

## **Introduction**

This report will cover the analysis of the dataset BOTSv3. Analysis was performed by the tool "Splunk" which allows filtering and formatting of data in large datasets, using this tool we can audit and analyse the operations performed captured in the dataset, who they were performed by, when they were performed and information about the technical specifics behind the users and tools used.

Each Question will have supporting evidence in the form of screenshots taken directly from the Splunk analysis. In addition to the general screenshot that shows the answer to the question, there will also be a link to a folder on this Github with more supporting evidence, this could be in the form of step by step walkthroughs of the analysis process for the question or additional relevant evidence.

## **SOC Roles & Incident Handling Reflection**

There are three Tiers to SOC analysts, below I will highlight their roles and how they relate to the BOTSv3 exercise.

### **SOC Analyst Tiers Overview [1]**

<b>Tier</b>	<b>Level</b>	<b>Responsibility</b>
<b>1</b>	<b>Junior Level</b>	<b>Largely responsible for vulnerability scanning.</b>
<b>1</b>	<b>Junior Level</b>	<b>Will determine the severity of threats and escalate threats that require further attention.</b>
<b>1</b>	<b>Junior Level</b>	<b>In charge of managing the security tools used, for example, Splunk or Wireshark.</b>
<b>1</b>	<b>Junior Level</b>	<b>Does not proactively search for threats, only determine the severity of threats caught IDS software.</b>
<b>2</b>	<b>Mid Level</b>	<b>Handle the escalated more complex threats that have been escalated by Tier One</b>
<b>2</b>	<b>Mid Level</b>	<b>Use further tools to build up a more comprehensive picture. Where tier one will find unusual activity, tier two will understand what happened and how can they prevent it.</b>
<b>3</b>	<b>Senior Level</b>	<b>Perform active analysis to spot threats that aren't picket up automatically.</b>
<b>3</b>	<b>Senior Level</b>	<b>Perform Penetration tests, which consists of testing the security of a system by acting as an attacker to point out any vulnerabilities</b>

## SOC Incident Handling Phases

Phase	Description
Prevention	<p>Using Intrusion detection systems to detect unusual activity and applying security practices, like encryption.</p> <p>The use of automated SIEM's that are leveraging AI to detect incoming threats in real time keeps the workload minimal so technical analysts can focus on more complex threats.</p> <p>In the case of the BOTSv3 exercise, ensuring data is protected, like in the case of the buckets by using an access control list with correct privacy would help prevent more serious data leaks.</p>
Detection	<p>Detection can be performed using tools like Wireshark to capture network traffic, and more relevant to this exercise, Splunk can be used to index and filter large amounts of data to detect vulnerabilities, misconfigurations and other security concerns.</p> <p>In the case of BOTSv3, using search filters a data storage bucket was found to have been set to public which is a huge security concern.</p>
Response	<p>The incident response process consists of creating and incident response plan which details the roles and processes that would be applied if a specific incident were to take place, in the context of this exercise, the response to an analyst finding the AWS buckets ACL public was to use that to upload the text file "OPEN_BUCKET_PLEASE_FIX.txt", however, in addition to this, a good response plan would have steps on who to report this to, and any steps that should be taken by the analyst that discovered the issue, if they are qualified to do so. The consequences for this error could have been far worse so it's important to have clear steps to ensure quick and comprehensive solutions.</p>
Recovery	<p>As part of an incident response plan, redundancy systems and other risk mitigation processes should be outlined and implemented. After an incident has been discovered, following the incident recovery plan to implement the solution will ensure a fix is correctly implemented and does not cause any unforeseen issues. In this case, the recovery process consisted of alerting the bucket owner that of the mistake using the text file.</p>

