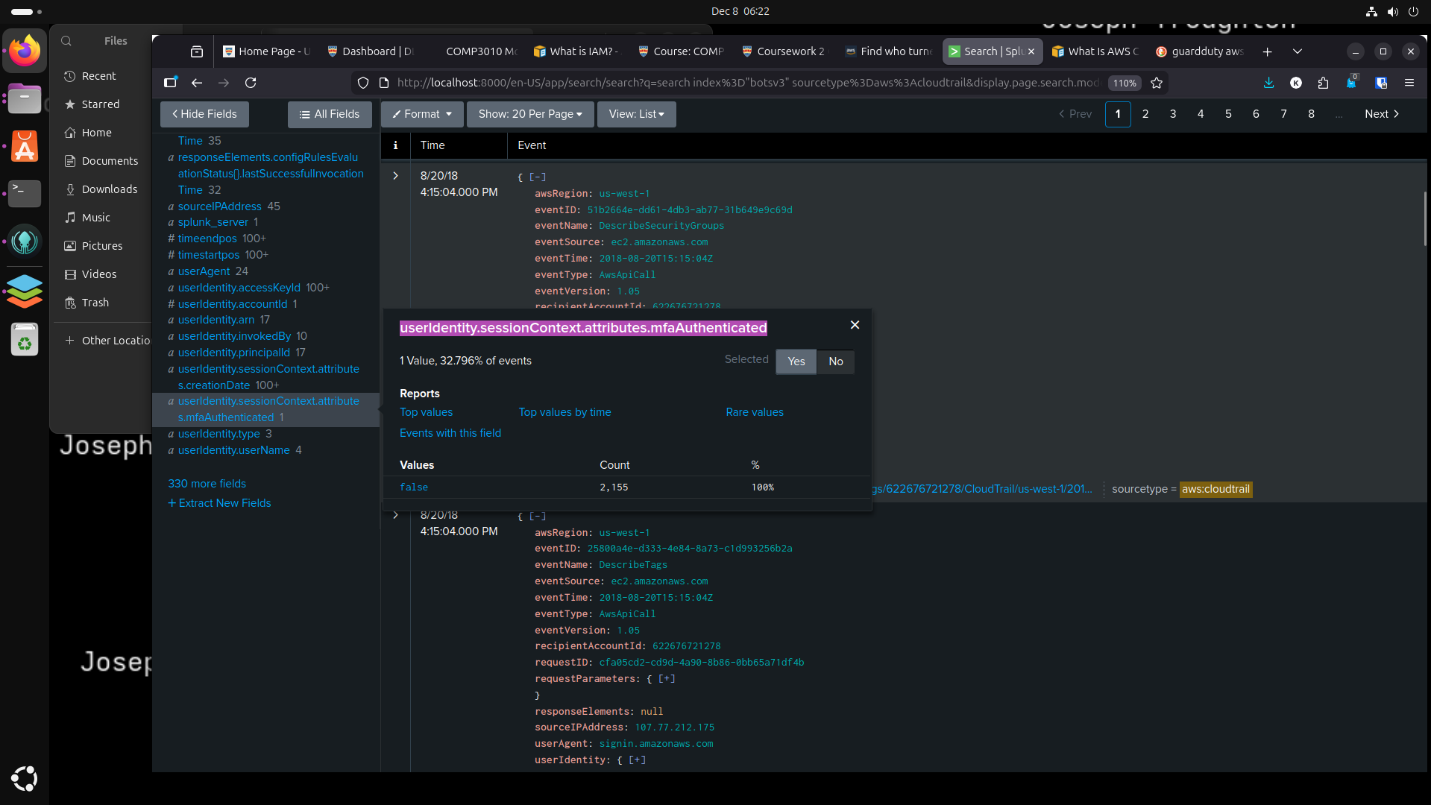
**‘index="botsv3" sourcetype=aws:cloudtrail’**

Methodology and Justification:  
CloudTrail records session authentication attributes within the userIdentity.sessionContext.attributes object. The mfaAuthenticated field explicitly indicates whether MFA was used during the session. This field is commonly used in SOC environments to detect risky API activity and enforce security policies.

