SPLUNK CLOUD SECURITY INVESTIGATION REPORT

COMP3010 Digital Forensics & Incident Response

Maha Almasoudi
10876192

University of Plymouth

1. Abstract

This report documents a forensic investigation performed on the Frothly BOTS v3 dataset using Splunk Enterprise. The aim of the investigation was to identify cloud misconfigurations, examine AWS user activity, analyse Windows host telemetry, and detect evidence of insecure configurations within Frothly's simulated cloud infrastructure. Through a structured methodology involving data ingestion, Splunk search processing, and event correlation, several high-impact security issues were discovered. These included a publicly accessible S3 bucket, API calls executed without MFA, suspicious IAM activity, and the upload of sensitive files to an exposed cloud storage resource. The findings are mapped against MITRE ATT&CK, assessed for their security impact, and presented alongside fully validated Splunk queries used in the analysis.

2. Introduction

The rise of cloud computing has transformed security challenges, particularly in digital forensics and incident response. Misconfigurations remain one of the largest root causes of cloud breaches, with S3 bucket exposure and IAM privilege misuse frequently exploited by attackers. This investigation uses the Frothly BOTS v3 dataset — a realistic set of AWS, operating system, and application logs — to simulate a threat-hunting and forensic workflow.

Splunk Enterprise was selected as the primary SIEM platform due to its advanced indexing, search capabilities, event correlation, and compatibility with the dataset. The assignment aims to demonstrate practical incident response skills, including log collection, query development, cloud forensics, and interpretation of results based on industry best practice.

3. Methodology

A structured cloud forensics methodology was applied:

Step 1 — Environment Preparation

- Linux Mint VM deployed.
- Splunk Enterprise 9.3.0 installed.
- BOTS v3 dataset extracted into $SPLUNK_HOME/etc/apps.
- Splunk restarted and index **botsv3** verified.

Step 2 — Sourcetype Identification

Using:

```
| metadata type=sourcetypes index=botsv3
```

to confirm the presence of:

- aws:cloudtrail
- aws:s3:accesslogs
- WinHostMon
- operatingsystem
- hardware

Step 3 — Searching and Event Extraction

SPL queries were written to:

- Identify IAM users
- Detect S3 access anomalies
- Extract bucket names
- Parse API calls
- Match Windows host telemetry

Step 4 — Interpretation and Correlation

All findings were analysed based on AWS security best practices and MITRE ATT&CK mapping.

4. Environment Setup

The following environment was established:

Splunk Components

- Splunk Enterprise 9.3.0
- Local instance running on Linux Mint
- botsv3 pre-indexed dataset placed into /opt/splunk/etc/apps/

Dataset Verification

Example:

```
index=botsv3 | stats count by sourcetype
```

Sourcetypes Identified

- **aws:cloudtrail** (AWS API logs)
- **aws:s3:accesslogs** (S3 bucket access logs)

- **WinHostMon** (Windows endpoint telemetry)
- **hardware** (CPU and device info)
- **operatingsystem**

This ensured the dataset was ready for deep forensic analysis.

5. Log Analysis & Findings

5.1 IAM Users Accessing AWS Services

Query:

```
index=botsv3 sourcetype="aws:cloudtrail"
| spath userIdentity.userName
| search userIdentity.userName!=""
| dedup userIdentity.userName
| sort userIdentity.userName
| table userIdentity.userName
```

Resulting IAM Users:

bstoll, btun, splunk_access, web_admin

These represent AWS IAM identities making API calls in the environment.

5.2 MFA Not Used in AWS API Activity

AWS CloudTrail logs include a field showing whether MFA was used.

Answer:
userIdentity.sessionContext.attributes.mfaAuthenticated

This field should always be "true" for privileged operations. Missing MFA represents a high-risk condition aligned with MITRE ATT&CK Technique **T1550 – Use of MFA-bypass**.

5.3 Processor Used on Web Servers

Query:

```
index=botsv3 sourcetype=hardware
| table host, _raw
| head 20
```

Answer:
Intel(R) Xeon(R) CPU E5-2676

This confirms the virtualised cloud hardware underlying the web servers.

5.4 Public S3 Bucket Exposure (PutBucketAcl Event)

Query:

```
index=botsv3 sourcetype="aws:cloudtrail"
eventName="PutBucketAcl"
| search "*AllUsers*"
| table _time userIdentity.userName eventID requestParameters
| sort _time
```

Event ID Identified:

ab45689d-69cd-41e7-8705-5350402cf7ac

This corresponds to a write operation setting a public ACL — a critical misconfiguration.