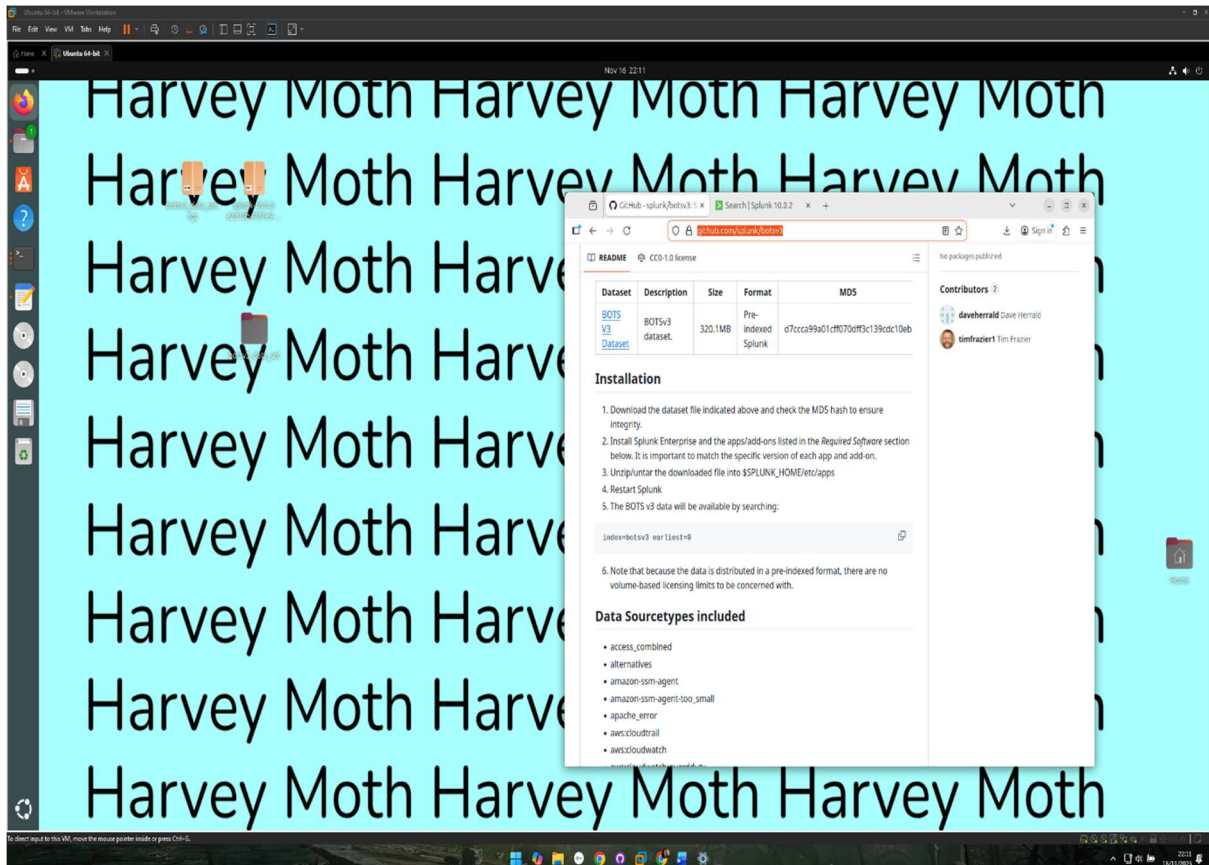
## Installation & Data Preparation

Installation of Splunk was performed on a VMware virtual machine running Ubuntu. Splunk Enterprise was downloaded from the official site

"[https://www.splunk.com/en\\_us/download.html](https://www.splunk.com/en_us/download.html)".

