COMP3010 CW2

Contents

Introduction:

In this report, I will be analysing the BOTSv3 dataset within Splunk. The BOTSv3 dataset is a public dataset made up of 2,083,056 events which simulates an incident to allow cyber security professionals to practice event analysis in a simulated environment. By utilising the 3 tiers of SOC – triage, investigation and hunting – I will answer a series of questions to determine how the incident occurred, including information about the user(s) and device(s) involved. This report will focus on 8 of those questions, explaining their relevance to SOC, as well as how to install Splunk and how to prepare the BOTSv3 dataset for analysis.

Screenshots can be found here: https://github.com/Mark7567/COMP3010

Video walkthrough: https://www.youtube.com/watch?v=ijebfZ6ZJcI

SOC Roles and Incident Handling Reflection:

SOC:

The Security Operations Centre (SOC) consists of 3 tiers, each having unique roles and responsibilities surrounding cyber security prevention and analysis. Tier 1 - triage - involves monitoring dashboards and events to determine if an incident has occurred. Since attacks cannot be 100% prevented, tier 1 acts as a way to catch incidents early, as well as filter out false positives which prevention methods may have caught.

Once an event has been flagged as an incident, tier 2 - investigation - takes place. This involves checking logs and confirming that an incident has occurred and containing them to prevent any more damage from occurring. Tier 2 also involves determining how the incidents occurred by performing an in-depth analysis.

Lastly, tier 3 - hunting - involves performing an advanced analysis on the incident and improving overall security following it. The BOTSv3 dataset focusses predominantly on tier 2 of SOC, since the dataset contains incidents which have already occurred, with the purpose of the dataset being to confirm this and investigate how they happened.

Prevention:

Prevention is an incident handling methodology which involves mitigating the risk of an incident occurring rather than directly stopping it, since it is not possible to 100% prevent an attack. Rules and proper system configurations are implemented to make a system as secure as it can be in an attempt to reduce the risk of successful attacks. Prevention relates to the BOTSv3 exercise since the events in the dataset can help to determine normal user behaviour which may make any outliers more obvious as potentially malicious. Furthermore, it links with tier 1 of SOC as there are links with monitoring dashboards and analysing behaviour.

Detection:

Detection is an incident handling methodology that involves identifying incidents which could be considered suspicious and may indicate an incoming incident. Typical detection methods include analysing network traffic and timestamping user actions or system events to find concerning behaviour. The BOTSv3 dataset involves using filters and searching within Splunk to

analyse millions of events and identify information about them, such as which users were involved and which operating system their device was running. This links to tiers 1 and 2 of SOC since detection involves flagging potentially malicious events and analysing them to determine their causes.

Response:

Response is an incident handling methodology which involves containing an incident to prevent it from causing any more damage to a system, as well as investigating how the incident occurred. The BOTSv3 dataset contains a variety of incidents which need to be confirmed as valid, as well as investigating into how they occurred by gathering information about the events using Splunk filters. Since it is a premade dataset, Splunk cannot contain the incidents in any way, but it can be used to investigate them. Response has strong links with tier 2 of SOC as the purpose of this methodology is to investigate how the incident occurred, which is also the main idea behind SOC tier 2.

Recovery:

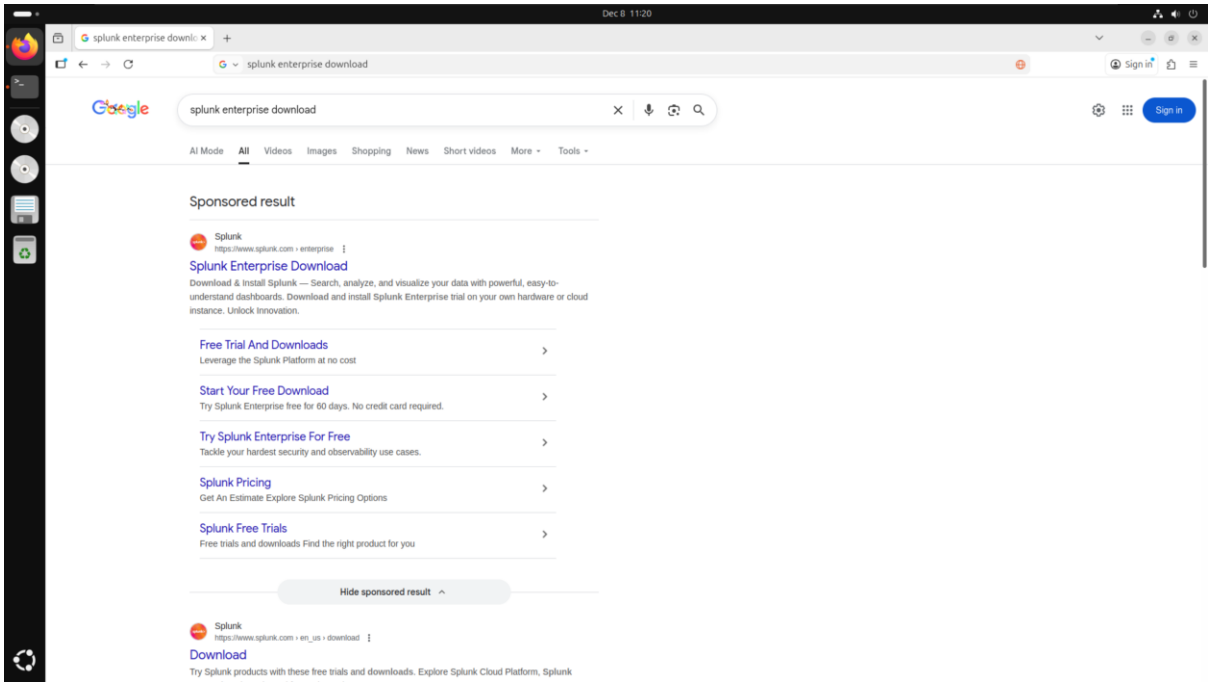
Recovery is an incident handling methodology which involves restoring damaged systems after an incident has been resolved. It also includes improving security to reduce the risk of a similar incident happening in future. While there is no active recovery within the BOTSv3 dataset, it helps with this methodology as the dataset is premade to contain events for analysis which can then be reviewed to prevent further incidents.

Installation and Data Preparation:

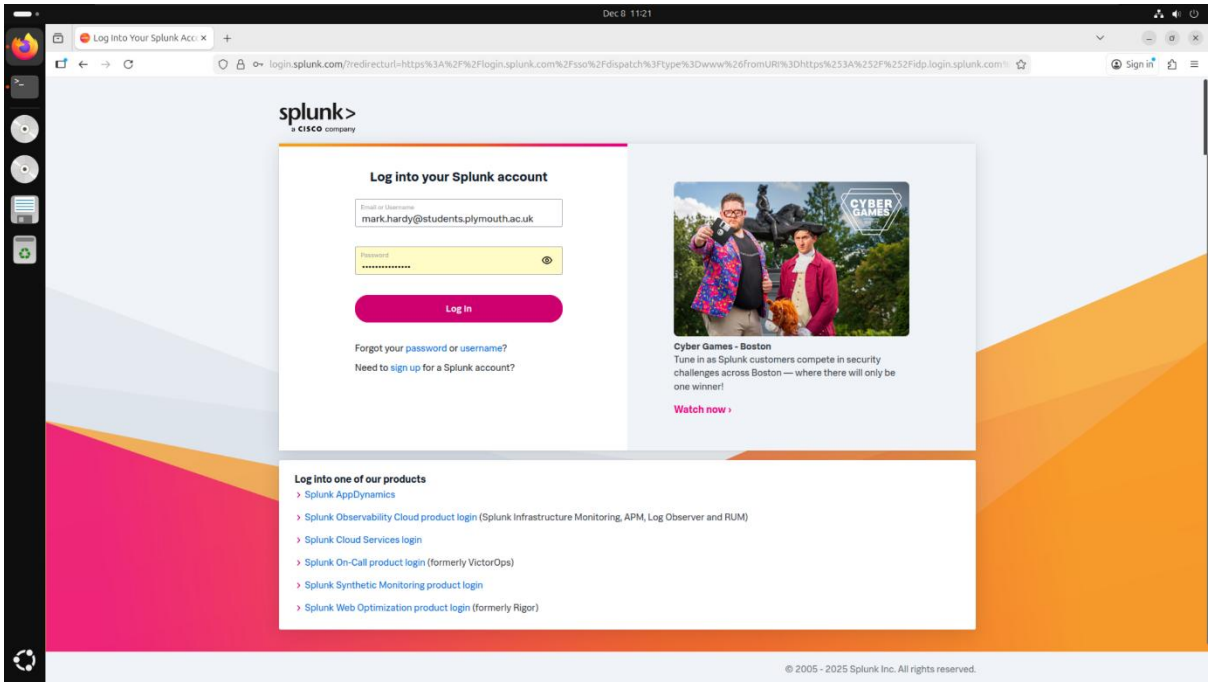
This section will explain how Splunk was installed within an Ubuntu Linux virtual machine hosted by VMWare, and how the BOTSv3 dataset was loaded. As I performed the installation before taking screenshots for this section, the screenshots display later timestamps compared to those for the guided questions. To effectively demonstrate the installation of Splunk, I reinstalled it on the same virtual machine ensuring the configuration of the machine remained the same throughout this report. The reason for using a Linux virtual machine is to mimic real world SOC environments which would typically use Linux-based systems. By installing Splunk locally on my device, this allowed me to have full control over the configuration, and which data was ingested, keeping the integrity and accuracy of the analysis.