Events related to console logins were excluded to ensure the focus remained on API-driven activity, as required by the question. Monitoring this field enables SOC teams to detect policy violations, credential abuse, and elevated risk activity where MFA protections are bypassed.

#### Answer:

userIdentity.sessionContext.attributes.mfaAuthenticated



### 4.3 Identification of Web Server Processor Type

Objective:  
Determine the processor model used on the web servers

#### SPL:

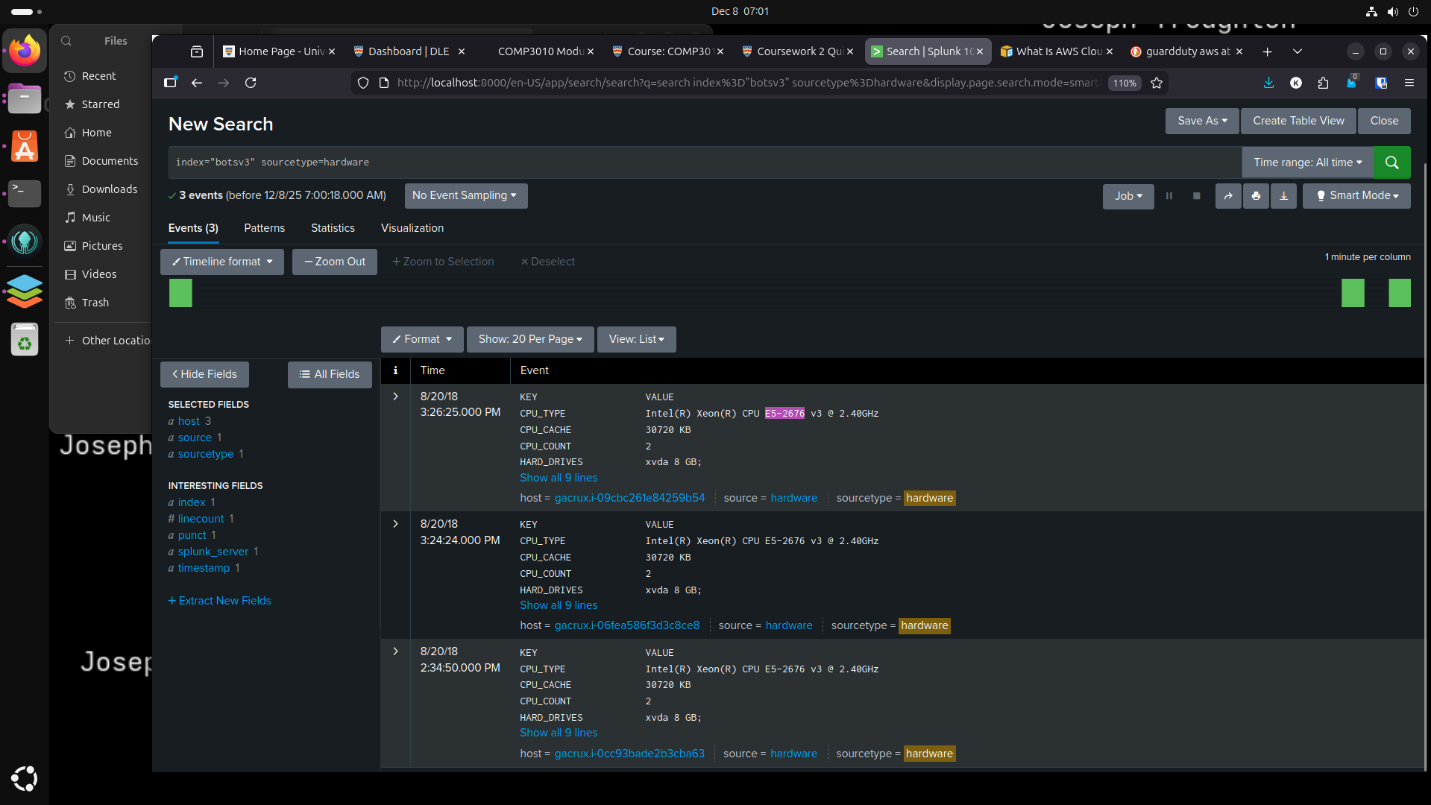
**‘index="botsv3" sourcetype=hardware’**

Methodology and Justification:  
The hardware sourcetype contains system inventory data, including CPU information. Reviewing the CPU\_TYPE field across events revealed a consistent processor model across the web servers.

