

COMP3010HK Security Operations and Incident Management 2025/2026

Coursework 2: BOTSV3 Incident Analysis and Presentation

Github repo: <https://github.com/aka331/COMP3010HK>

1.0 Introduction.....	2
2.0 SOC Roles & Incident Handling Reflection.....	3
2.1 SOC Roles.....	3
Tier 1 (T1) — Monitoring & Triage.....	3
Tier 2 (T2) — Investigation & Scoping.....	3
Tier 3 (T3) — Threat Hunting & Detection Engineering.....	4
SOC department head.....	4
2.2 Incident Handling Reflection.....	5
Prevention.....	5
Detection.....	5
Response.....	5
Recovery.....	5
3.0 Installation & Data Preparation.....	6
3.1 Environment Setup.....	6
3.2 Dataset Ingestion.....	6
4.0 Investigation.....	8
4.1 Key Findings.....	8
4.2 Splunk Analysis & Evidence.....	9
4.2.1 IAM User Enumeration.....	9
4.2.2 MFA Alerting Logic.....	11
4.2.3 Web Server Hardware Identification.....	13
4.2.4 S3 Public Access Incident.....	14
4.2.5 Suspicious File Upload.....	16
4.2.6 Endpoint Outlier Detection.....	18
5.0 Conclusion.....	21
6.0 Reference.....	22

1.0 Introduction

This report investigates AWS and endpoint-related activity in the BOTSV3 dataset using Splunk. Also, it will answer a set of 200-level guiding questions and use the results to derive practical detection logic and event-handling insights relevant to SOC operations. For the scope that is limited to suspected IAM misuse involving the Frothly environment, potential S3 bucket misconfigurations and file access, and abnormal conditions on critical endpoints. All other BOTSV3 scenarios and non-AWS sources are excluded.

The analysis assumes BOTSV3 logs are complete and trustworthy, and that timestamps follow the dataset's provided format and timezone. No external validation is performed. All work is based solely on BOTSV3 data and Splunk Search Processing Language (SPL) as specified in the course. The investigation is strictly observational and does not make any changes to the Frothly environment. Subsequent sections describe the SOC's role and incident-handling reflections, the Splunk setup and data preparation, the guiding question queries and results, and the overall findings and improvement recommendations.

A Security Operations Center (SOC) is responsible for continuous monitoring, detection, and response to security threats across an organization's systems. In practice, SOC teams focus on three core functions:

- (1) tracking activity across networks, servers, applications, and endpoints
- (2) investigating and responding to suspicious events
- (3) supporting compliance while improving the organization's security posture.

Depending on scale and operational needs, SOC functions may be handled by a single internal team, distributed global teams (GSOCs), or outsourced providers.

2.0 SOC Roles & Incident Handling Reflection

2.1 SOC Roles

In the SOC structure, it has 3 tiers of security analysts and operators. They have different responsibilities and technical skills in the monitoring, analysis, and response to security incidents. In the BOTSV3 scenario, it can be regarded as simulating a complete incident in which an endpoint and cloud resources are compromised, allowing analysts at different levels to carry out their roles across the four phases of prevention, detection, response, and recovery.

Tier 1 (T1) — Monitoring & Triage

For Tier-1, usually referred to as a “front-tier” or “junior” security analyst who responds to 24x7 monitoring of alerts and performs initial triage using Splunk. For example, when the T1 operator validates the alert from Splunk Triggered Alerts, they could determine and review BOTSV3’s basic enrichment, such as AWS CloudTrail, the hostname, source IP, logs, timestamps, affected asset, etc. Once reviewed, they could take action (runbooks or playbooks), such as blocking the intruder’s IP address, escalating the alert/issue to Tier-2 security analysis for further investigation and handling if malware alerts or account hacking attempts. After that, they were required to generate the report for the user or T2 analysts for notification and record.

Tier 2 (T2) — Investigation & Scoping

For Tier-2, usually referred to as “second-line” or “senior” security analysts who have deeper technical knowledge and experience. In a real SOC, they typically understand the organization’s environment and what “normal” looks like, which helps them separate true threats from false positives. They respond to more complex and in-depth security incident investigations. In the BOTSV3 exercise, Tier-2 would build on the Tier-1 handover by running deeper SPL searches, tuning or developing new queries, and correlating multiple data sources (e.g., cloud, authentication, and endpoint logs). Their main tasks are to reconstruct the attack path, validate whether suspicious activity is actually malicious, and determine the full scope of impact—who was affected, what resources were accessed, and how far the activity spread. They also document findings in a structured investigation report, including evidence, timeline, and recommended containment steps.

Tier 3 (T3) — Threat Hunting & Detection Engineering

Tier-3 analysts are the “third-line” experts in a SOC. They handle the most complex, high-impact incidents and provide deep technical capability in areas such as threat hunting, advanced log analysis, incident response support, malware analysis, and digital forensics. In many organizations, Tier-3 also plays a specialist role when evidence must be preserved and explained rigorously—especially in cases that may involve legal review—where a qualified forensic expert may be required to justify methodology, interpret technical findings, and clearly present evidence.

In the BOTSV3 scenario, Tier-3’s focus is less about answering individual alerts and more about long-term improvement. They translate investigation results into durable, reusable capabilities: building and tuning correlation searches, defining high-fidelity detection logic, and developing baselines for “normal” CloudTrail and endpoint behavior. For example, they can extract repeatable IAM misuse patterns, risky S3 configuration indicators, and endpoint anomaly baselines from the dataset, then package these as detection rules and playbook guidance for Tier-1 and Tier-2 to use in daily operations. This turns one investigation into sustained SOC maturity: better alert quality, faster triage, and fewer repeat incidents.

SOC department head

The position of the SOC department head coordinates the overall response, manages communication, and makes risk-based decisions during an incident. In the BOTSV3 context, they confirm severity and likely business impact, assign ownership to the right teams (e.g., cloud or endpoint), and ensure containment actions—such as disabling IAM keys or isolating endpoints—follow organizational policy and minimize operational disruption.

2.2 Incident Handling Reflection

Prevention

In BOTSV3, prevention is shown through strong controls—least-privilege IAM with MFA and key rotation, secure S3 configurations (Block Public Access, strict policies, encryption/logging), and hardened endpoints with patching and EDR. The key lesson is that many incidents start from a vulnerable configuration, so prevention is shared across SOC and cloud/IT teams.

Detection

Detection is the core of the exercise: use Splunk to spot IAM anomalies (rare/privileged actions, unusual IPs), S3 risks (policy changes, suspicious object access), and endpoint abnormalities. Good detection depends on context and correlation across time, identity, asset, and action.