Splunk Installation:

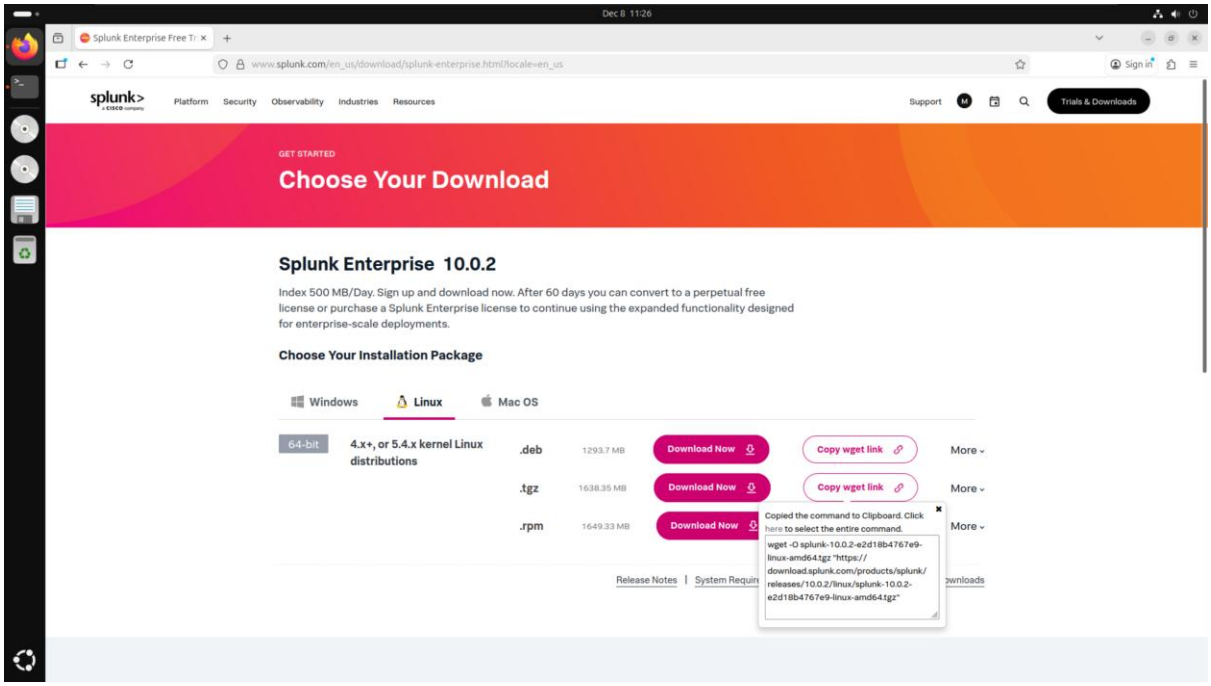
I opened Firefox inside an Ubuntu Linux virtual machine and searched for “splunk enterprise download”, clicking on the first link in the search results.



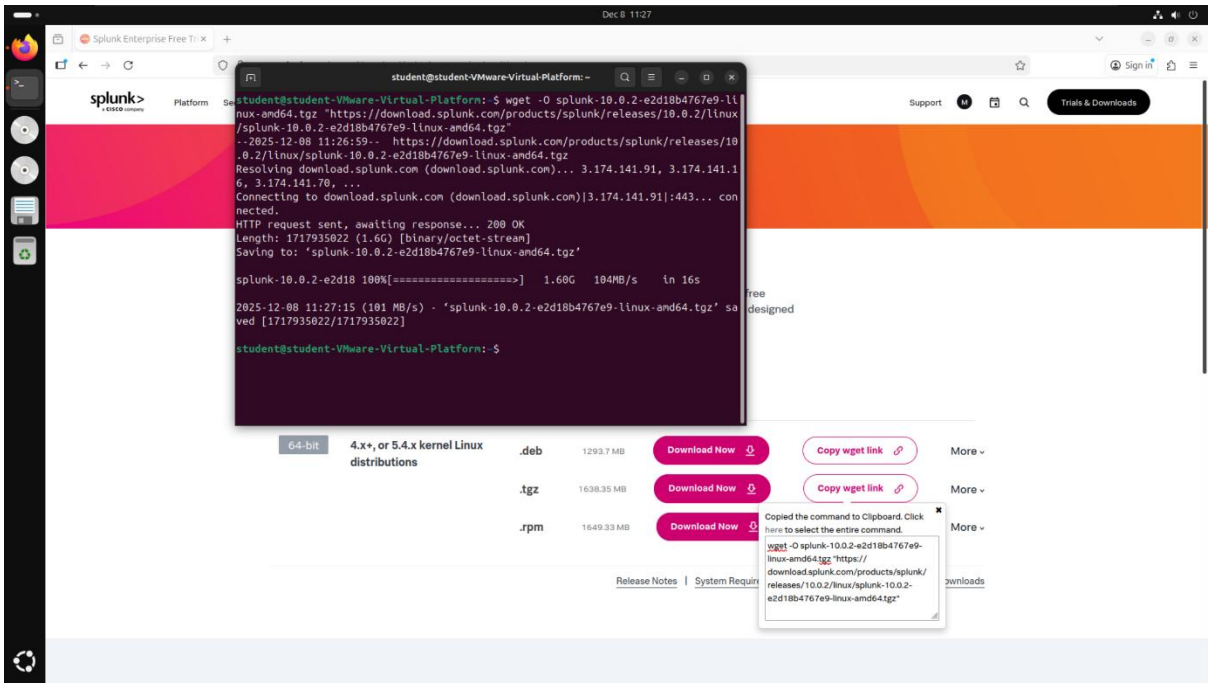
This link took me to the official Splunk website, where I logged into an account which I had already created.



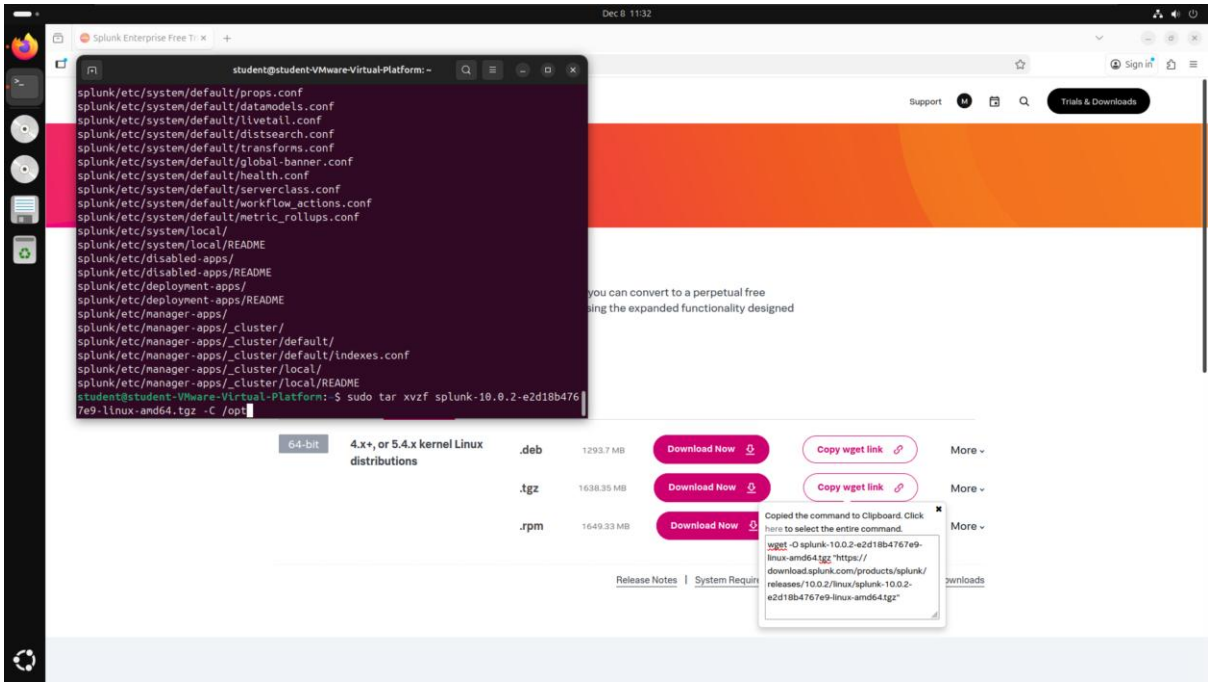
I selected “Copy wget link” for the .tgz version of Splunk, to ensure that all packages were downloaded alongside the main Splunk software. This meant I did not have to individually download all relevant packages to complete this coursework.