Hardware baselining is important in SOC environments to detect anomalies such as unauthorised virtual machines or unexpected infrastructure changes.

#### Answer:

E5-2676



### 4.4 Identification of Public S3 Bucket Misconfiguration

Objective:  
Identify the API event that made an S3 bucket publicly accessible.

#### SPL:

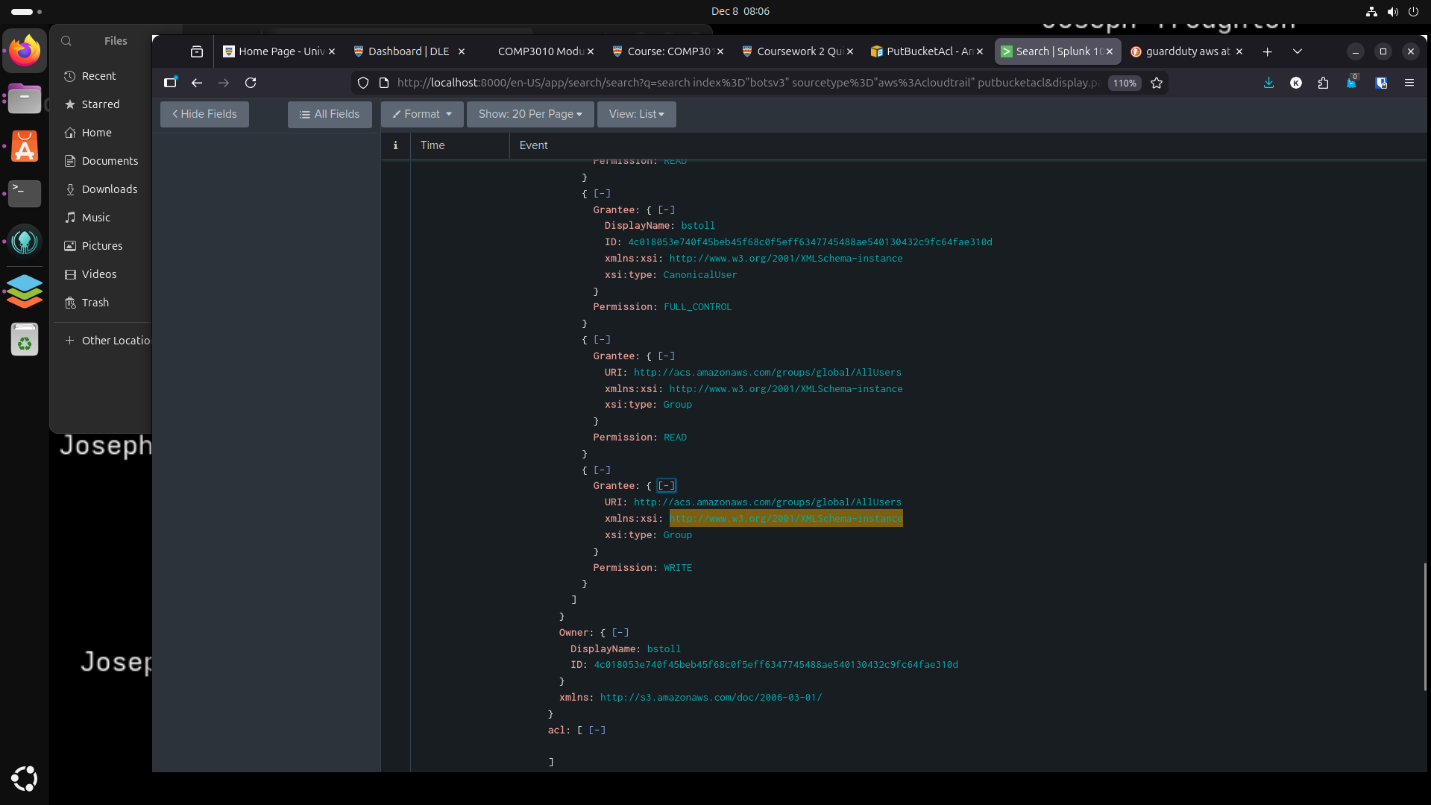
**‘index="botsv3" sourcetype="aws:cloudtrail" putbucketacl’**