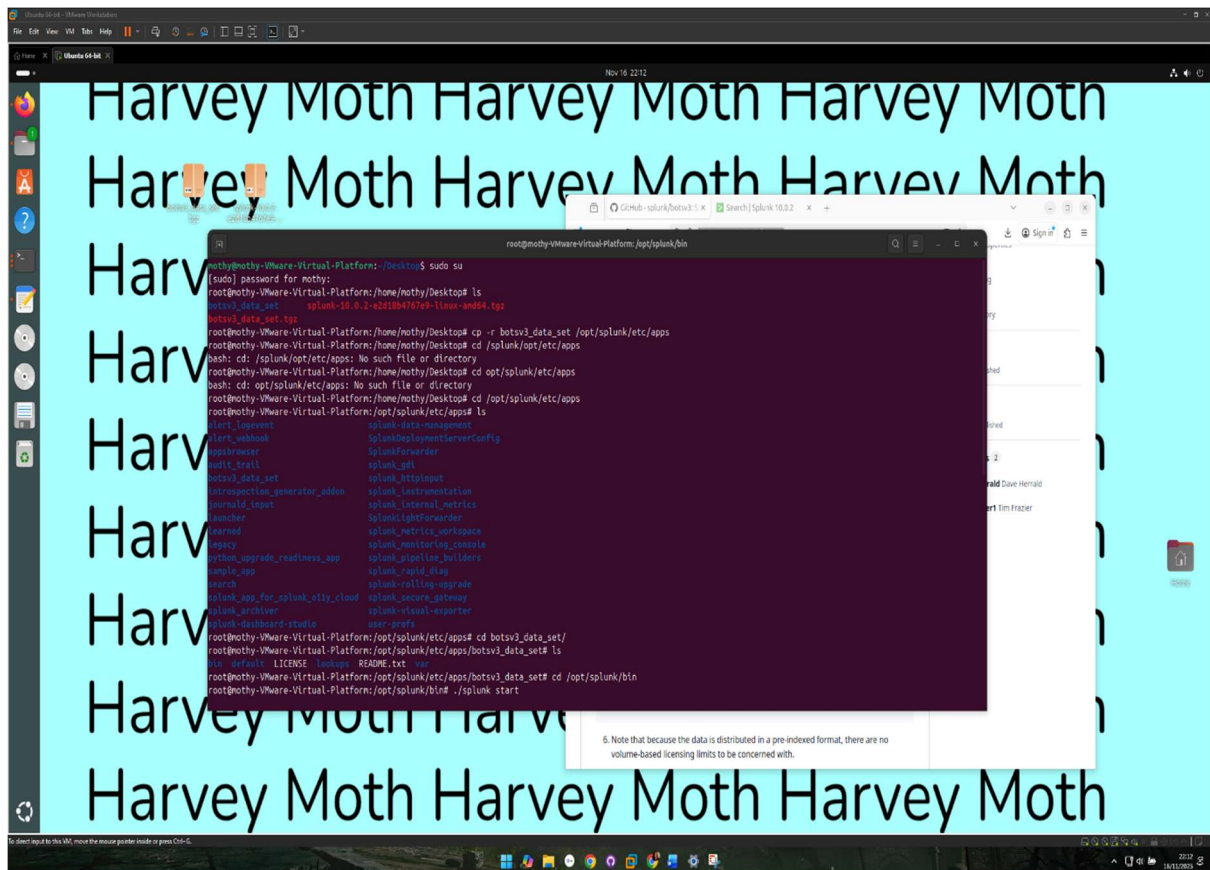
**Splunk Installation Installation Walkthrough Screenshots:** <https://github.com/Harvey-Moth/Comp3010/tree/615202f18d654ddf5638b6f17b9aa27cb7d89d15/Walkthrough%20Screenshots/Splunk%20installation>

### 1. Downloading the dataset from the Github repository



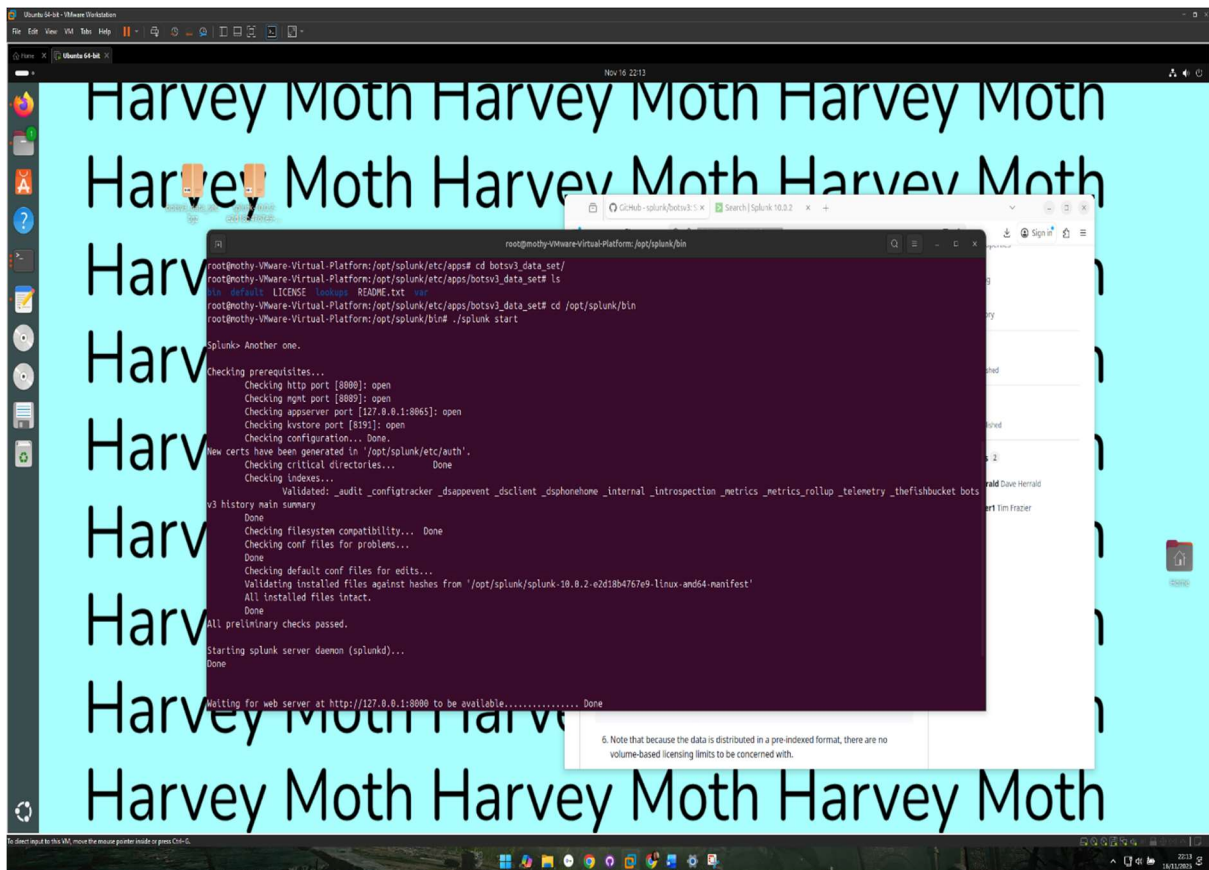
The BOTSv3 dataset is available from a public Github repository. Downloading it is as simple as downloading the archive to a Linux machine or virtual machine.