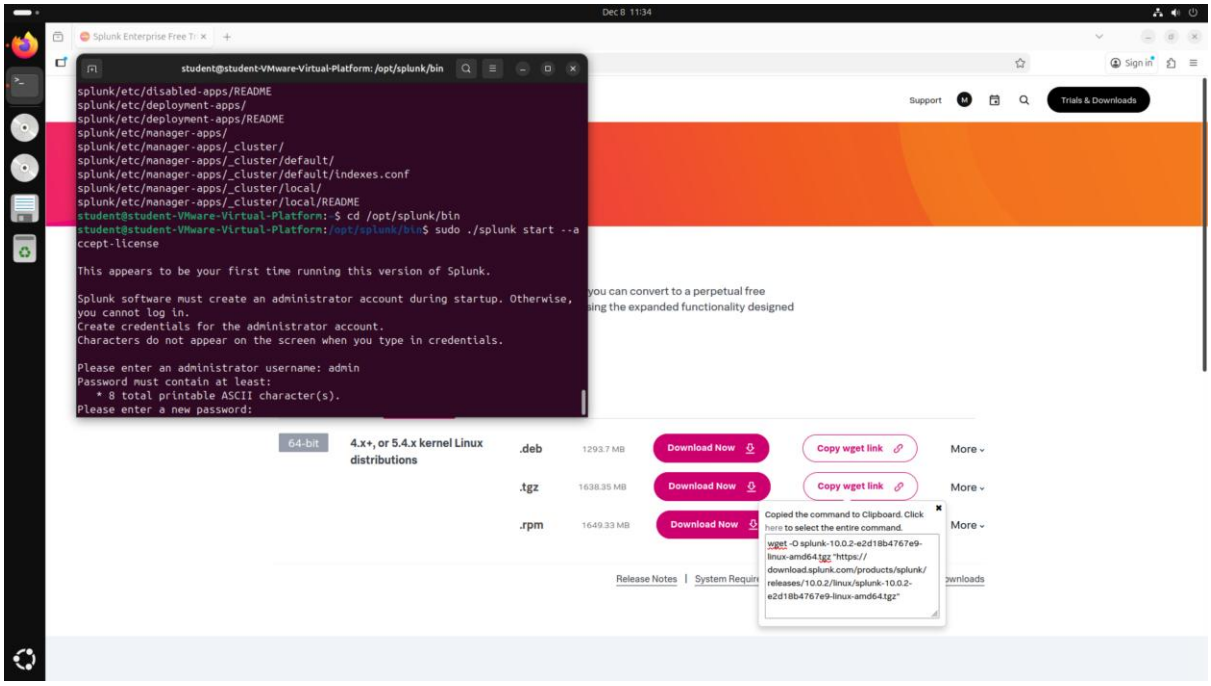
I downloaded Splunk directly from the Linux terminal by pasting the wget link directly into the terminal and running it. I received a confirmation message in the terminal to state that Splunk had been successfully downloaded.



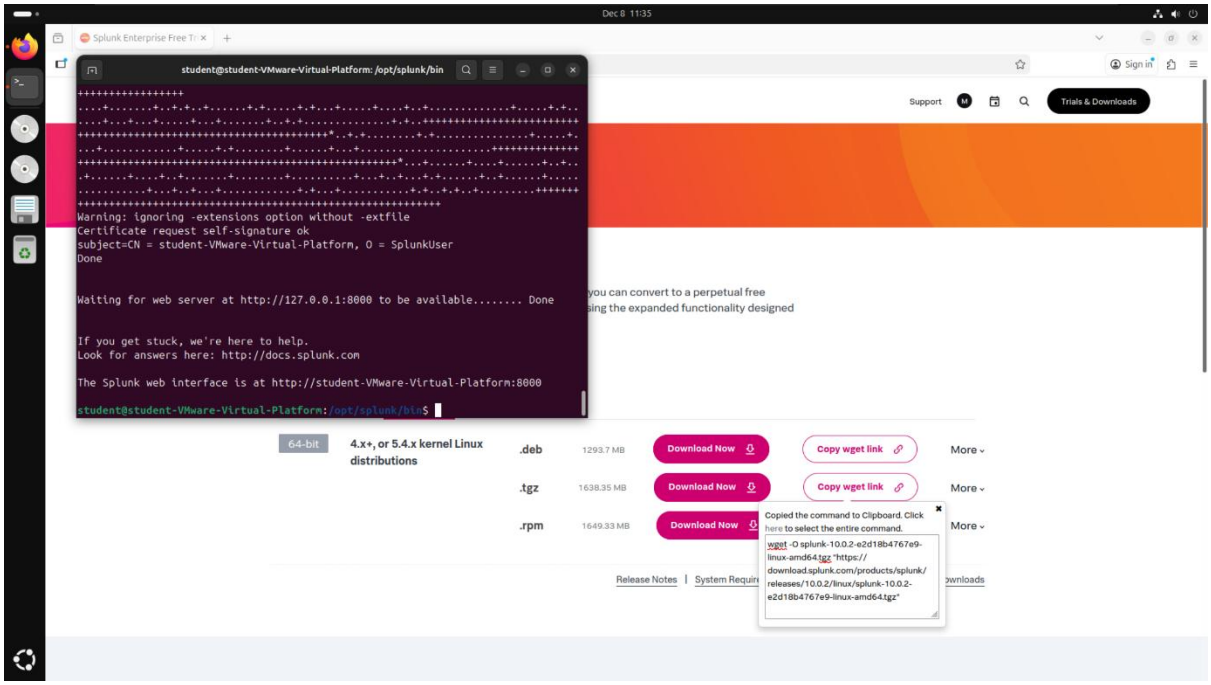
I ran the command “sudo tar xvzf splunk-10.0.2-e2d18b4767e9-linux-amd64.tgz -C /opt” which installed Splunk and all its dependencies into the /opt folder.



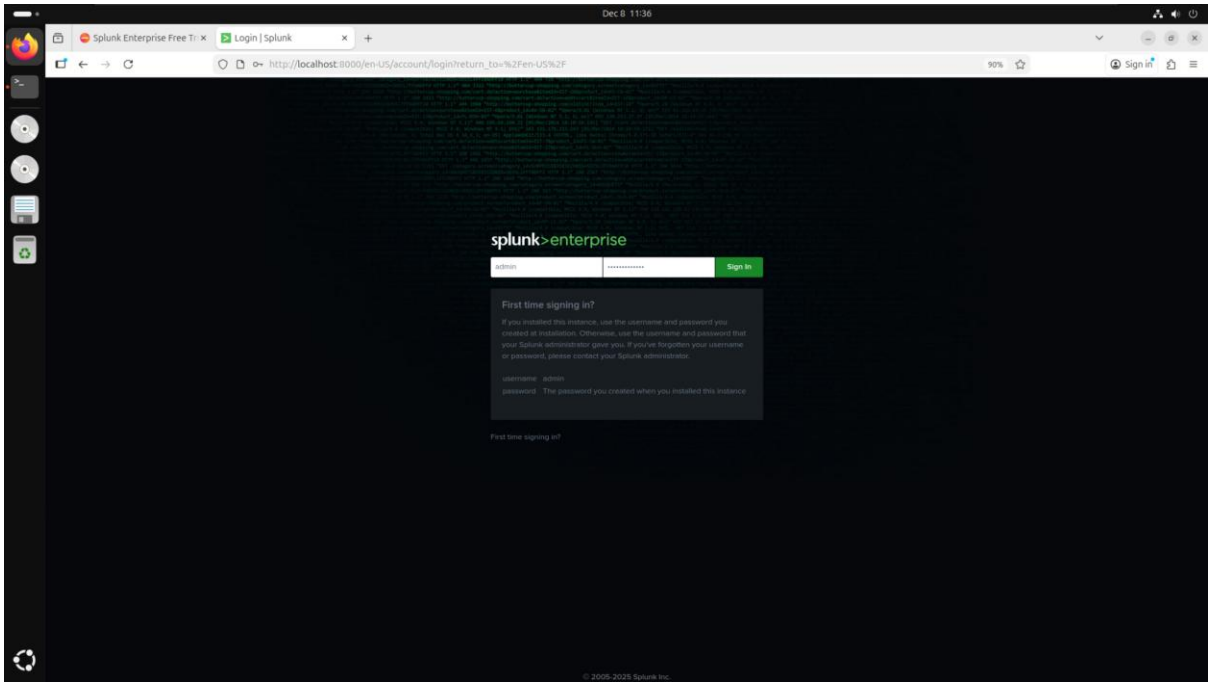
To start Splunk, I ran the command “sudo ./splunk start” from within the “/opt/splunk/bin” folder. Since this was my first time running Splunk on this installation, I had to include the command “--accept-license” which accepted the user license agreement for Splunk. I had to set a username and password for the administrator account which I could then use to sign in once on localhost:8000.



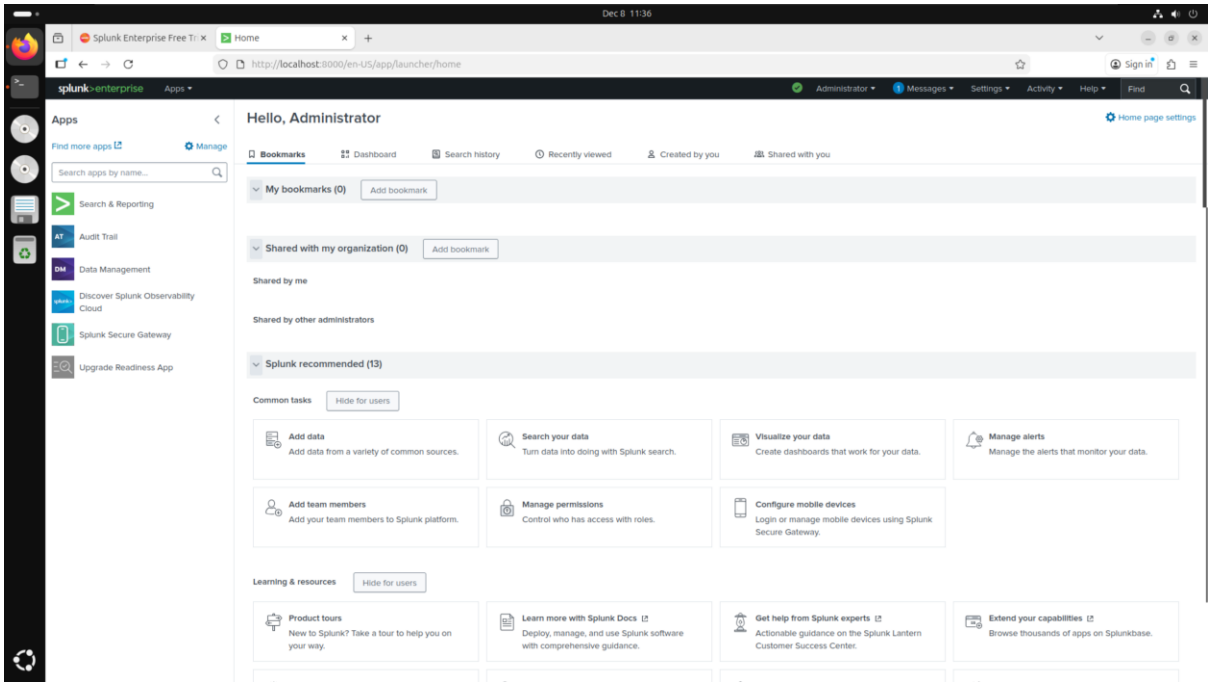
After setting up the initial login details, Splunk generated the web server on localhost:8000, providing a clickable link to the website to allow access into Splunk.