Response

Response turns alerts into action—T1 escalates, T2 confirms scope, and T3 supports with deeper analysis. Typical containment would include disabling compromised IAM keys, locking down S3 permissions, and isolating affected endpoints, while documenting evidence and decisions to avoid either overreaction or delay.

Recovery

Recovery restores normal operations safely and prevents repeat incidents by rotating credentials, validating S3 access, rebaselining endpoints, and converting findings into improved detections and playbooks for faster future response.

3.0 Installation & Data Preparation

3.1 Environment Setup

The SOC lab environment was deployed in a virtualized environment using VMware Workstation / Oracle VirtualBox, hosting a Linux virtual machine (VM), such as Ubuntu / CentOS / Red Hat Enterprise Linux (RHEL). Virtualization provides isolation and reproducibility, which is aligned with SOC lab practices for safely testing datasets and repeated investigations.

Splunk Enterprise (Linux x86_64) was installed under /opt/splunk following common Linux deployment conventions. The installation was performed by downloading the Splunk tarball, extracting it into /opt, and starting Splunk with --accept-license to automate the license acceptance step.

```
wget -O splunk-10.0.1.tgz
"https://download.splunk.com/products/splunk/releases/10.0.1/linux
/splunk-10.0.1-c486717c322b-linux-amd64.tgz"
sudo tar -xzf splunk-10.0.1.tgz -C /opt
sudo /opt/splunk/bin/splunk start --accept-license
```

Bash command 1: Download and install (extract) Splunk

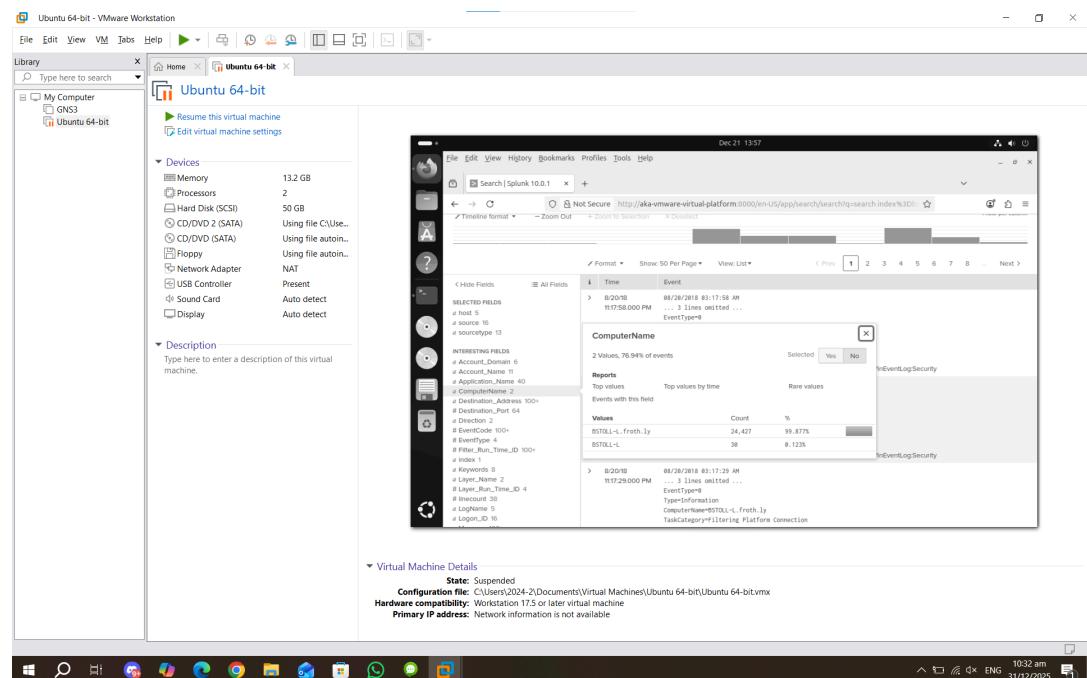


Fig 1. Install the Ubuntu environment in VMware

3.2 Dataset Ingestion

After Splunk was successfully installed, the BOTSV3 dataset was downloaded and ingested by extracting the dataset package into Splunk's apps directory. BOTSV3 is packaged as a Splunk app containing the required data configurations, dashboards, and supporting objects for SOC-style investigations.

Dataset source: BOTSV3 dataset archive (downloaded separately):

https://botsdataset.s3.amazonaws.com/botsv3/botsv3_data_set.tgz

```
cd ~/Downloads  
wget -O botsv3_data_set.tgz  
"https://botsdataset.s3.amazonaws.com/botsv3/botsv3_data_set.tgz"
```

Bash command 2: Download the dataset

```
sudo tar zxvf botsv3_data_set.tgz -C /opt/splunk/etc/apps/
```

Bash command 3: Extract BOTSV3 dataset into Splunk apps path

After extraction, Splunk was restarted to ensure the app and knowledge objects were properly loaded.

```
sudo /opt/splunk/bin/splunk restart
```