## 2. Moving the dataset to the correct folder so Splunk can access it



Moving the dataset to the correct folder is important. We must ensure the data is stored in the same place as the Splunk files for ease of indexing later on in the process.

### 3. Running Splunk from the dataset directory



```
root@mothy-Virtual-Platform:/opt/splunk/etc/apps# cd botsv3_data_set/
root@mothy-Virtual-Platform:/opt/splunk/etc/apps/botsv3_data_set# ls
bin  default  LICENSE  lookups  README.txt  var
root@mothy-Virtual-Platform:/opt/splunk/etc/apps/botsv3_data_set# cd /opt/splunk/bin
root@mothy-Virtual-Platform:/opt/splunk/bin# ./splunk start

Splunk> Another one.

Checking prerequisites...
Checking http port [8080]: open
Checking https port [8089]: open
Checking appserver port [127.0.0.1:8065]: open
Checking kvstore port [8191]: open
Checking configuration... Done.
New certs have been generated in '/opt/splunk/etc/auth'.
Checking critical directories... Done
Checking indexes...
v3 history main summary
Done
Checking filesystem compatibility... Done
Checking conf files for problems... Done
Checking default conf files for edits... Done
Validating installed files against hashes from '/opt/splunk/splunk-10.0.2-e2d18b4767e9-linux-and64-manifest'
All installed files intact.
Done
All preliminary checks passed.
Starting splunk server daemon (splunkd)...
Done

Waiting for web server at http://127.0.0.1:8080 to be available..... Done

6. Note that because the data is distributed in a pre-indexed format, there are no
volume-based licensing limits to be concerned with.
```

Once the dataset is in the correct place, we can start Splunk with the `./Splunk start` command. It is important we use the superuser "Sudo" prefix in order to prevent any protected storage locations from interfering with the software, "Sudo" allows us to run it as a administrator.

#### 4. Opening the dataset in Splunk and indexing it

The screenshot displays a Splunk search interface. The search bar contains the query `index="botsv3"`. The results show 2,083,098 events. The interface includes a sidebar with navigation options like Search, Analytics, Dashboards, Reports, Alerts, and Dashboards. The main panel shows a list of events with columns for Time and Event. The event list is currently empty, but the search bar and sidebar are visible.

Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth  
Harvey Moth Harvey Moth Harvey Moth

By using 'Index = "BOTSv3"' we can load the dataset into Splunk and begin applying filters to manage the data.

## Guided Questions

Question 1 - Identify all users that accessed the AWS service using Frothyly's AWS Environment

### Walkthrough Screenshots:

<https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecee3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%201>

Methodology:

- index="botsv3" sourcetype = "aws=cloudtrail"
- used the field select to filter for useridentity.userName
- used the filter to show all the values.

Answer: splunk\_access, web\_admin, bstoll, btun

By finding out all the users that accessed the AWS environment, successfully or unsuccessfully, we can begin to piece together if this is a simple mistake made by a team member or something more malicious, caused by an attacker.