Methodology and Justification:  
The PutBucketAcl API call is used to modify access control lists (ACLs) on S3 buckets. Examination of the requestParameters.AccessControlPolicy field showed permissions being granted to the AllUsers group with both READ and WRITE access, confirming that the bucket was made publicly accessible.

This represents a common cloud misconfiguration and is a high-severity finding in SOC investigations.

#### Answer:

9a33d8df-1e16-4d58-b36d-8e80ce68f8a3



### 4.5 Identification of Responsible IAM User

Objective:  
Identify the IAM user responsible for making the S3 bucket public.

#### SPL:

**‘index="botsv3" sourcetype="aws:cloudtrail" putbucketacl’**

Methodology and Justification:  
The userIdentity.userName field within the PutBucketAcl event identifies the IAM user who performed the action. Show in the image above in section 4.4 under the owner tab.

Attribution is critical in incident response to determine whether an action was accidental, malicious, or the result of credential compromise.

#### Answer:

bstoll

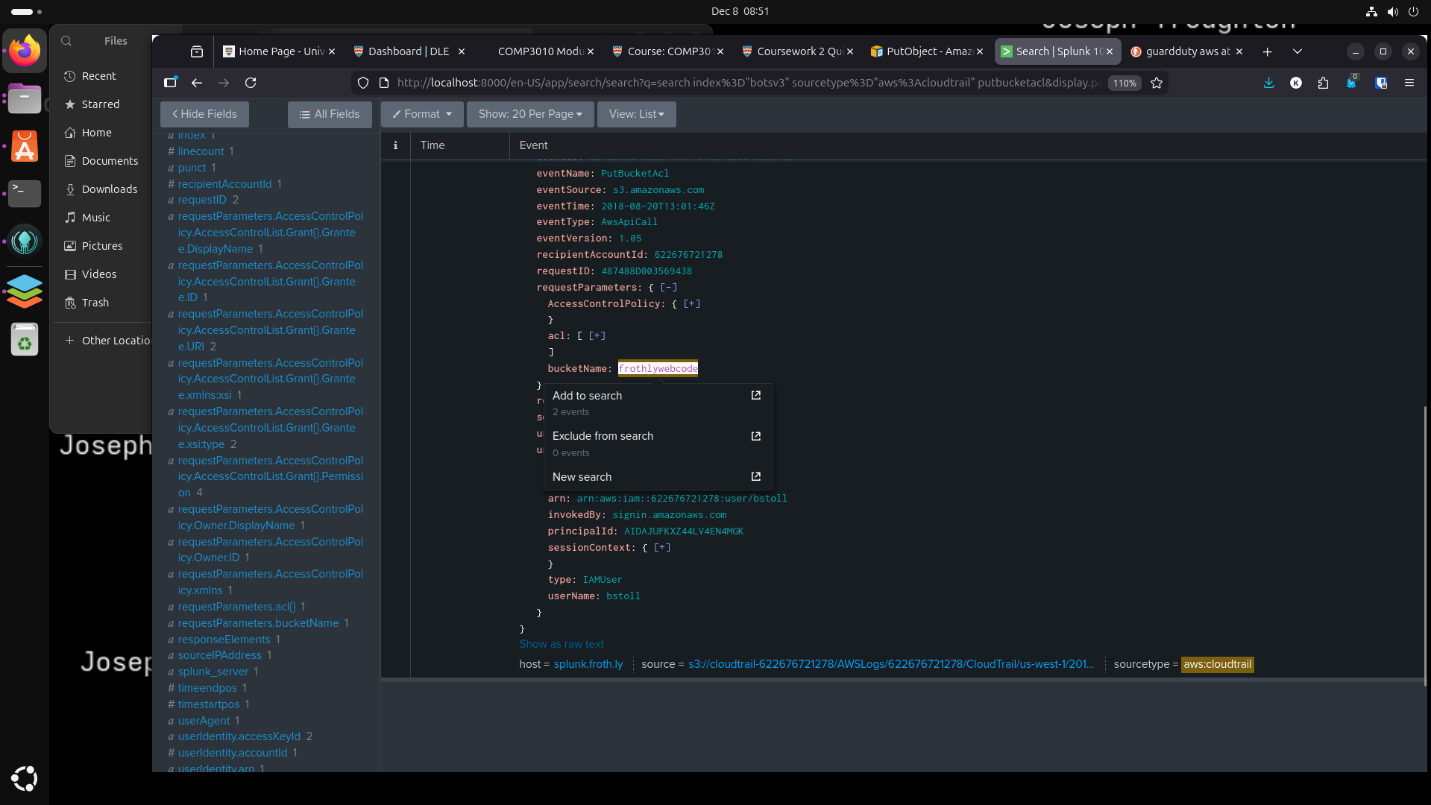
### 4.6 Identification of the Publicly Accessible S3 Bucket