Searching for localhost:8000 on Firefox took me to the Splunk enterprise login page. I entered the username and password I created previously.

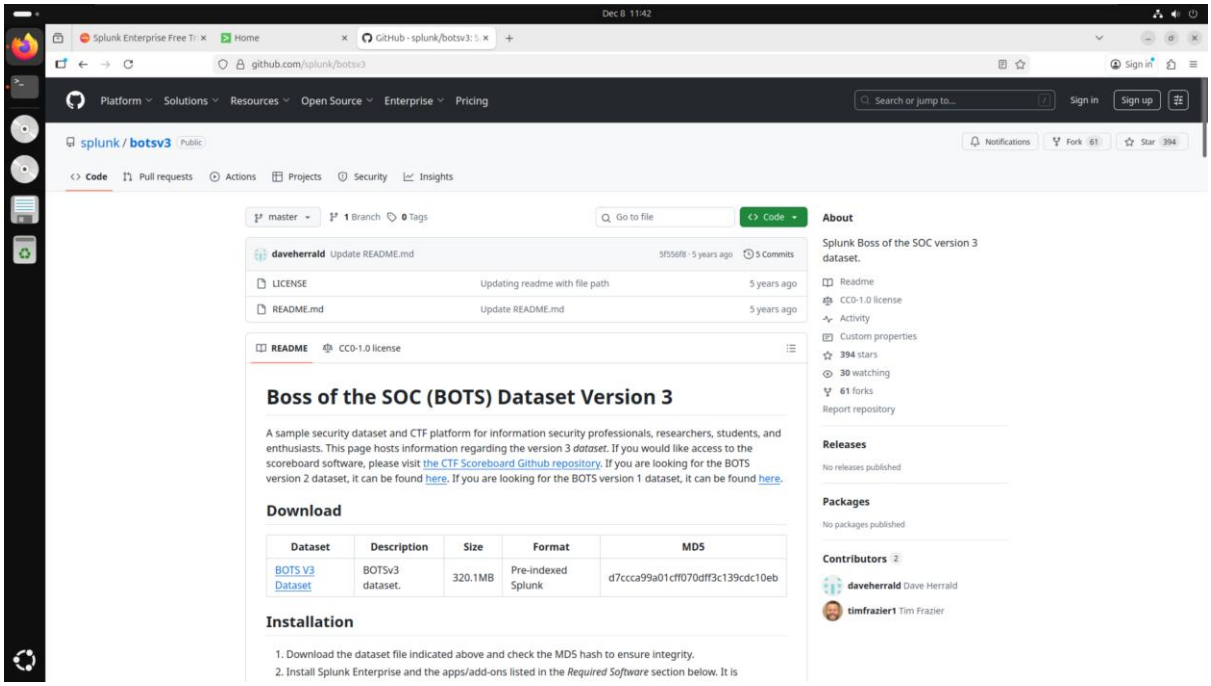


I was able to see the dashboard which houses all of the features I used to answer the guided questions, most notably the “Search & Reporting” tab, which will be explained below.

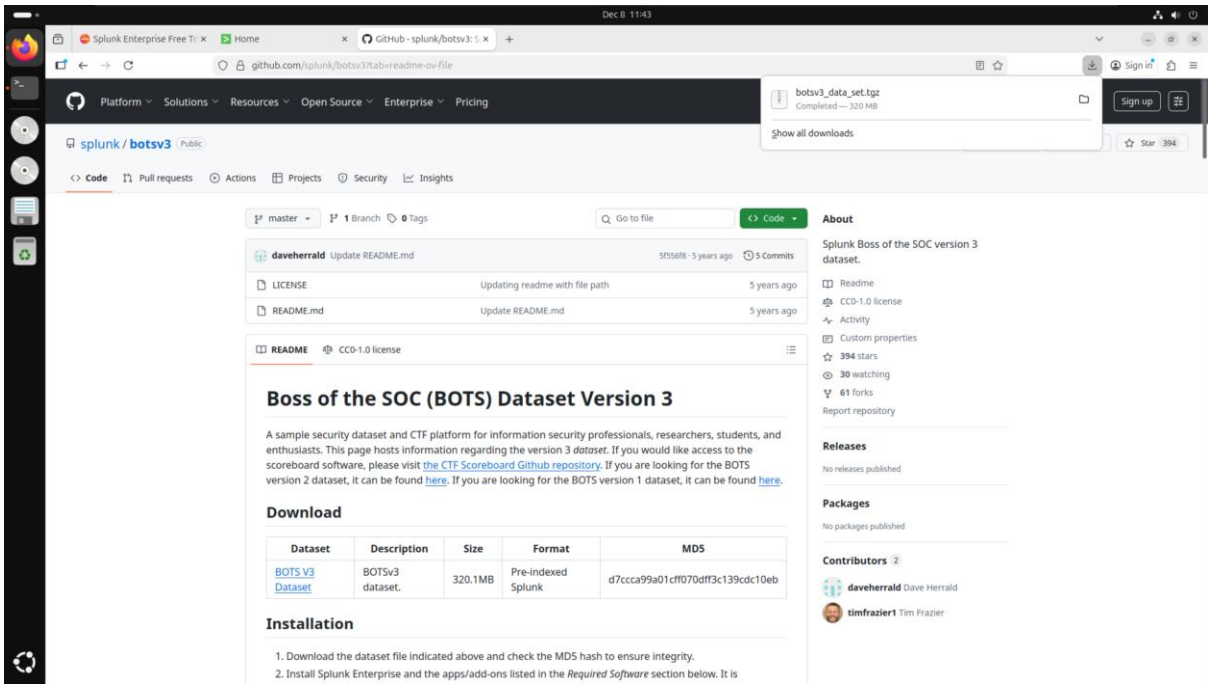


BOTSv3 Dataset Preparation:

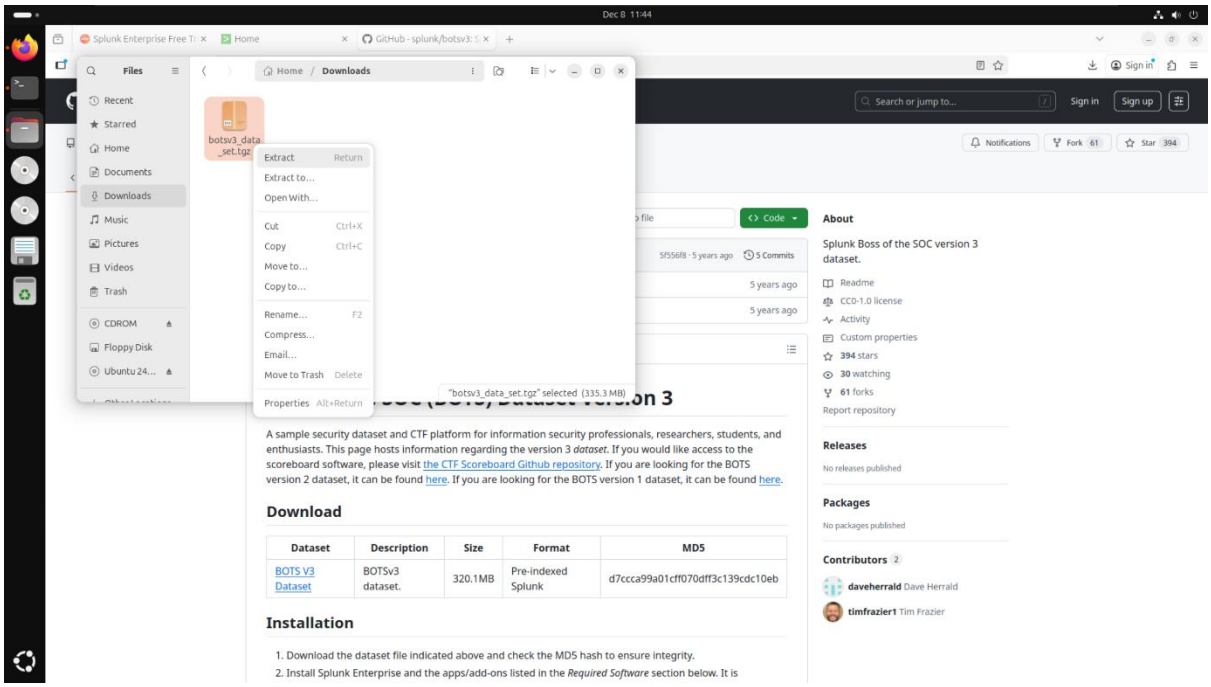
I visited the link https://github.com/splunk/botsv3 and clicked on the link under dataset in the table to download BOTSv3 to my virtual machine.