Bash command 4: Restart Splunk to load the BOTSV3 app

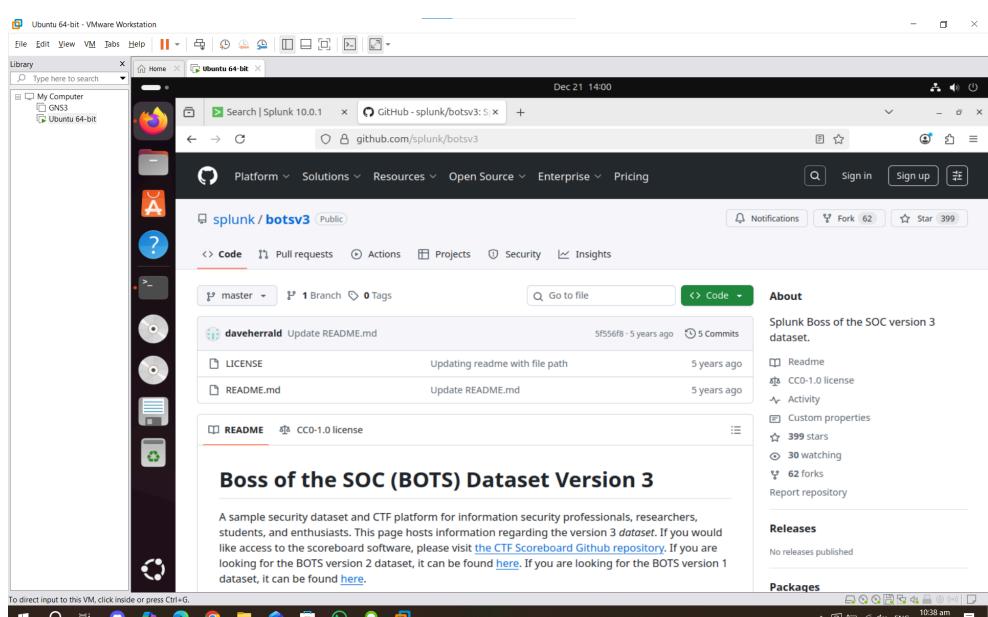


Fig 2. BOTSV3 Github repo

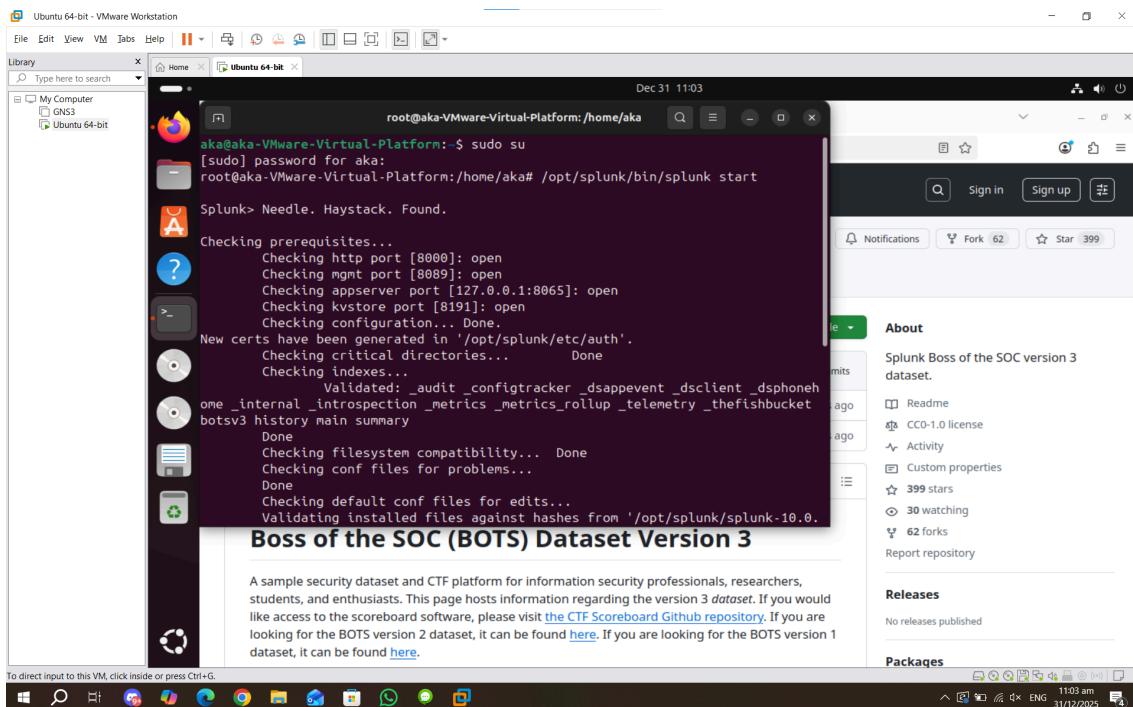


Fig 3. Start the splunk environment in terminal (1)

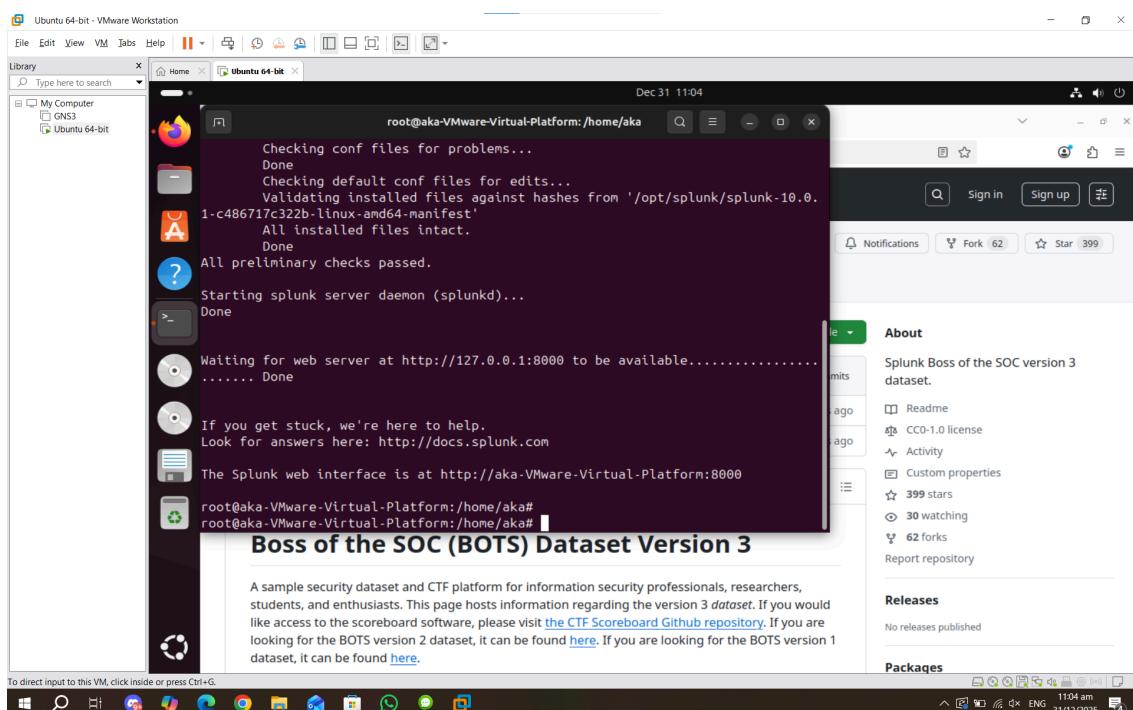


Fig 4. Start the splunk environment in terminal (2)

Finally, Splunk Web can be accessed via:

Inside the VM: <http://127.0.0.1:8000>

From host machine (if needed): http://<VM_IP>:8000 (depends on NAT/bridged setup)

To validate that the BOTSV3 dataset was successfully ingested and searchable, the following SPL query was executed in the Splunk Search interface:

```
index=botsv3 earliest=0
```

SPL 1: Validate BOTSV3 data availability (full time range)