Objective:  
Identify the name of the S3 bucket made publicly accessible

Methodology and Justification:  
The requestParameters.bucketName field within the PutBucketAcl event specifies which bucket was modified.

#### Answer:

frothlywebcode



### 4.7 Identification of Uploaded File to Public S3 Bucket

Objective:  
Identify a text file uploaded while the bucket was publicly accessible.

#### SPL:

**‘index="botsv3" "frothlywebcode" .txt’**

#### Methodology and Justification:

S3 access logs record object-level activity, including uploads. Filtering for HTTP 200 responses and REST.PUT.OBJECT actions confirmed successful file uploads.

This confirms that the misconfiguration had real impact, escalating the incident severity.

#### Answer:

OPEN\_BUCKET\_PLEASE\_FIX.txt

### 4.8 Identification of Endpoint OS Anomaly

Objective:  
Identify an endpoint running a different Windows OS edition

#### SPL:

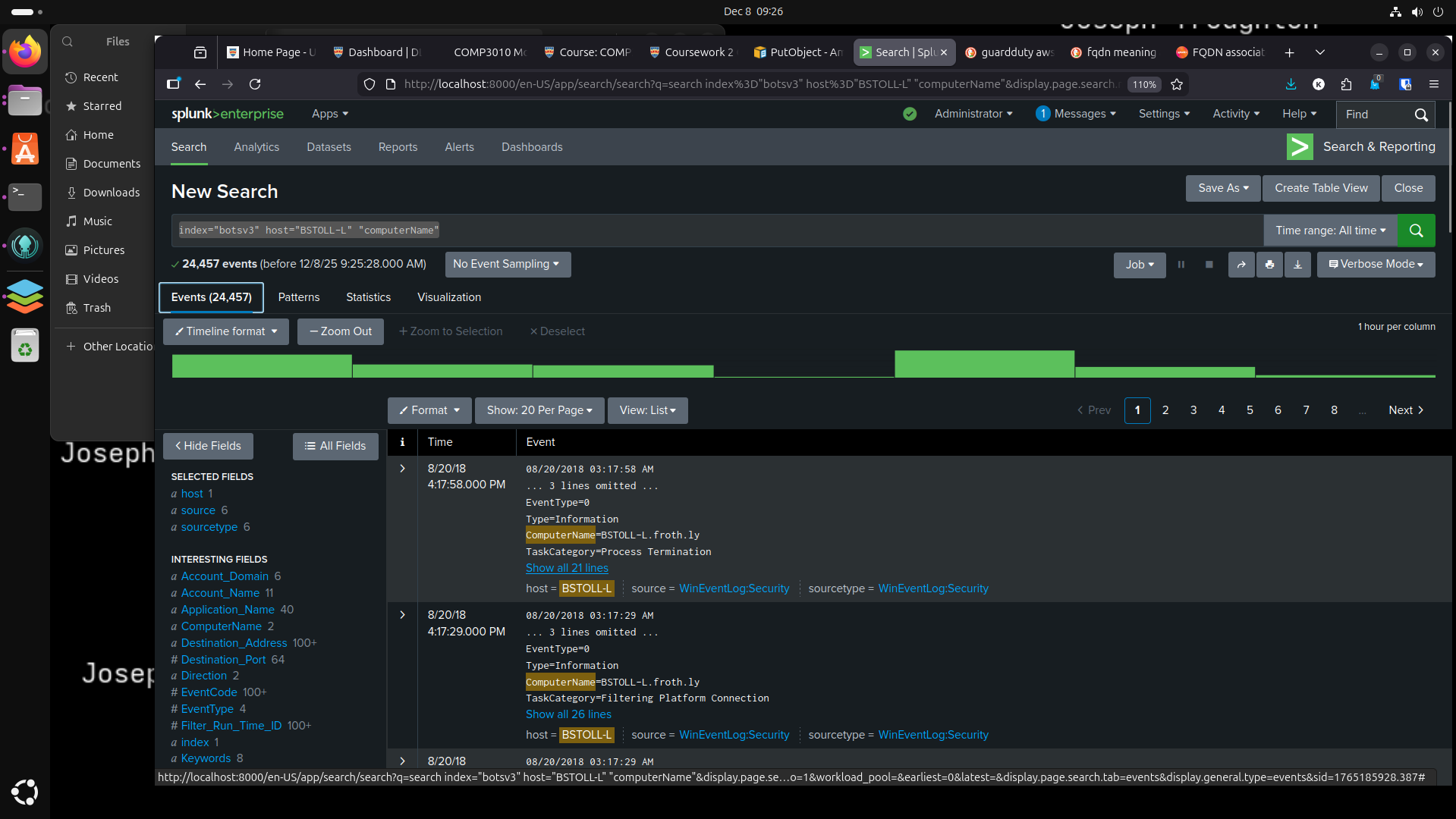
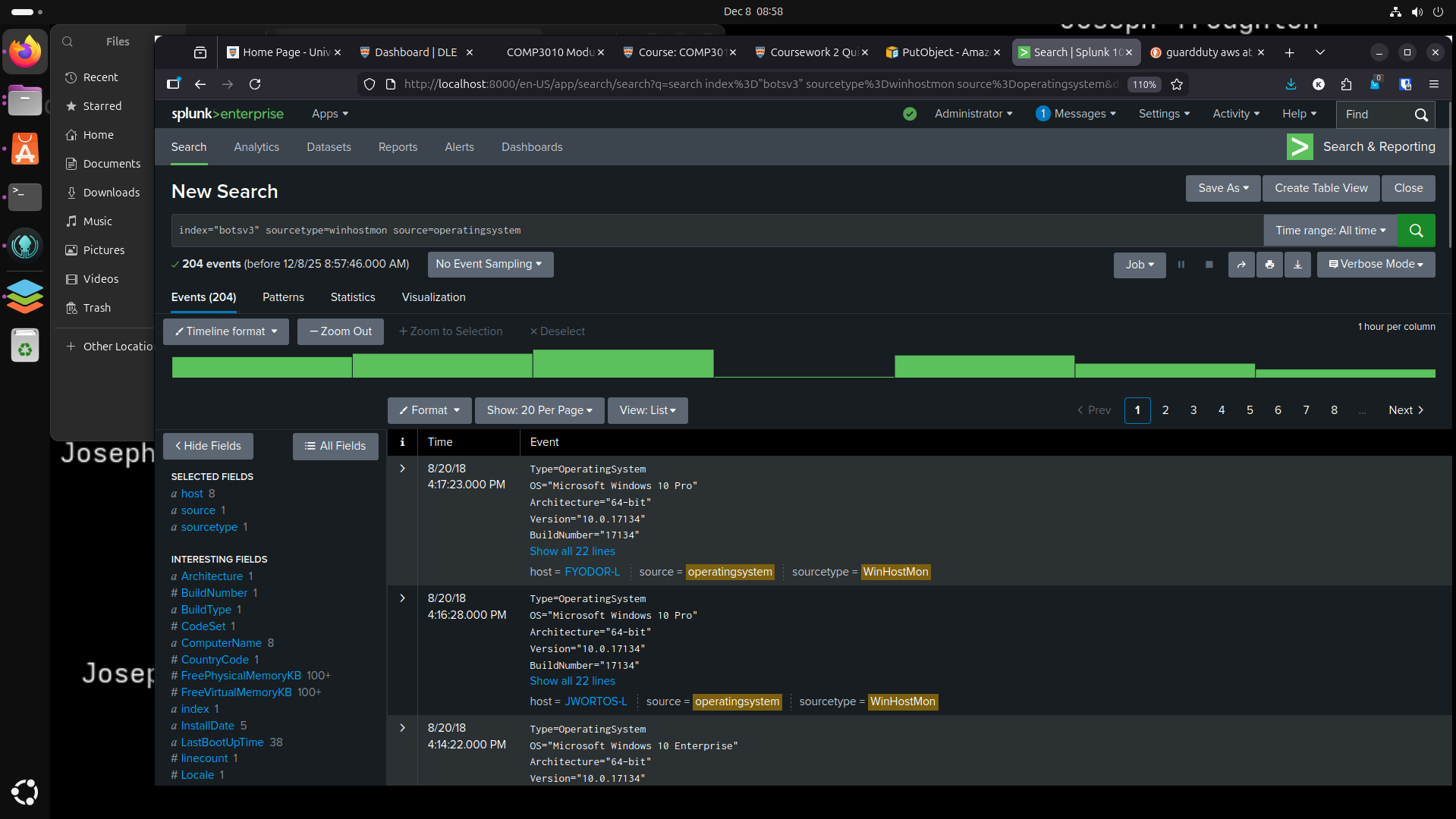
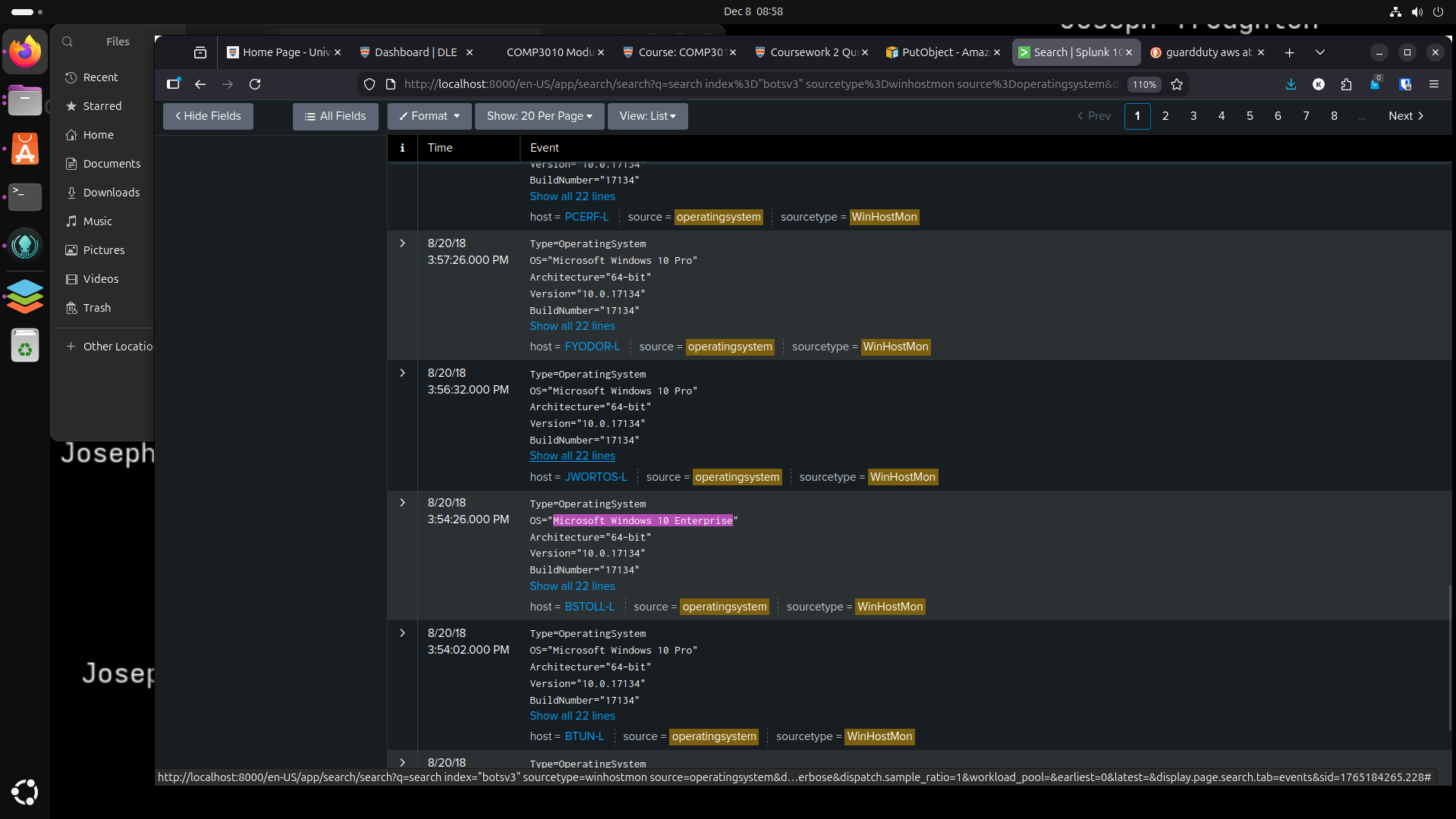
**‘index="botsv3" sourcetype=winhostmon source=operatingsystem’**