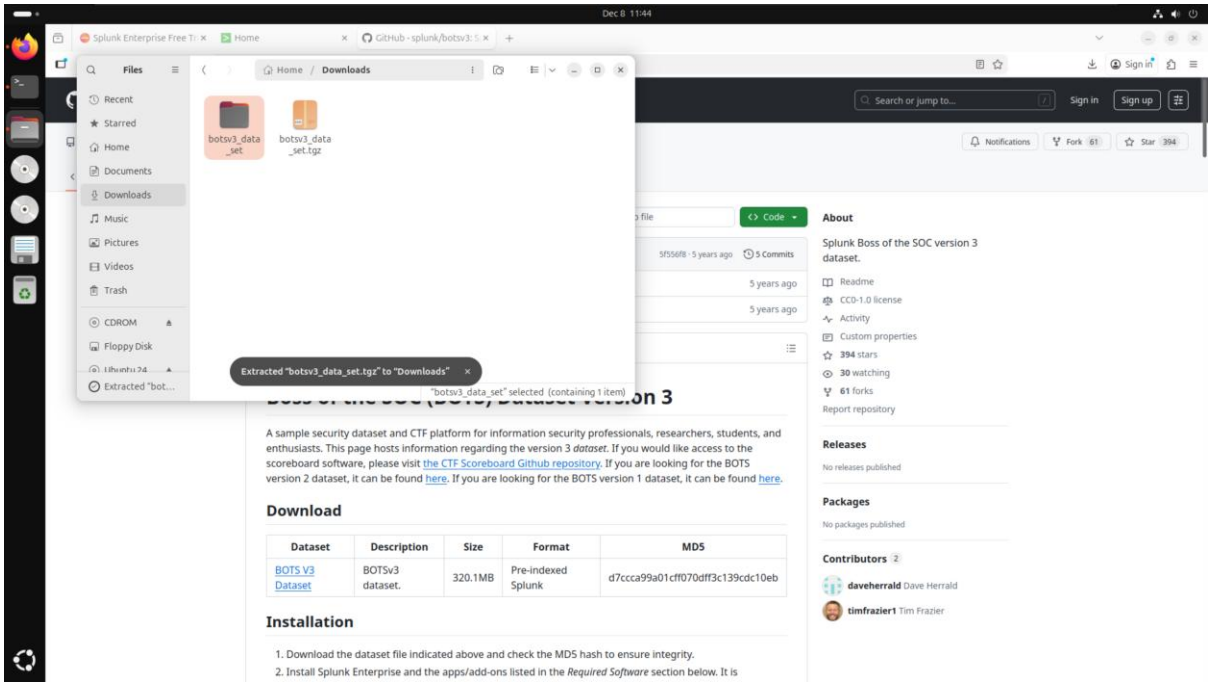


Clicking on the link downloaded the file botsv3_data_set.tgz onto my virtual machine as shown in the downloads tab on Firefox.

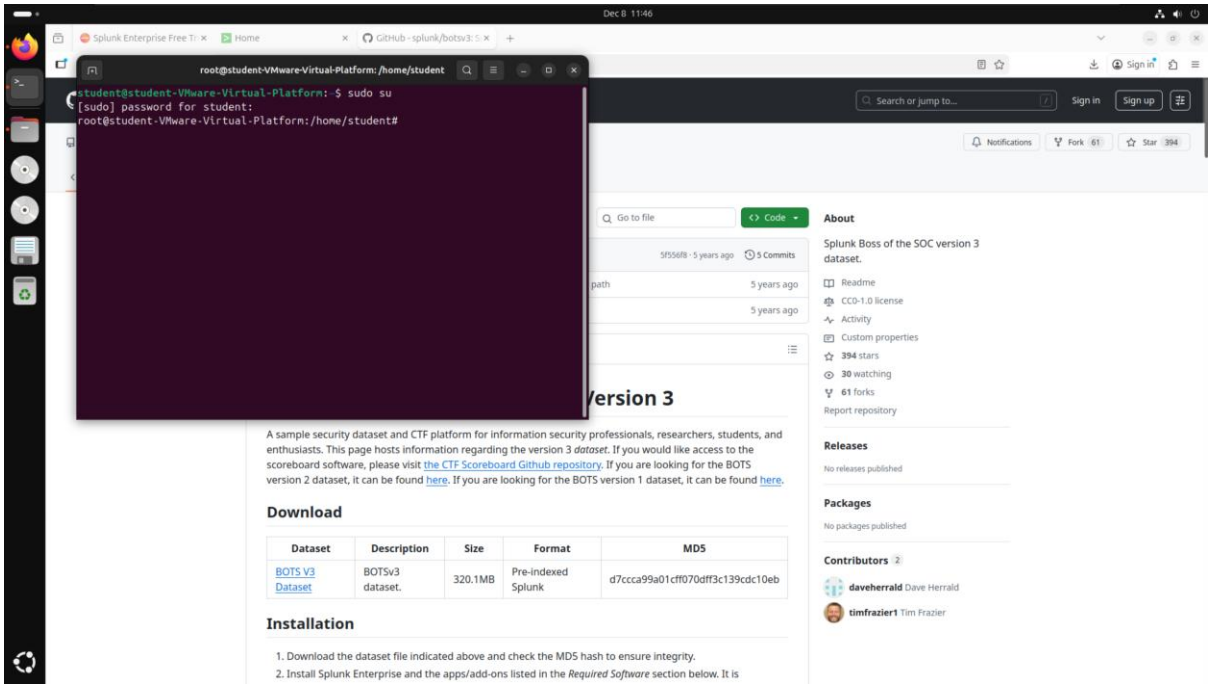


I opened up my file explorer and found the file in the downloads folder where I then extracted it.