5.5 Identification of Bud's Username

Query:

```
index=botsv3 sourcetype="aws:cloudtrail" eventName=ConsoleLogin
| table _time userIdentity.userName
userIdentity.sessionContext.sessionIssuer.userName _raw
```

Bud's Username:

bstoll

5.6 Name of the Publicly Accessible S3 Bucket

Query:

```
index=botsv3 sourcetype="aws:cloudtrail"
eventName="PutBucketAcl" userIdentity.userName="bstoll"
("*AllUsers*")
| table _time requestParameters.bucketName eventID _raw
```

Bucket Name:

frothlywebcode

This bucket was exposed to unauthenticated public access.

5.7 Text File Uploaded While Bucket Was Public

Query:

```
index=botsv3 sourcetype="aws:s3:accesslogs"
"REST.PUT.OBJECT" " 200 "
| rex field=_raw "(?<filename>[A-Za-z0-9._-]+\.txt)"
| dedup filename
| table filename
```

Uploaded File:
OPEN_BUCKET_PLEASE_FIX.txt

This file was successfully uploaded to the exposed bucket, demonstrating the security risk.

5.8 Endpoint Running Different OS Edition

Query:

```
index=botsv3 sourcetype=WinHostMon source="operatingsystem"
| spath
| stats values(OS) as OS by host
| stats count by OS host
| where count=1
| table host, OS
```

Unique Host:
BSTOLL-L

This device was found to be running a different Windows edition compared to other hosts.

6. MITRE ATT&CK Mapping

Finding	MITRE Technique	Description
IAM user API calls	**T1078 – Valid Accounts**	Use of legitimate credentials
Missing MFA	**T1550.001 – Bypass MFA**	Weak authentication controls
Public S3 bucket	**T1530 – Data from Cloud Storage**	Data exposure via misconfigured storage
File upload to exposed bucket	**T1105 – Ingress Tool Transfer**	Uploading files into accessible cloud environments
Discovery of hardware/OS	**T1082 – System Information Discovery**	Host profiling

7. Security Impact Assessment

Risk: Public S3 Bucket

Severity: **Critical**
Impact: Full unauthorised read/write access
Attackers could:

- Steal sensitive data
- Upload malicious payloads
- Tamper with application assets
- Stage malware for later attacks

Risk: Lack of MFA

Severity: **High**
Increases likelihood of credential abuse.

Risk: IAM Activity by Multiple Users

Severity: **Medium**
Requires monitoring for lateral movement.

Risk: Public File Upload

Severity: **High**
Indicates possible malicious interaction with cloud storage.

Risk: Anomalous OS Version

Severity: **Medium**
May indicate non-standard device or misconfigured endpoint.

---

8. Limitations

- Dataset is static, not live cloud telemetry.
- AWS metadata limited for deeper investigations.
- Missing contextual logs (e.g., IAM role policy details).
- Windows logging limited compared to real-world EDR systems.

---

9. Conclusion

This investigation successfully demonstrated the effective use of Splunk Enterprise for cloud forensic analysis. Significant misconfigurations were identified, including a publicly exposed S3 bucket, missing MFA protections, and unusual endpoint behaviour. Using structured log analysis and event correlation, the investigation mapped findings to recognised MITRE ATT&CK techniques and assessed potential impact. The results highlight the importance of continuous cloud monitoring, strong IAM governance, and secure configuration management.

---

10. Appendix — Full Splunk Queries

IAM Users

```
index=botsv3 sourcetype="aws:cloudtrail"
| spath userIdentity.userName
| search userIdentity.userName!=""
| dedup userIdentity.userName
| sort userIdentity.userName
| table userIdentity.userName
```

MFA Field

```
userIdentity.sessionContext.attributes.mfaAuthenticated
```

Hardware Processor

```
index=botsv3 sourcetype=hardware
```

Public ACL Event

```
index=botsv3 sourcetype="aws:cloudtrail"
eventName="PutBucketAcl"
| search "*AllUsers*"
```

Bud Username

```
index=botsv3 sourcetype="aws:cloudtrail" eventName=ConsoleLogin
```

Public Bucket Name

```
index=botsv3 sourcetype="aws:cloudtrail"
eventName="PutBucketAcl" userIdentity.userName="bstoll"
```

Uploaded File

```
index=botsv3 sourcetype="aws:s3:accesslogs"
"REST.PUT.OBJECT" " 200 "
| rex "(?<filename>[A-Za-z0-9._-]+\.txt)"
```

Unique OS Host

```
index=botsv3 sourcetype=WinHostMon source="operatingsystem"
| spath
| stats values(OS) as OS by host
| stats count by OS host
| where count=1
```