### Evidence:

The screenshot displays a VMware Workstation environment with an Ubuntu 64-bit VM. The VM is running a Splunk installation, and the Splunk web interface is visible in the background. The foreground shows a terminal window with the following output:

```
Done
Checking filesystem compatibility... Done
Checking conf files for problems... Done
Checking default conf files for edits... Done
Validating installed files against hashes from '/opt/splunk/splunk-10.0.0-2-0218b47679-linux-and64-manifest'
All installed files intact.
All preliminary checks passed.
Starting splunk server daemon (splunkd)... Done
Waiting for web server at http://127.0.0.1:8000 to be available..... Done
If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com
The Splunk web interface is at http://mothy-vmware-virtual-platform:8000
mothy@mothy-vmware-virtual-platform: /opt/splunk/bin $
```

The Splunk web interface shows a search for the query: `index="botsv3" sourcetype="aws=cloudtrail" useridentity.userName`. The search results show 5,425 events. The field select dropdown is open, showing the selected fields: `host`, `source`, `sourceip`, `sourcetype`, and `useridentity.userName`. The field select menu is also open, showing the selected field: `useridentity.userName`. The field select menu is also open, showing the selected field: `useridentity.userName`.

Values	Count	%
splunk_access	4,881	75.41%
web_admin	846	11.96%
bstoll	815	11.13%
btun	73	1.34%



### Question 3 - What is the processor number used on the web servers?

Walkthrough Screenshots:

[https://github.com/Harvey-](https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecce3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%203)

[Moth/Comp3010/tree/cb6aa285ecce3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%203](https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecce3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%203)

Methodology: Used the hardware search filter

Answer: Intel(R) Xeon(R) CPU E5-2676 @2.40Ghz

Understanding details about the hardware for the server helps us determine if systems are running as usual, as slow performance can be a sign of an issue. Knowing the hardware being used by the server, especially one you do not have physical access too like a cloud server, helps stay on top of hardware vulnerabilities.

### Evidence:

The screenshot displays a VMware Workstation environment. On the left, a terminal window shows the installation of Splunk on a Linux system. The terminal output includes messages such as 'Checking filesystem compatibility...', 'Checking conf files for problems...', 'Checking default conf files for edits...', 'Validating installed files against hashes from /opt/splunk/splunk-18.0...', 'All installed files intact.', 'All preliminary checks passed.', 'Starting splunk server daemon (splunkd)...', 'Waiting for web server at http://127.0.0.1:8080 to be available..... Done', and 'The Splunk web interface is at http://mothy-Virtual-Platform:8080'. On the right, a web browser window shows the Splunk 'New Search' page. The search query is 'index="hardware" source=type="Hardware"'. The search results are displayed in a table with columns for Time, Event, and Value. The table shows three events, all of which are related to the hardware search filter. The first event is at 8:20:16, the second at 8:20:16, and the third at 8:20:16. The table also shows the source and sourcetype for each event.

Time	Event	Value
8:20:16	KEY	index="hardware" source=type="Hardware"
8:20:16	CPU_TYPE	Intel(R) Xeon(R) CPU E5-2676 @ 2.40GHz
8:20:16	CPU_CODE	38728 KB
8:20:16	CPU_COUNT	2
8:20:16	HARD_DISK1	virtio 1 (0)
8:20:16	Host	genova-096a0b6d84258054
8:20:16	source	hardware
8:20:16	sourcetype	hardware

**Question 4 - Bud accidentally makes an S3 bucket publicly accessible. What is the event ID of the API call that enabled public access?**

Walkthrough Screenshots:

<https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecee3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%204-6/4>

Methodology:

- Added the eventID filter
- Added the bucketname parameter to gather all bucket api calls
- Added the eventname parameter to search for the PutBucketAcl name as that is always the type of api call performed.

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

Using the event ID, we can flag the specific API call that caused the issue, this specific event ID is the API call "PutBucketAcl" that made the bucket public and created the vulnerability.

**Evidence:**

The screenshot shows a Splunk search interface with a search bar containing the query: `awsRegion: us-west-1 eventID: ab45689d-69cd-41e7-8705-5350402cf7ac`. The search results table displays the following event:

Time	Event
2:01:46.000 PM	<pre>awsRegion: us-west-1 eventID: ab45689d-69cd-41e7-8705-5350402cf7ac eventName: PutBucketAcl eventSource: s3.amazonaws.com eventTime: 2018-08-20T13:01:46Z eventType: AwsApiCall eventVersion: 1.05 recipientAccountId: 622676721278 requestID: 487488003569438 requestParameters: { [*] } responseElements: null sourceIPAddress: 107.77.212.175 userAgent: sign-in.amazonaws.com userIdentity: { [*] } }</pre>

The event details show that the API call was `PutBucketAcl` made by `sign-in.amazonaws.com` at `2018-08-20T13:01:46Z`. The event ID `ab45689d-69cd-41e7-8705-5350402cf7ac` is highlighted in the original image.