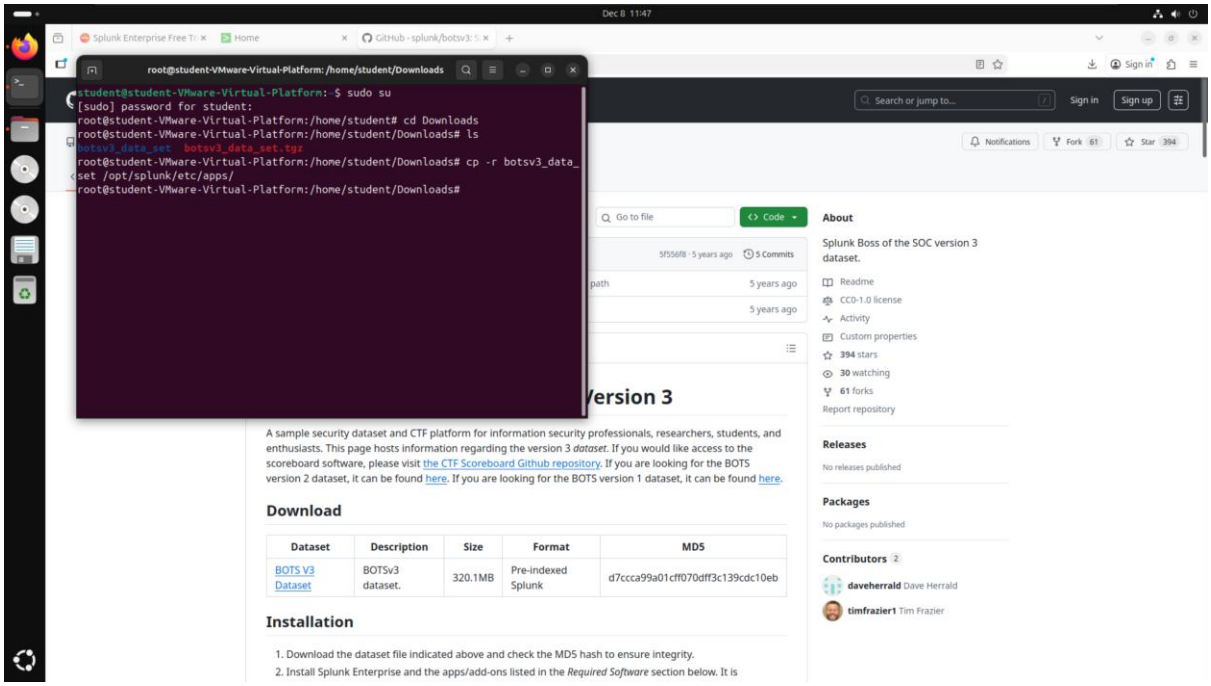




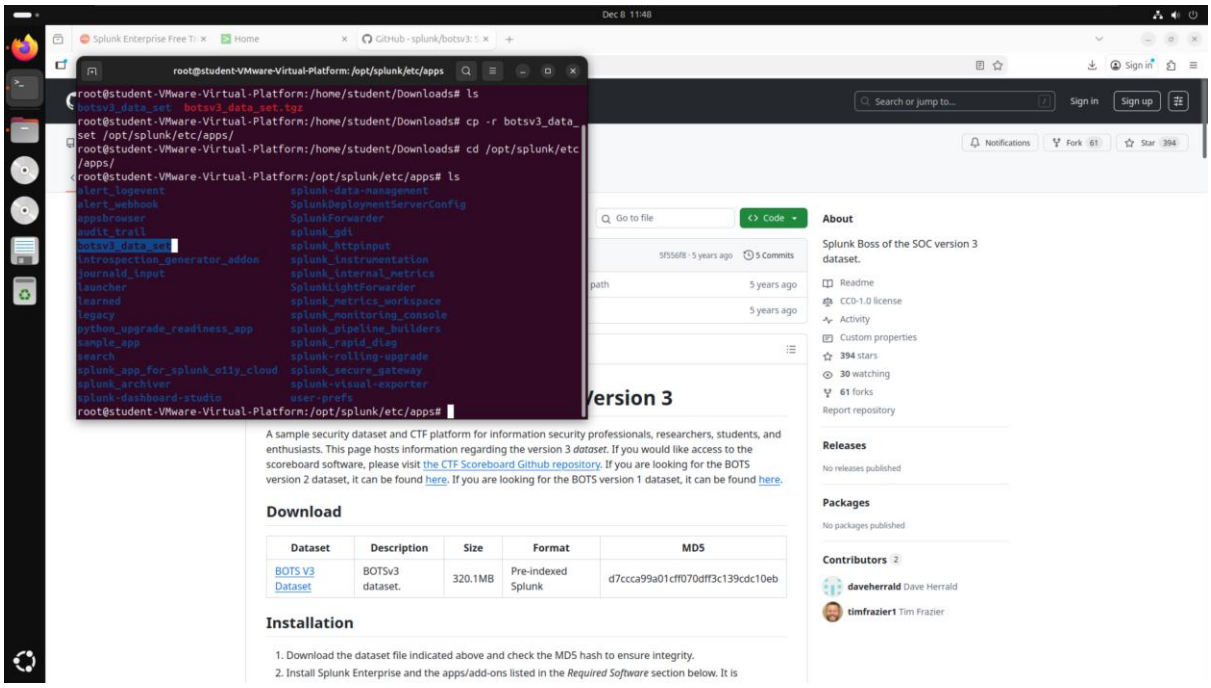
To copy the folder to the correct path for Splunk, I entered root mode in the Linux terminal by running the command “sudo su” and entering the password for my virtual machine.



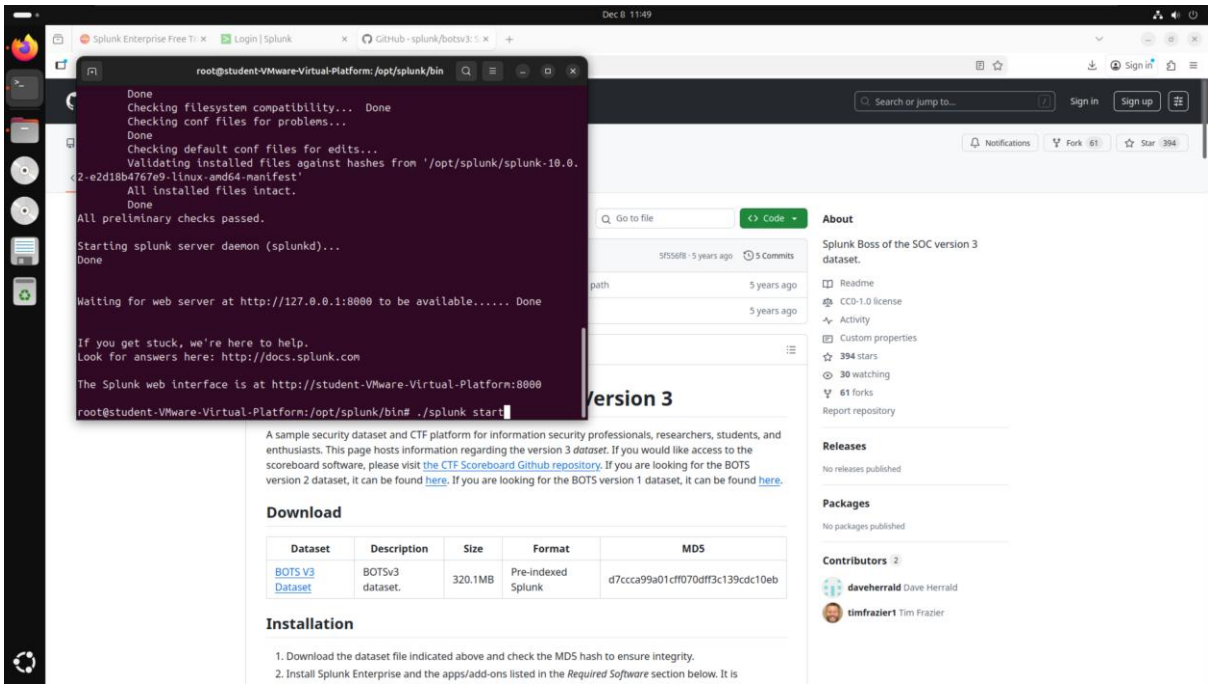
Once in root mode, I ran the command “cp -r botsv3_data_set /opt/splunk/etc/apps/” to copy the file (botsv3_data_set) into the correct path (/opt/splunk/etc/apps/) to allow it to load into Splunk correctly.



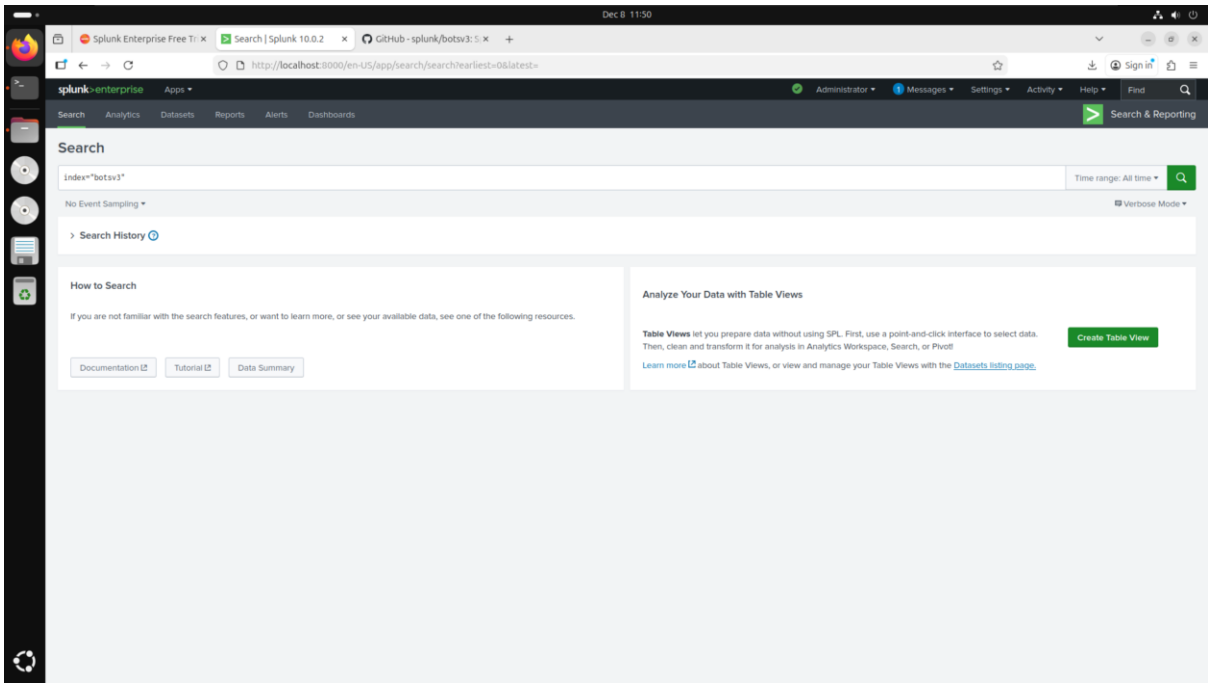
I navigated to the correct path to confirm that the file had been correctly copied.



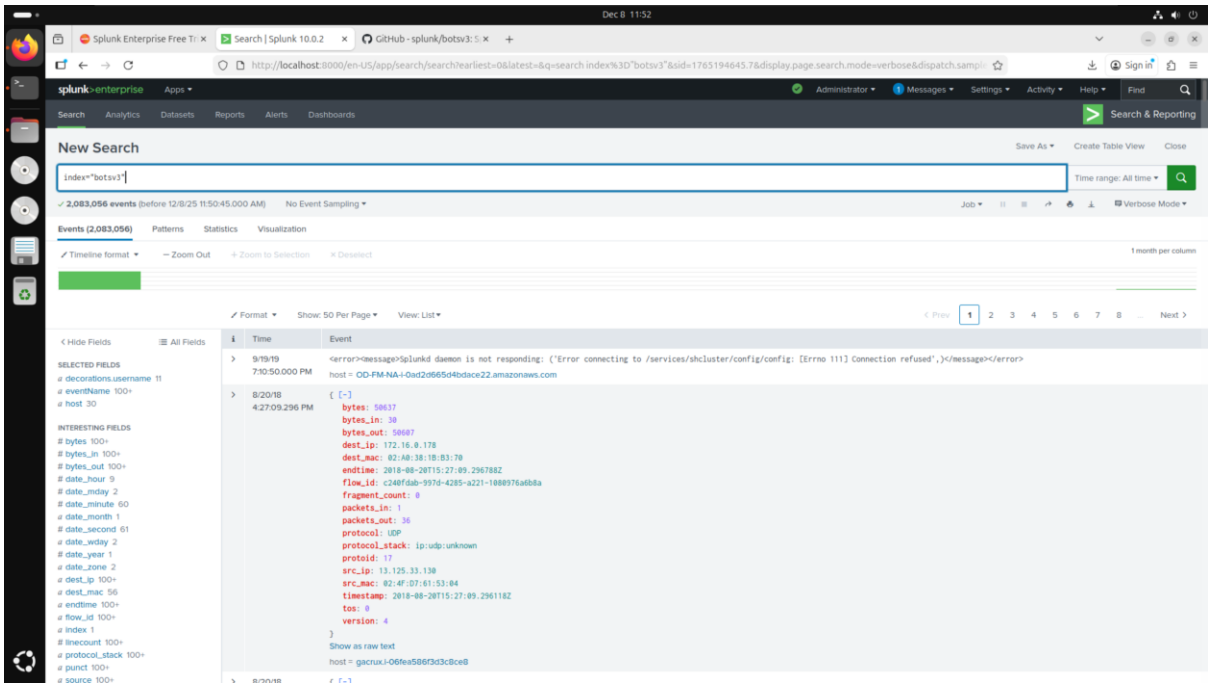
I started Splunk using the command “./splunk start”. I did not have to include the “sudo” command at the beginning of the line as I was still in root mode.