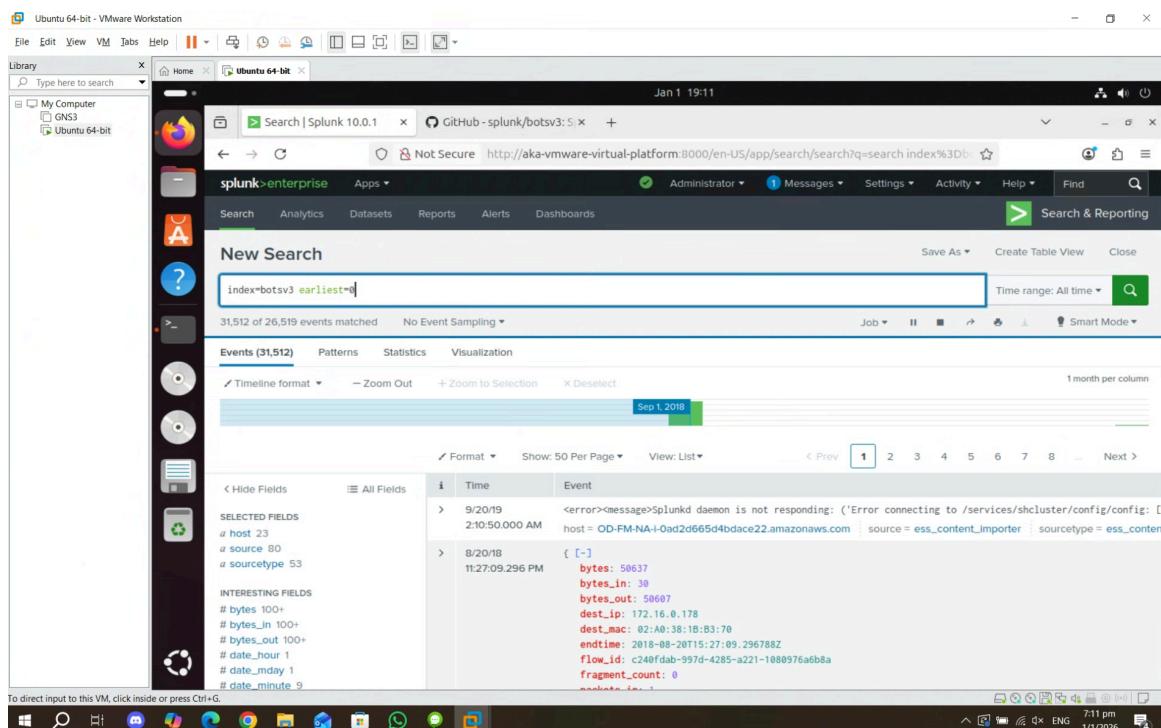


Fig 5. Screenshot of validate BOTSV3 data availability

4.0 Investigation

This section details the investigation of unauthorized IAM activity and S3 bucket compromise within the Frothly AWS environment. The analysis uses Splunk to identify the attacker, the method of compromise, and the extent of data exposure.

4.1 Key Findings

Question	Finding / Answer	SOC Significance
Q1	Active IAM Users: bstoll, btun, splunk_access, web_admin	Identity Baselining: Distinguishes human users from service accounts to tune anomaly detection.
Q2	MFA Alert Field: userIdentity.sessionContext.attributes.mfaAuthenticated	API Security: reliably detects MFA bypass on programmatic API calls, unlike console logs.
Q3	Web Server CPU: E5-2676	Asset Inventory: Identifies hardware-level vulnerabilities.
Q4	Public Access Event ID: ab45689d-69cd-41e7-8705-5350402cf7ac	Incident Timeline: Pinpoints the exact moment of exposure for accurate Time-to-Detect (TTD) metrics.
Q5	Bud's username: bstoll	Attribution: Links the security breach to a specific identity for insider threat investigation.
Q6	S3 bucket: frothlywebcode	Containment: Identifies the specific compromised asset for surgical remediation (ACL rollback).
Q7	text file that was uploaded: OPEN_BUCKET_PLEASE_FIX.txt	Impact Assessment: Confirms unauthorized write access (integrity loss), not just data leakage.
Q8	FQDN: BSTOLL-L.froth.ly	Outlier Detection: Highlights non-standard endpoints that may lack server-grade security controls.

4.2 Splunk Analysis & Evidence

4.2.1 IAM User Enumeration

Objective: Identify all IAM users interacting with AWS services.

Query:

```
index=botsv3 earliest=0 sourcetype=aws:cloudtrail
```

SPL 2. Filter the information from the AWS CloudTrail log

```
index=botsv3 earliest=0 sourcetype=aws:cloudtrail | stats count by userIdentity.userName
```

SPL 3. View statistics about userIdentity.userName in AWS CloudTrail log

The screenshot shows the Splunk Enterprise search interface. The search bar contains the query: `index=botsv3 earliest=0 sourcetype="aws:cloudtrail"`. Below the search bar, it says `6,571 events (12/19/25 2:00:00.000 PM to 12/20/25 2:29:14.000 PM)`. The results table has columns: `i`, `Time`, and `Event`. One event row is expanded, showing detailed fields like `awsRegion: us-west-1`, `eventID: 97c6bfcb-c3cf-437c-8b05-4043635ce306`, etc. The interface includes navigation buttons for pages 1 through 8, and a sidebar with sections for `SELECTED FIELDS` and `INTERESTING FIELDS`.

Fig 1. Capture to filter the information from the AWS CloudTrail log

Select Fields

Select All Within Filter Deselect All Coverage: 1% or more ▾ user X + Extract New Field

i	✓ ▾	Field ▾	# of Values ▾	Event Coverage ▾	Type ▾
>	<input type="checkbox"/>	userIdentity.accessKeyId	>100	90.15%	String
>	<input type="checkbox"/>	userIdentity.sessionContext.attributes.creationDate	>100	32.8%	String
>	<input type="checkbox"/>	userAgent	24	99.54%	String
>	<input type="checkbox"/>	userIdentity.arn	17	94.57%	String
>	<input type="checkbox"/>	userIdentity.principalId	17	94.57%	String
>	<input type="checkbox"/>	responseElements.assumedRoleUser.arn	12	4.52%	String
>	<input type="checkbox"/>	responseElements.assumedRoleUser.assumedRoleId	12	4.52%	String
>	<input type="checkbox"/>	userIdentity.invokedBy	10	24.06%	String
>	<input type="checkbox"/>	userIdentity.sessionContext.sessionIssuer.arn	7	12.01%	String
>	<input type="checkbox"/>	userIdentity.sessionContext.sessionIssuer.principalId	7	12.01%	String
>	<input type="checkbox"/>	userIdentity.sessionContext.sessionIssuer.userName	7	12.01%	String
>	<input type="checkbox"/>	userIdentity.userName	4	82.56%	String
>	<input type="checkbox"/>	userIdentity.type	3	99.62%	String
>	<input type="checkbox"/>	responseElements.userId	2	4.63%	String
>	<input type="checkbox"/>	requestParameters.userData	1	8.86%	String