Answer: frothlywebcode

Knowing the name of the bucket in question is essential to moving forward with a recovery plan. We can determine the extent of possible damage, how important the data inside the bucket is as if it's sensitive data it could cause GDPR breaches and timeframe for how long the vulnerability was unnoticed and unfixed.

## Evidence:

The screenshot shows a VMware Workstation interface with an Ubuntu 64-bit VM. The terminal window displays the installation of Splunk, including commands like `sudo apt-get install splunk` and `splunkd`. A web browser window shows the Splunk web interface with a search for 'rothlywebcode' in the bucketName field. The search results show a single event with details like recipientAccountId, requestId, and source.

Terminal Output:

```
Done
Checking filesystem compatibility... Done
Checking conf files for problems... Done
Checking default conf files for edits... Done
Validating installed files against hashes from '/opt/splunk/etc/default/installer-hash-check-manifest'
All installed files intact.
All preliminary checks passed.
Starting splunk server daemon (splunkd)...
Waiting for web server at http://127.0.0.1:8000 to be available...
If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com
The Splunk web interface is at http://mothy-vmware-virtual-platform:8000/
mothy@mothy-vmware-virtual-platform:~$
```

Search Results:

Time	Event
2023-12-28T13:13:13.131Z	<pre>{   "recipientAccountId": "622676721278",   "requestId": "4874880003569438",   "requestParameters": {     "bucketName": "rothlywebcode"   },   "responseElements": null,   "sourceIPAddress": "107.77.212.175",   "userAgent": "signin.amazonaws.com",   "userIdentity": {     "type": "IAMUser",     "principalId": "AIDAIQ4TQW462676721278",     "arn": "arn:aws:iam::622676721278:user/AIDAIQ4TQW462676721278",     "sessionContext": {       "attributes": {         "mfaAuthenticated": false       }     }   } }</pre>

## Question 7 - What is the name of the text file that was successfully uploaded into the S3 bucket while it was publicly accessible?

Walkthrough Screenshots:

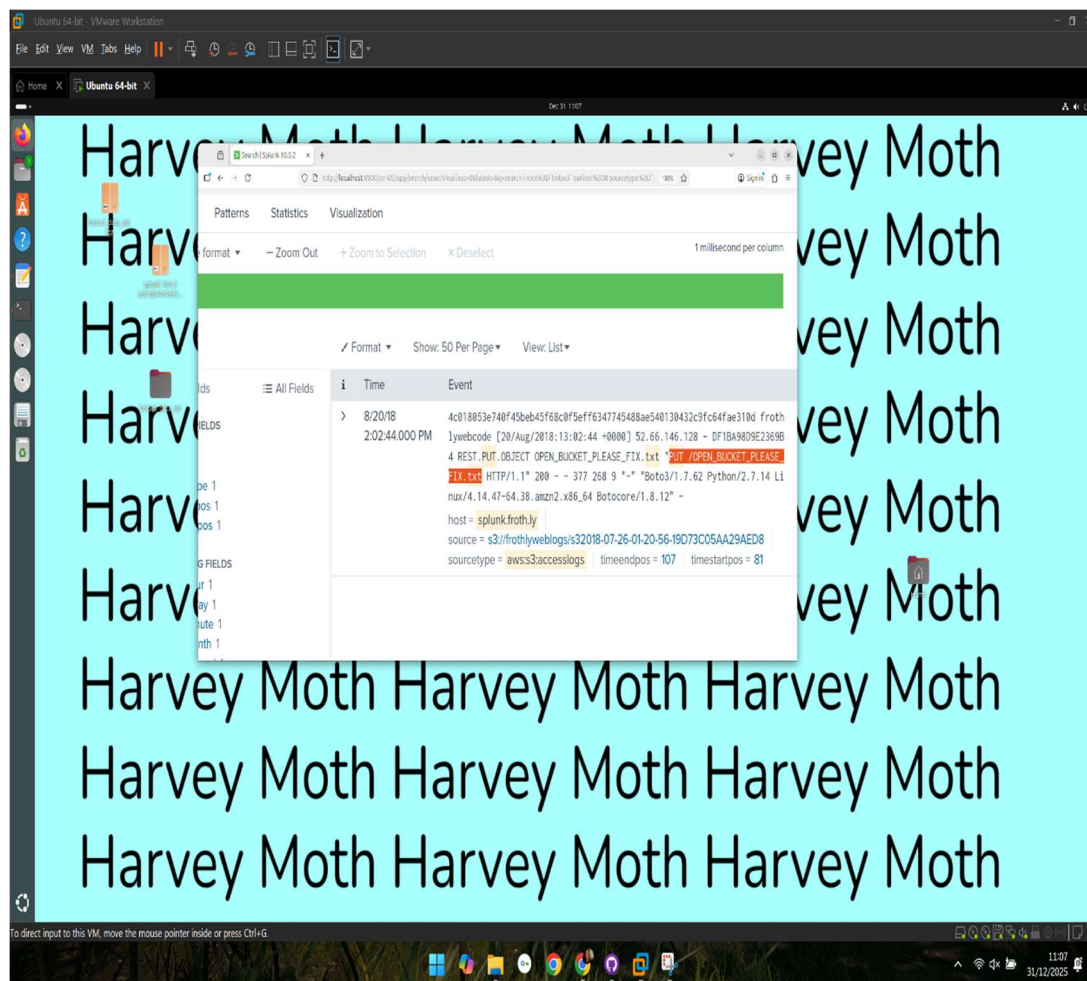
<https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecee3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%207>