I opened Splunk by searching for localhost:8000 on Firefox. I navigated to the Search & Reporting page then applied the filter index=”botsv3”.



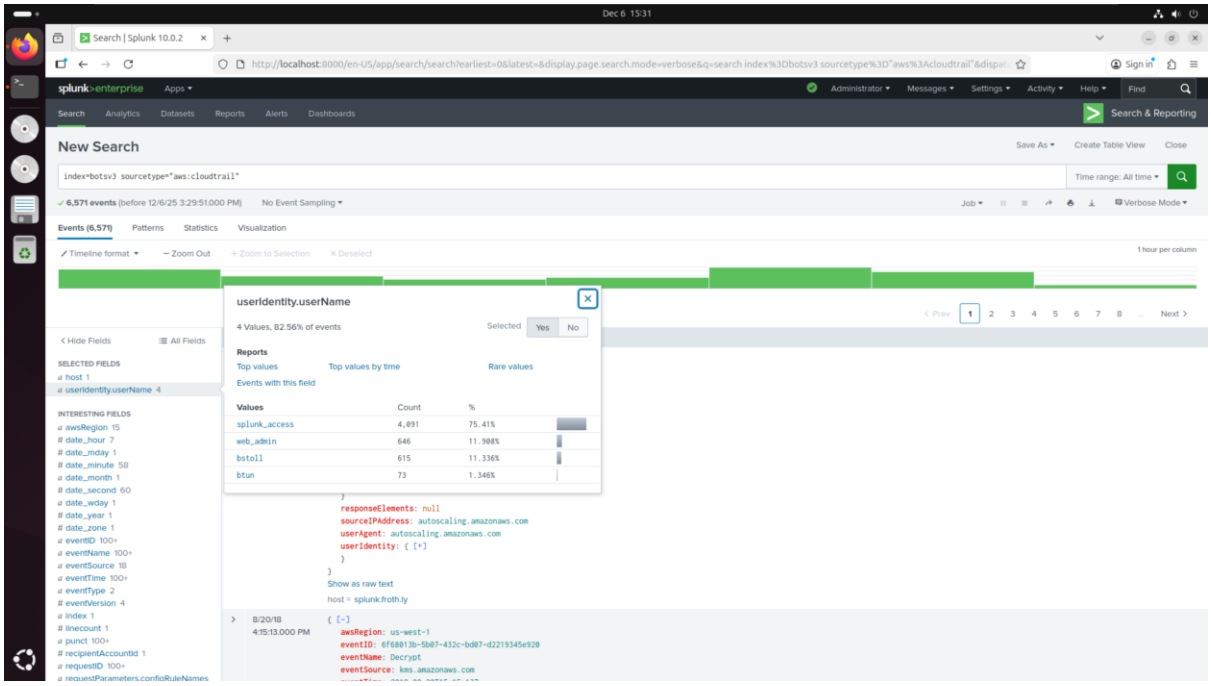
Running the search with the filter index=”botsv3” and setting the time range to filter for all time, I confirmed that the dataset had correctly installed as I had the correct number of 2,083,056 search results. This meant my data was correctly prepared and ready for me to answer the questions. Validating the correct number of events was important to keep data integrity, which is critical in SOC environments since incorrect data could lead to incorrect results.



Guided Questions:

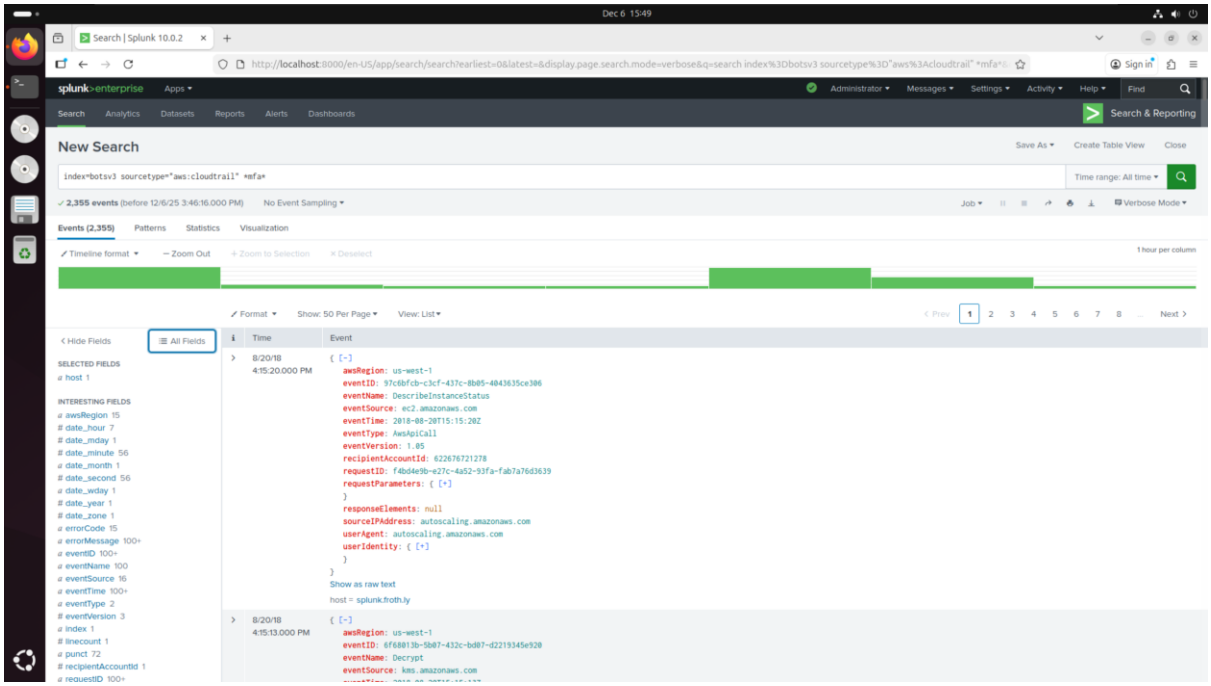
This section answers some of the 200-level questions relating to BOTSv3, showing a typical real-world SOC workflow, determining an attacker and identifying information surrounding an incident. There are 8 questions total that I will answer, outlining which Splunk filters I used and what my results were.

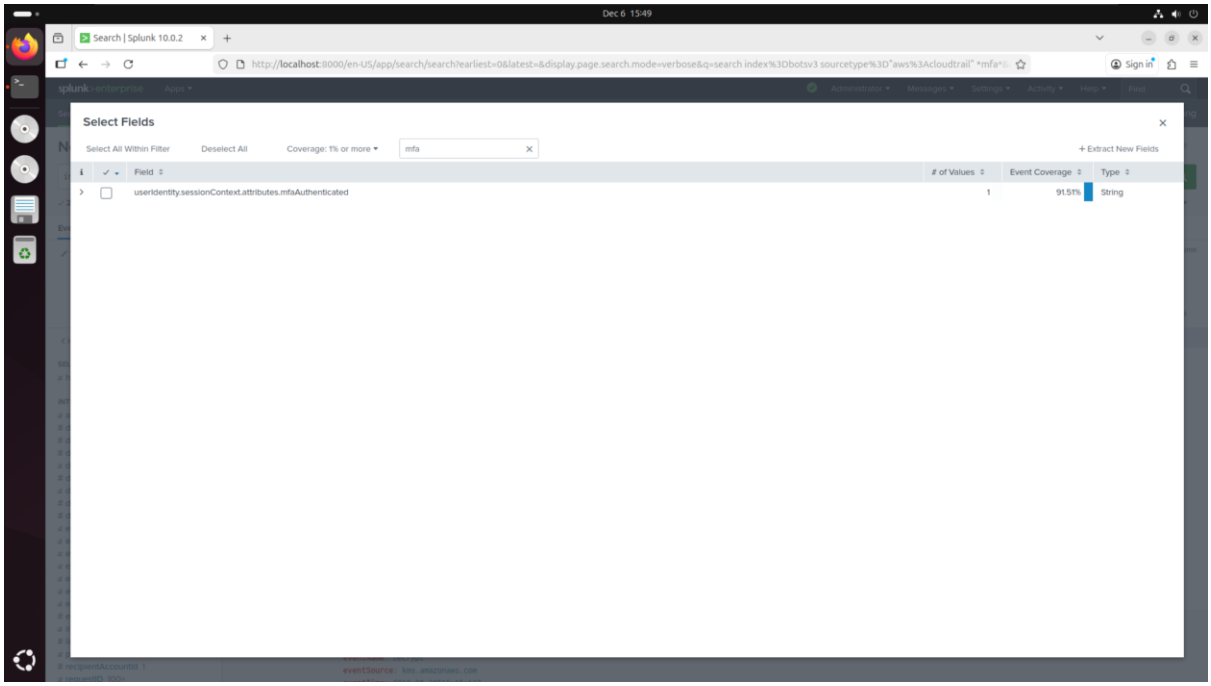
Question 1:



I used the provided hint to filter Splunk by sourcetype=”aws:cloudtrail”. I then searched through the available filters to find anything which could be related to usernames. Here I found the filter for “userIdentity.username”, which I then applied to show me the 4 usernames; bstoll, btun, splunk_access and web_admin. This links to tier 1 of SOC as I am monitoring the dashboards to find if an incident has occurred by checking which users have access.