Fig 2. View the available field in the AWS CloudTrail log



Fig 3. Show the IAM username

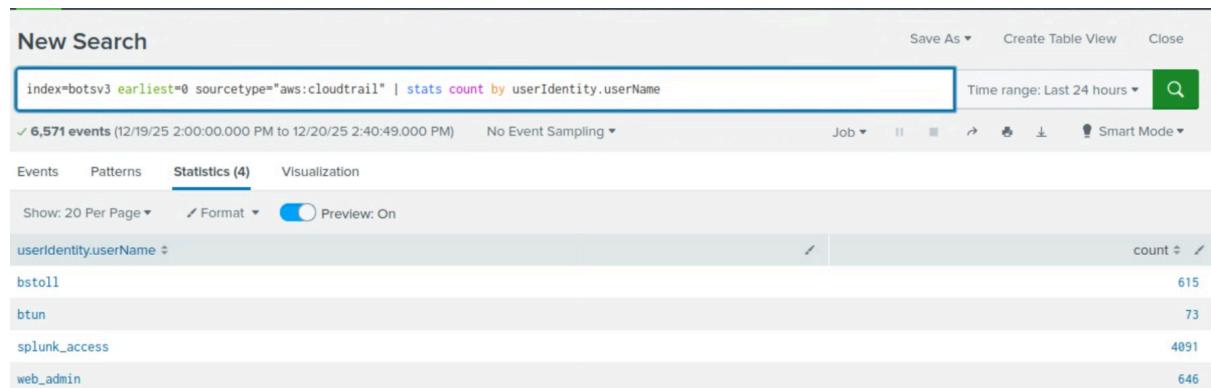


Fig 4. List the IAM username

In the investigation, it reviews the source type based on the hint (`sourcetype:"aws:cloudtrail"`). Using the “user” in the search field to filter information related to the username of IAM. Then, it found the list of IAM users that accessed an AWS service, which is bstoll, btun, splunk_access, web_admin.

Result:

`bstoll, btun, splunk_access, web_admin`

4.2.2 MFA Alerting Logic

Objective: Identify the correct field to detect API activities performed without MFA.

Query:

```
index=botsv3 earliest=0 sourcetype="aws:cloudtrail" | regex "MFA"
```

SPL 4. View statistics that contain the word “MFA” in AWS CloudTrail log

The screenshot shows the Splunk Enterprise interface with the following details:

- Search Bar:** Contains the SPL query: `index=botsv3 earliest=0 sourcetype="aws:cloudtrail" | regex "MFA"`.
- Results Summary:** Shows 129 events from 12/19/25 2:00:00.000 PM to 12/20/25 2:53:06.000 PM.
- Event List:** Displays a single event entry in list view. The event occurred on 8/20/18 at 11:13:30.000 PM. It includes fields such as `awsRegion: us-west-1`, `eventId: d112eb8f-99a1-4fe2-90f8-b21644ea8173`, and `eventName: DescribeConfigRules`.
- Selected Fields:** A sidebar on the left lists selected fields: `@host 1` and `@source 66`.

Fig 5. Capture of statistics contains the word “MFA” in the AWS CloudTrail log

The screenshot shows the 'Select Fields' dialog with the following details:

- Search Bar:** Contains the search term `MFA`.
- Selected Fields:** A list of fields selected for extraction:
 - `additionalEventData.MFAUsed` (Type: String, Coverage: 3.1%, # of Values: 1)
 - `userIdentity.sessionContext.attributes.mfaAuthenticated` (Type: String, Coverage: 0.78%, # of Values: 1)

Fig 6. Search for the MFA in the select field option

Event Actions ▾			
Type	Field	Value	Actions
Selected	<input checked="" type="checkbox"/> host ▾	splunk.froth.ly	▼
	<input checked="" type="checkbox"/> source ▾	s3://cloudtrail-622676721278/AWSLogs/622676721278/CloudTrail_us-east-1_20180725T2335Z_5eML1N4HtL6b4Nwp.json.gz	▼
	<input checked="" type="checkbox"/> sourcetype ▾	aws:cloudtrail	▼
Event	<input type="checkbox"/> additionalEventData.LoginTo ▾	https://console.aws.amazon.com/console/home?state=hashArgs%23&isauthcode=true	▼
	<input type="checkbox"/> additionalEventData.MFAUsed ▾	No	▼
	<input type="checkbox"/> additionalEventData.MobileVersion ▾	No	▼
	<input type="checkbox"/> awsRegion ▾	us-east-1	▼
	<input type="checkbox"/> eventID ▾	6ae589cc-9b68-4a08-a3c2-c29c9cd59b9b	▼
	<input type="checkbox"/> eventName ▾	ConsoleLogin	▼
	<input type="checkbox"/> eventSource ▾	signin.amazonaws.com	▼
	<input type="checkbox"/> eventTime ▾	2018-08-20T15:04:44Z	▼
	<input type="checkbox"/> eventType ▾	AwsConsoleSignIn	▼

Fig 7. The log contain MFA field/attribute

During the research, it displays 2 fields: “additionalEventData.MFAUsed” and “userIdentity.sessionContext.attributes.mfaAuthenticated”. Then, I try to do the research for those 2 values.

The “additionalEventData.MFAUsed” is to determine if the user uses MFA during this specific login attempt. It used to record whether the user successfully provided an MFA token at the moment they signed into the AWS Management Console. The values use “Yes” or “No”.

On the other hand, “userIdentity.sessionContext.attributes.mfaAuthenticated” is to determine if the user performing the command has a valid MFA-verified session. It indicates the state of the session that is making the request. It tells you if the temporary security credentials (session token) being used were originally issued with MFA verification. The values use “True” or “False”.

The question is to find the field that is used to alert that AWS API activity has occurred without MFA. The “**userIdentity.sessionContext.attributes.mfaAuthenticated**” meets the requirement to alert the AWS API activity that has occurred without MFA because the “**additionalEventData.MFAUsed**” is only for login behaviour, excluding API activity.

Result:

`userIdentity.sessionContext.attributes.mfaAuthenticated`

4.2.3 Web Server Hardware Identification

Objective: Determine the specific processor model used on web servers.

Query:

```
index=botsv3 earliest=0 sourcetype="hardware"
```

SPL 5. Filter the information from the hardware log

The screenshot shows a Splunk search interface titled "New Search". The search bar contains the SPL query: `index=botsv3 earliest=0 sourcetype="hardware"`. Below the search bar, it says "3 events (12/19/25 4:00:00.000 PM to 12/20/25 4:40:31.000 PM) No Event Sampling". The search results table has columns: Time, Event, and a sidebar with "Selected Fields" and "Interesting Fields".