Methodology and Justification:Operating system information was compared across hosts. Most systems were running Windows 10 Pro; however, one endpoint was identified running Windows 10 Enterprise. The host was then correlated to its fully qualified domain name (FQDN).

OS deviations may indicate elevated privileges, licensing differences, or misconfigured endpoints and are often flagged during endpoint baseline analysis.

#### Answer:

BSTOLL-L.froth.ly



## Conclusion

This investigation demonstrated how a Security Operations Centre (SOC) analyst can use Splunk to identify and analyse security incidents across cloud and endpoint environments using the BOTSv3 dataset. By examining AWS CloudTrail logs, S3 access logs, endpoint monitoring data, and hardware, the investigation identified key security issues including insecure S3 bucket configurations, identity-based access risks, and endpoint configuration anomalies.

The analysis highlighted how a single cloud misconfiguration such as a publicly accessible S3 bucket can lead to external interaction and increased organisational risk if not detected promptly. The correlation of CloudTrail and S3 access logs confirmed that the misconfiguration resulted in successful object uploads, reinforcing the importance of cross-source log analysis within SOC operations. Additionally, the identification of AWS API activity performed without multi-factor authentication and deviations in endpoint operating system baselines demonstrated common weaknesses that SOC teams must continuously monitor.

Overall, this exercise reinforces the value of centralised log management and structured incident handling methodologies. Implementing proactive detections for cloud misconfiguration, enforcing MFA for API activity, and maintaining endpoint baselines would significantly improve detection capability and reduce response times in real-world SOC environments. The BOTSv3 dataset provides an effective platform for developing practical SOC investigation skills aligned with industry best practices.

# References

[1] Palo Alto Networks, “Security Operations Center (SOC) Roles and Responsibilities,”

https://www.paloaltonetworks.com/cyberpedia/soc-roles-and-responsibilities

[2] Radiant Security, “SOC Analyst Tier 1 vs Tier 2 vs Tier 3,” Feb. 03, 2025,

https://radiantsecurity.ai/learn/soc-tier-1-vs-tier-2-vs-tier-3/

[3] Amazon “AWS Cloud Trail Docs” *Amazon.com*, 2023. https://docs.aws.amazon.com/cloudtrail/

‌[4] “Boss of the SOC (BOTS) Dataset Version 3,” *GitHub*, Mar. 26, 2022. https://github.com/splunk/botsv3

‌