Methodology: Using the sourcetype "aws:s3:accesslogs" and the known hostname "splunk.froth.ly" I searched for anything using "Put" as that would be used in a PutObject operation. As we know the uploaded file is a text file, I searched for listings containing "docx" firstly but no results appeared, I then tried "txt" and got a match, to confirm, I made a note of the timestamp and compared it to the time we know the bucket was open for. The bucket was made public at 2:01:46.000 PM and the file was uploaded at 2:02:44.000 PM meaning after the bucket was made public.

Answer: OPEN\_BUCKET\_PLEASE\_FIX.txt

Being able to see the names of files within the bucket means we were able to see the API calls made to the open bucket, and even the contents of the "PUT" call which shows the lack of security for this specific bucket and on a more serious note, would allow us to see if anything malicious was uploaded during the vulnerability period.

### Evidence:



Additional timeline evidence:

<https://github.com/Harvey-Moth/Comp3010/blob/eb221c0a2b4c732cc0fc1ff56804d47a6f902d26/Walkthrough%20Screenshots/Questions/Question%207/Timeline%20evidence.png>

**Question 8 - What is the FQDN of the endpoint that is running a different Windows operating system edition than the others?**

Walkthrough Screenshots:

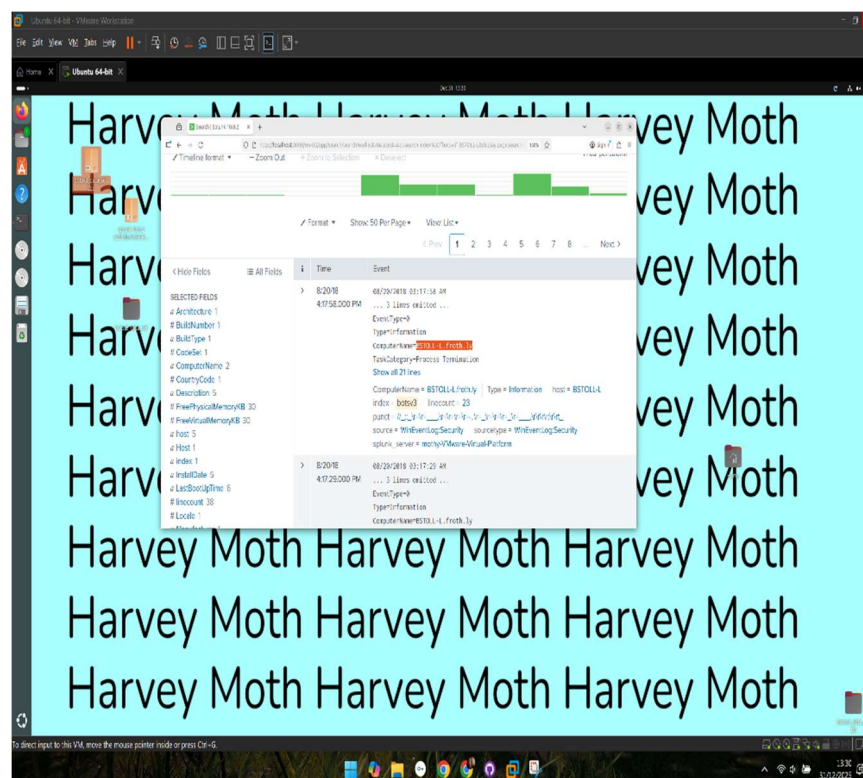
<https://github.com/Harvey-Moth/Comp3010/tree/cb6aa285ecce3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%208>