Time	Event
8/20/18 10:26:25.000 PM	KEY VALUE Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz CPU_TYPE CPU_CACHE CPU_COUNT HARD_DRIVES Show all 9 lines host = gacrux.i-09cbc261e84259b54 : source = hardware : sourcetype = hardware
8/20/18 10:24:24.000 PM	KEY VALUE Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz CPU_TYPE CPU_CACHE CPU_COUNT HARD_DRIVES Show all 9 lines host = gacrux.i-06fea586f3d3c8ce8 : source = hardware : sourcetype = hardware

Fig 8. Capture of information from the hardware log

Hardware logs reveal the full CPU specification string.

Result:

E5-2676

4.2.4 S3 Public Access Incident

Objective: Investigate the S3 bucket exposure event (Event ID, Actor, and Bucket Name).

Q4

Query:

```
index=botsv3 earliest=0 sourcetype=aws:cloudtrail | regex  
“PutBucketAcl”
```

SPL 6. Filter the information from the AWS CloudTrail log and view the log that contains “PutBucketAcl”

Format ▾			Show: 20 Per Page ▾	View: List ▾
i	Time	Event		
>	8/20/18 9:57:54.000 PM	{ [-] awsRegion: us-west-1 eventID: 9a33d8df-1e16-4d58-b36d-8e80ce68f8a3 eventName: PutBucketAcl eventSource: s3.amazonaws.com eventTime: 2018-08-20T13:57:54Z eventType: AwsApiCall eventVersion: 1.05 recipientAccountId: 622676721278 requestID: 6A18BDBBC85C6E81 requestParameters: { [+] } responseElements: null sourceIPAddress: 107.77.212.175 userAgent: signin.amazonaws.com userIdentity: { [+] } }		
		Show as raw text host = splunk.froth.ly source = s3://cloudtrail-622676721278/AWSLogs/622676721278/CloudTrail/us-west-1/201... sourcetype = aws:cloudtrail		
>	8/20/18 9:01:46.000 PM	{ [-] awsRegion: us-west-1 eventID: ab45689d-69cd-41e7-8705-5350402cf7ac eventName: PutBucketAcl eventSource: s3.amazonaws.com eventTime: 2018-08-20T13:01:46Z eventType: AwsApiCall }		

Fig 9. Capture of the log that contains “PutBucketAcl”

According to the timestamp. There are 2 events named “PutBucketAcl” that appeared at 8/20/18 9:01:46 PM and 9:57:54 PM

Result:

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

Q5

<input type="checkbox"/> userIdentity.accessKeyId ▾		ASIAZB6TMXZ7OA 2RDK5X	▼
<input type="checkbox"/> userIdentity.accountId ▾		622676721278	▼
<input type="checkbox"/> userIdentity.arn ▾		arn:aws:iam::62267 6721278:user/bstoll	▼
<input type="checkbox"/> userIdentity.invokedBy ▾		signin.amazonaw s.com	▼
<input type="checkbox"/> userIdentity.principalId ▾		AIDAJUFKXZ44LV4 EN4MGK	▼
<input type="checkbox"/> userIdentity.sessionContext.attributes.creationDate ▾		2018-08-20T12:19:4 4Z	▼
<input type="checkbox"/> userIdentity.sessionContext.attributes.mfaAuthenticated ▾		false	▼
<input type="checkbox"/> <u>userIdentity.type</u> ▾		IAMUser	▼
<input type="checkbox"/> userIdentity.userName ▾		bstoll	▼
Time +	_time ▾		2018-08-20T21:01:46.000+0 8:00
Default	<input type="checkbox"/> index ▾ <input type="checkbox"/> linecount ▾ <input type="checkbox"/> punct ▾ <input type="checkbox"/> splunk_server ▾	botsv3 1  aka-VMware-Virtua l-Platform	▼

Fig 10. The user who updates the ACL in the S3 bucket

When opening the event details, the `userIdentity.userName` showed the action belongs to bstoll.

Result:

bstoll

Q6

Format ▾		Show: 20 Per Page ▾	View: List ▾
i	Time	Event	
		READ	▼
		FULL_CONTROL	▼
		READ	▼
		WRITE	▼
		requestParameters.AccessControlPolicy.Owner.DisplayName ▾	bstoll ▾
		requestParameters.AccessControlPolicy.Owner.ID ▾	4c018053e740f45 ▾ beb45f68c0f5eff6 347745488ae5401 30432c9fc64fae31 0d
		requestParameters.AccessControlPolicy.xmns ▾	http://s3.amazonaws.com/doc/2006-03-01/ ▾
		requestParameters.acl[] ▾	▼
		requestParameters.bucketName ▾	frothlywebcode ▾
		responseElements ▾	null ▾
		sourceIPAddress ▾	107.77.212.175 ▾
		userAgent ▾	signin.amazonaws.com
		userIdentity.accessKeyId ▾	ASIAZB6TMXZ7OA ▾ 2RDK5X
		userIdentity.accountId ▾	622676721278 ▾
		userIdentity.arn ▾	arn:aws:iam::622676721278:user/bstoll

Fig 11. The S3 bucket name

Result:

frothlywebcode

4.2.5 Suspicious File Upload

Objective: Identify the text file uploaded during the public exposure window.

Query:

```
index=botsv3 earliest=0 frothlywebcode | regex ".txt"
```

SPL 7. Filter the log related to the S3 bucket and containing ".txt" in the regex

Splunk > enterprise Apps

Administrator Messages Settings Activity Help Find Search & Reporting

New Search

Search Analytics Datasets Reports Alerts Dashboards

Events (6) Patterns Statistics Visualization

Time range: Last 24 hours

	Time	Event
>	8/20/18 10:25:21.000 PM	Cloud-init v. 0.7.6 running 'modules:final' at Thu, 26 Jul 2018 01:37:21 +0000. Up 11.07 seconds. ... 228 lines omitted ... Completed 2.9 MiB/2.9 MiB (6.8 MiB/s) with 1 file(s) remaining download: s3://frothlywebcode/frothly_html_memcached.tar.gz to ./frothly_html_memcached.tar.gz Completed 17.1 KiB/17.1 KiB (45.9 KiB/s) with 1 file(s) remaining download: s3://frothlyweb/configs/http_conf.tar.gz to ./http_conf.tar.gz Show all 257 lines host = gacrux.i-09cbc261e84259b54 source = /var/log/cloud-init-output.log sourcetype = cloud-init-output
>	8/20/18 10:23:19.000 PM	Cloud-init v. 0.7.6 running 'modules:final' at Thu, 26 Jul 2018 01:35:19 +0000. Up 13.56 seconds. ... 229 lines omitted ... Completed 2.9 MiB/2.9 MiB (6.3 MiB/s) with 1 file(s) remaining

Fig 12. Capture of the log related to the S3 bucket and containing “.txt” in regex (1)

Splunk > enterprise Apps

Administrator Messages Settings Activity Help Find Search & Reporting

New Search

Search Analytics Datasets Reports Alerts Dashboards

Events (6) Patterns Statistics Visualization

Time range: Last 24 hours

	Time	Event
>	8/20/18 9:33:24.000 PM	host = gacrux.i-0cc93bade2b3cba63 source = /var/log/cloud-init-output.log sourcetype = cloud-init-output ... 229 lines omitted ... Completed 2.9 MiB/2.9 MiB (6.8 MiB/s) with 1 file(s) remaining download: s3://frothlywebcode/frothly_html_memcached.tar.gz to ./frothly_html_memcached.tar.gz Completed 17.1 KiB/17.1 KiB (42.1 KiB/s) with 1 file(s) remaining download: s3://frothlyweb/configs/http_conf.tar.gz to ./http_conf.tar.gz Show all 257 lines host = gacrux.i-0cc93bade2b3cba63 source = /var/log/cloud-init-output.log sourcetype = cloud-init-output
>	8/20/18 9:03:46.000 PM	4c018053e740f45beb45f68c0f5eff6347745488ae540130432c9fc64fae310d frothlywebcode [20/Aug/2018:13:03:46 +0000] [0] 35.182.246.222 - 6CF2A6F4DE3DCE1E8 REST.GET.OBJECT OPEN_BUCKET_PLEASE_FIX.txt "GET /OPEN_BUCKET_PLEASE_FIX.FIX HTTP/1.1" 200 - 377 377 14 13 "-" "aws-cli/1.14.8 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64 botocore/1.8.12" host = splunk.frothly source = s3://frothlyweblogs/s32018-07-26-01-25-30-F2258C3FF62970B6 sourcetype = aws:s3:accesslogs
>	8/20/18 9:02:45.000 PM	4c018053e740f45beb45f68c0f5eff6347745488ae540130432c9fc64fae310d frothlywebcode [20/Aug/2018:13:02:45 +0000] [0] 52.66.146.128 - A01BFC3123EC114C REST.GET.BUCKET - "GET /?prefix=OPEN_BUCKET_PLEASE_FIX.txt&encoding-type=url HTTP/1.1" 200 - 575 - 11 10 "-" "Boto3/1.7.62 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64 Botocore/1.8.12" host = splunk.frothly source = s3://frothlyweblogs/s32018-07-26-01-20-56-19D73C05AA29AED8 sourcetype = aws:s3:accesslogs
>	8/20/18 9:02:44.000 PM	4c018053e740f45beb45f68c0f5eff6347745488ae540130432c9fc64fae310d frothlywebcode [20/Aug/2018:13:02:44 +0000] [0] 52.66.146.128 - DF1BA98D9E2369B4 REST.PUT.OBJECT OPEN_BUCKET_PLEASE_FIX.txt "PUT /OPEN_BUCKET_PLEASE_FIX.FIX HTTP/1.1" 200 - 377 268 9 "-" "Boto3/1.7.62 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64 Botocore/1.8.12" host = splunk.frothly source = s3://frothlyweblogs/s32018-07-26-01-20-56-19D73C05AA29AED8 sourcetype = aws:s3:accesslogs

Fig 13. Capture of the log related to the S3 bucket and containing “.txt” in regex (2)

```
47745488ae540130432c9fc64fde310d frothlywebcode [20/Aug/2018:13:02:11 +0000] "PUT /?prefix=OPEN_BUCKET_PLEASE_FIX.txt&encod 14C REST.GET.BUCKET - "GET /?prefix=OPEN_BUCKET_PLEASE_FIX.txt&encod 10 "-" "Boto3/1.7.62 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64
```

```
//frothlyweblogs/s32018-07-26-01-20-56-19D73C05AA29AED8
```

```
47745488ae540130432c9fc64fde310d frothlywebcode [20/Aug/2018:13:02:11 +0000] "PUT /OPEN_BUCKET_PLEASE_FIX.txt "PUT /OPEN_BUCKET_PLEASE_FIX.txt 9B4 REST.PUT.OBJECT OPEN_BUCKET_PLEASE_FIX.txt "PUT /OPEN_BUCKET_PLEASE_FIX.txt 9 "-" "Boto3/1.7.62 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64
```

```
//frothlyweblogs/s32018-07-26-01-20-56-19D73C05AA29AED8
```

Fig 14. File uploaded into the S3 bucket

Analysis of S3 access logs shows an external PUT request for a .txt file with a 200 OK status.

Result:

OPEN_BUCKET_PLEASE_FIX.txt

4.2.6 Endpoint Outlier Detection

Objective: Identify the endpoint running a different OS edition than the standard fleet.

Query:

```
index=botsv3 earliest=0 sourcetype=winhostmon  
source="operatingsystem" | stats count by host OS
```

SPL 8. Search the operating system in winhostmon

```
index=botsv3 earliest=0 BSTOLL-L
```

SPL 9. Search the data related to BSTOLL-L

Fig 15. View the OS information in “Select Fields”