Methodology: Firstly, I searched driver providers to see which ones were Microsoft drivers but that did not give me much. After attempting that method, I found the "OperatingSystem" filter and when I applied it, I found there were 2 operating systems used by the hosts, Windows 10 Pro (Which all but one used) and windows 10 enterprise which was the odd one out of the list. From there I saw the hostname is BSTOLL-L but it did not give me the Fully Qualified Domain Name. After trying all the filters I could to attempt to find the domain name and looking at external sources like domain lookup tools with no luck, I found that if the host name is searched generally right after indexing, information about the host shows up including the computer name, which, in this case was the FQDN.

Answer: BSTOLL-L.froth.ly

Knowing the FQDN we can determine the source of the endpoint running a different operating system. If the server is uniform and only one operating system is the norm for this system, an endpoint on a different operating system and from an unknown domain could be very suspect.

**Evidence:**



---

## Conclusion

Based on the investigation, we can conclude the incident unfolded as follows

The s3 bucket "frothlywebcode" had its access control list changed to public using the API call "PutBucketAcl", the event ID for this is "ab45689d-69cd-41e7-8705-5350402cf7ac". Another analyst discovered this vulnerability and uploaded the text file "OPEN\_BUCKET\_PLEASE\_FIX.txt" using the API call "PutObject". This file served to alert the analysts of the issue and to prevent any further issues.

Thankfully, the file upload occurred within one minute of the bucket being made public, as shown in the timeline evidence (linked below). The consequences for this bucket remaining public for an extended period of time could include GDPR breaches, which could lead to loss of business license, fines and even legal prosecution.

**Timeline evidence screenshot** (<https://github.com/Harvey-Moth/Comp3010/blob/cb6aa285ecee3e2b082eff977680b7f7e895c7a1/Walkthrough%20Screenshots/Questions/Question%207/Timeline%20evidence.png>)

## References

[1]"Tier 1 vs. Tier 2 vs. Tier 3 Cybersecurity | ConnectWise," [www.connectwise.com](https://www.connectwise.com/cybersecurity-center/glossary/tier-1-vs-tier-2-vs-tier-3-cybersecurity). <https://www.connectwise.com/cybersecurity-center/glossary/tier-1-vs-tier-2-vs-tier-3-cybersecurity>

## 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.

<b>Solo Work</b>	<b>S1 - Generative AI tools have not been used for this assessment.</b>	<input type="checkbox"/>
<b>Assisted Work</b>	<b>A1 – Idea Generation and Problem Exploration</b> 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"/>
	<b>A2 - Planning &amp; Structuring Projects</b> 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"/>
	<b>A3 – Code Architecture</b> 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"/>
	<b>A4 – Research Assistance</b> 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"/>
	<b>A5 - Language Refinement</b> 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"/>
	<b>A6 – Code Review</b> 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.	<input type="checkbox"/>
	<b>A7 - Code Generation for Learning Purposes</b> 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.	<input type="checkbox"/>
	<b>A8 - Technical Guidance &amp; Debugging Support</b> 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.	<input type="checkbox"/>
	<b>A9 - Testing and Validation Support</b> 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.	<input type="checkbox"/>
	<b>A10 - Data Analysis and Visualization Guidance</b> AI tools can help suggest ways to analyse datasets or visualize results (e.g.	<input type="checkbox"/>

	recommending chart types or statistical methods). Students must perform the analysis themselves and understand the implications of the results.	
	<b>A11 - Other uses not listed above</b> Please specify:	<input type="checkbox"/>
Partnered Work	<b>P1 - Generative AI tool usage has been used integrally for this assessment</b> Students can adopt approaches that are compliant with instructions in the assessment brief. Please Specify: Readme Formatting.	<input checked="" type="checkbox"/>

**Please provide details of AI usage and which elements of the coursework this relates to:**

Used to help format tables in the readme. Wrote the information as paragraphs and asked ChatGPT not to change the content, but to format it as a table for clarity.

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