Fig 16. View statistics for operationsystem in winhostmon source

Fig 17. Windows 10 Enterprise hosted by BSTOLL-L

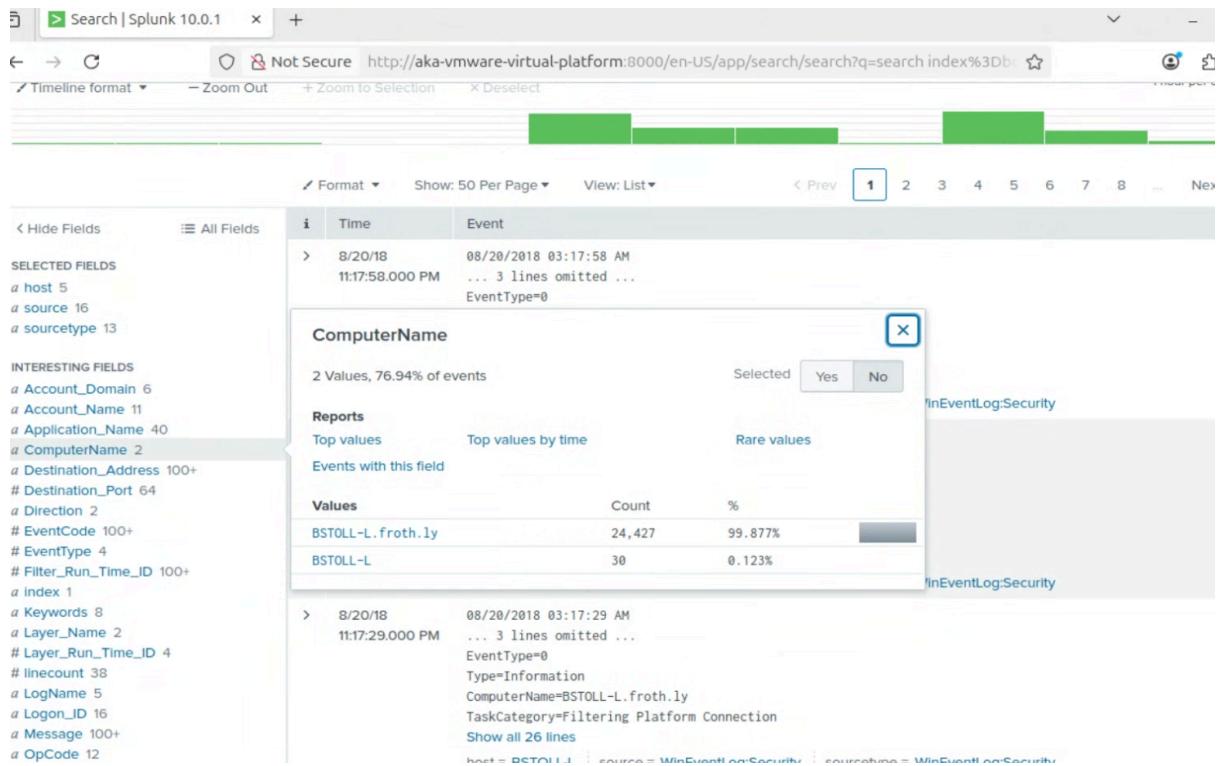


Fig 18. BSTOLL-L.froth.ly (ComputerName) in any data related to BSTOLL-L

While most hosts run Windows Server, one specific host is running Windows 10 Enterprise.

Result:

BSTOLL-L.froth.ly

5.0 Conclusion

The investigation successfully identified a critical data exposure incident caused by a misconfigured S3 ACL on the frothlywebcode bucket. The breach, initiated by user bstoll from the endpoint BSTOLL-L.froth.ly, allowed unauthorized public access and resulted in the upload of OPEN_BUCKET_PLEASE_FIX.txt.

Key Recommendations:

1. Prevention: Enable "Block Public Access" at the AWS Account level and enforce mandatory MFA for all privileged API calls to prevent accidental exposure.
2. Detection: Implement real-time Splunk alerts for high-risk events like PutBucketAcl targeting "AllUsers" to minimize Time-to-Detect (TTD).
3. Response: Deploy SOAR playbooks to automatically revert unauthorized public bucket policies and isolate non-standard endpoints (like BSTOLL-L) that deviate from the security baseline.

6.0 Reference

1. Cloudflare. (n.d.). What is a security operations center (SOC)?
<https://www.cloudflare.com/zh-tw/learning/security/glossary/what-is-a-security-operations-center-soc/>
2. Tutorialspoint. (n.d.). Splunk - Search Processing Language. Retrieved December 24, 2025, from https://www.tutorialspoint.com/splunk/splunk_search_language.htm

Student Declaration of AI Tool use in this Assessment

Please indicate your level of usage of generative AI for this assessment - please tick the appropriate category(s).

If the “Assisted Work” or “Partnered Work” category is selected, please expand on the usage and in which elements of the assignment the usage refers to.

Solo Work	S1 - Generative AI tools have not been used for this assessment.	<input type="checkbox"/>
Assisted Work	A1 – Idea Generation and Problem Exploration Used to generate project ideas, explore different approaches to solving a problem, or suggest features for software or systems. Students must critically assess AI-generated suggestions and ensure their own intellectual contributions are central.	<input type="checkbox"/>
	A2 - Planning & Structuring Projects AI may help outline the structure of reports, documentation and projects. The final structure and implementation must be the student’s own work.	<input type="checkbox"/>
	A3 – Code Architecture AI tools maybe used to help outline code architecture (e.g. suggesting class hierarchies or module breakdowns). The final code structure must be the student’s own work.	<input type="checkbox"/>
	A4 – Research Assistance Used to locate and summarise relevant articles, academic papers, technical documentation, or online resources (e.g. Stack Overflow, GitHub discussions). The interpretation and integration of research into the assignment remain the student’s responsibility.	<input type="checkbox"/>
	A5 - Language Refinement Used to check grammar, refine language, improve sentence structure in documentation not code. AI should be used only to provide suggestions for improvement. Students must ensure that the documentation accurately reflects the code and is technically correct.	<input type="checkbox"/>

<p>A6 – Code Review</p> <p>AI tools can be used to check comments within the code and to suggest improvements to code readability, structure or syntax. AI should be used only to provide suggestions for improvement. Students must ensure that the code accurately reflects their knowledge and is technically correct.</p>	<input type="checkbox"/>
<p>A7 - Code Generation for Learning Purposes</p> <p>Used to generate example code snippets to understand syntax, explore alternative implementations, or learn new programming paradigms. Students must not submit AI-generated code as their own and must be able to explain how it works.</p>	<input type="checkbox"/>
<p>A8 - Technical Guidance & Debugging Support</p> <p>AI tools can be used to explain algorithms, programming concepts, or debugging strategies. Students may also help interpret error messages or suggest possible fixes. However, students must write, test, and debug their own code independently and understand all solutions submitted.</p>	<input type="checkbox"/>
<p>A9 - Testing and Validation Support</p> <p>AI may assist in generating test cases, validating outputs, or suggesting edge cases for software testing. Students are responsible for designing comprehensive test plans and interpreting test results.</p>	<input type="checkbox"/>
<p>A10 - Data Analysis and Visualization Guidance</p> <p>AI tools can help suggest ways to analyse datasets or visualize results (e.g. recommending chart types or statistical methods). Students must perform the analysis themselves and understand the implications of the results.</p>	<input type="checkbox"/>
<p>A11 - Other uses not listed above</p> <p>Please specify:</p>	<input type="checkbox"/>

Partnered Work	<p>P1 - Generative AI tool usage has been used integrally for this assessment</p> <p>Students can adopt approaches that are compliant with instructions in the assessment brief.</p> <p>Please Specify:</p> <ul style="list-style-type: none"> - The analysis of the botsv3 log, and guide the splunk's query - Report drafting and improvement - Verify the output - Understand the SOC environment 	<input checked="" type="checkbox"/>
-----------------------	---	-------------------------------------

<p>Please provide details of AI usage and which elements of the coursework this relates to:</p> <p>The analysis of botsv3 log, and guide the splunk's query, report improvement and the soc report structure</p>

<p>I understand that the ownership and responsibility for the academic integrity of this submitted assessment falls with me, the student.</p>	<input checked="" type="checkbox"/>
<p>I confirm that all details provided above are an accurate description of how AI was used for this assessment.</p>	<input checked="" type="checkbox"/>