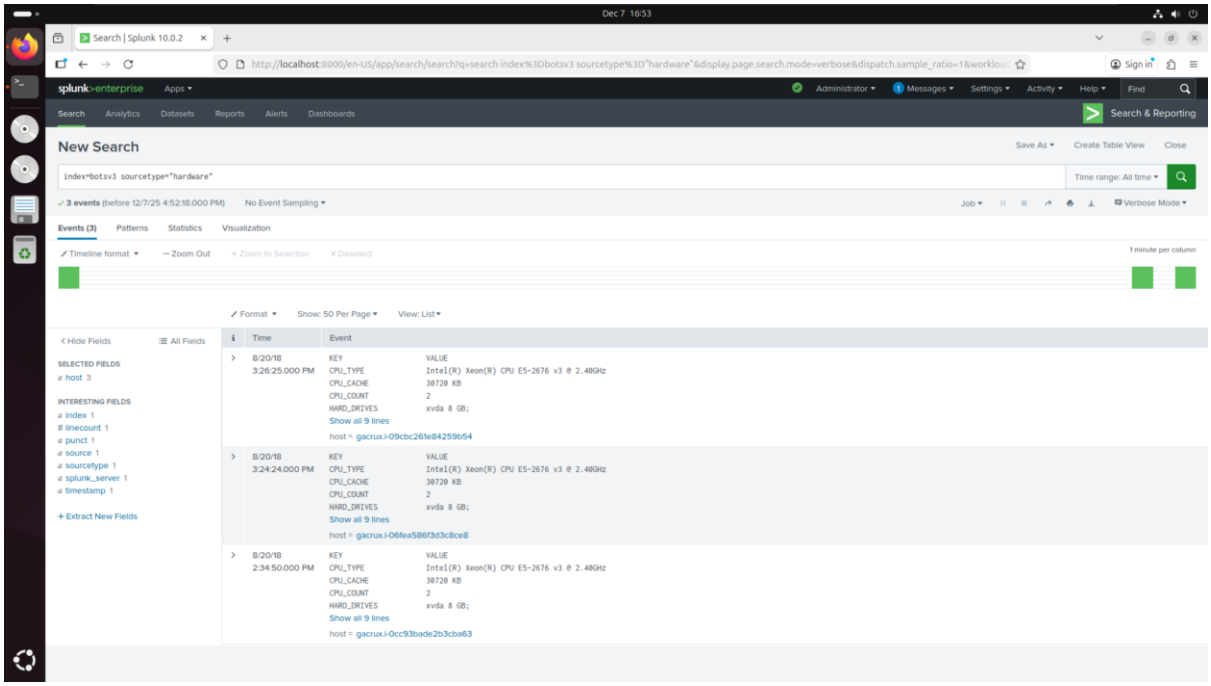
Question 2:





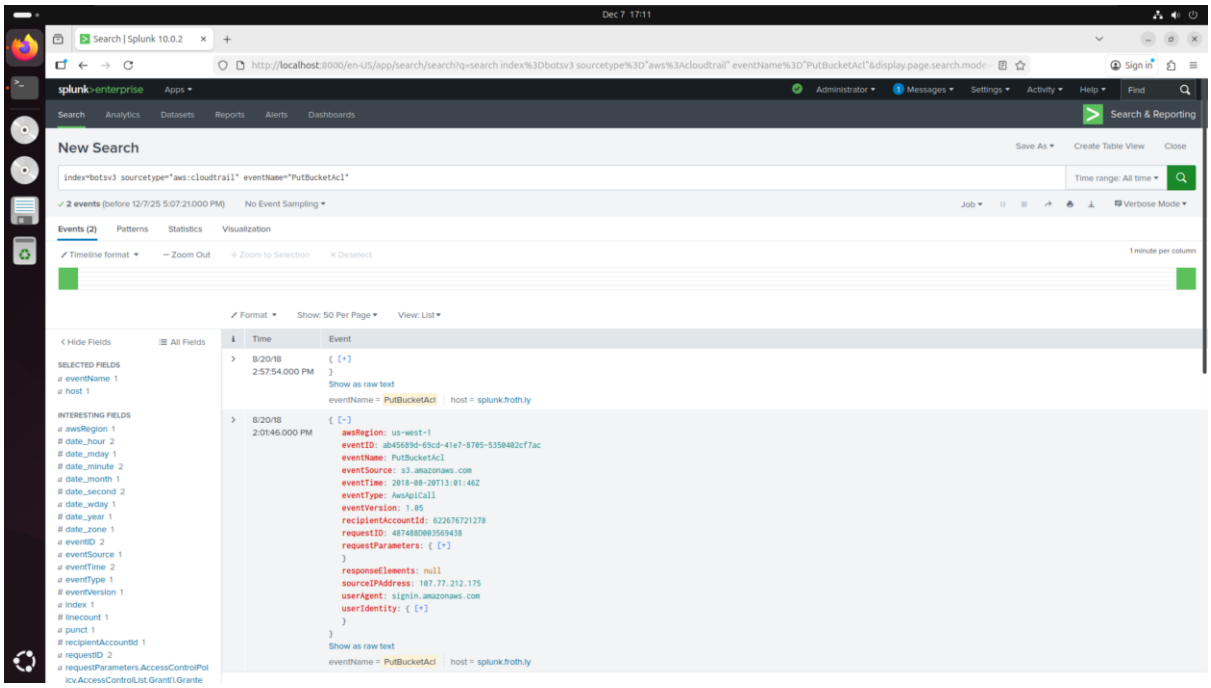
I kept the same filter as question 1 and searched additionally for *MFA* using the provided hint to find anything relating to multi-factor authentication. I then searched for “mfa” in the additional filters which led me to find the filter userIdentity.sessionContext.attributes.mfaAuthenticated.

Question 3:



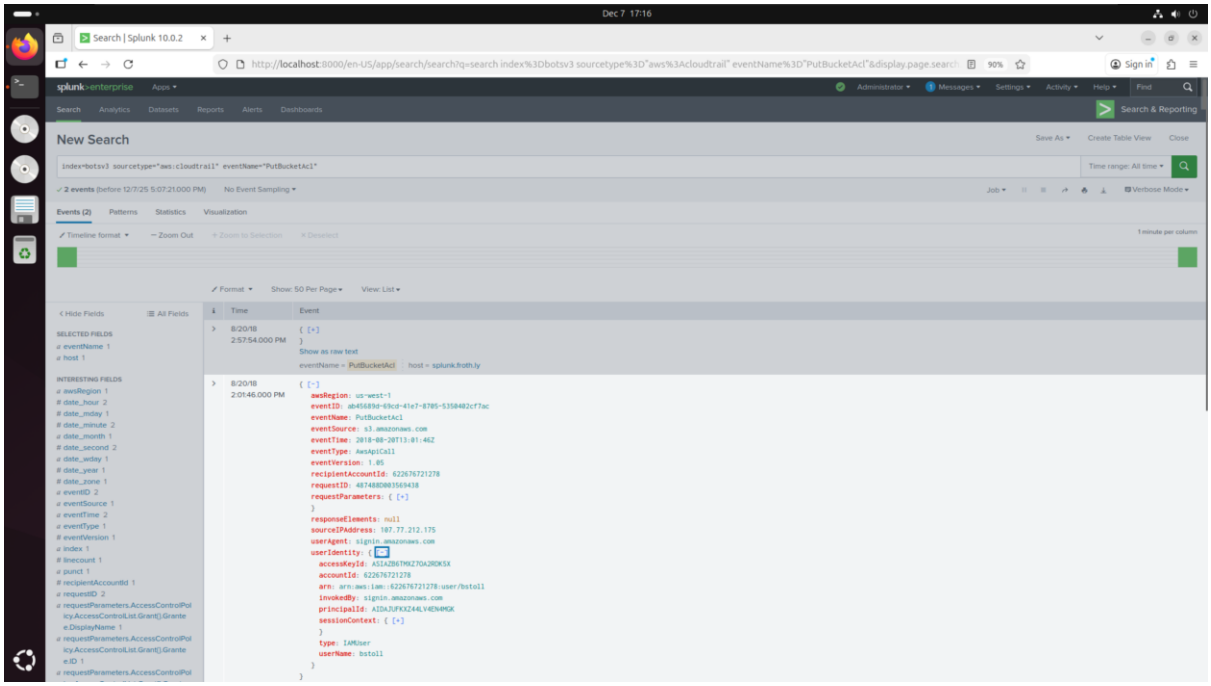
I changed the filter to be sourcetype=”hardware” using the provided hint, which then showed me the list of hardware being used to connect to the server. Looking into these values, I found the value CPU_TYPE which was listed as the model number E5-2676.

Question 4:



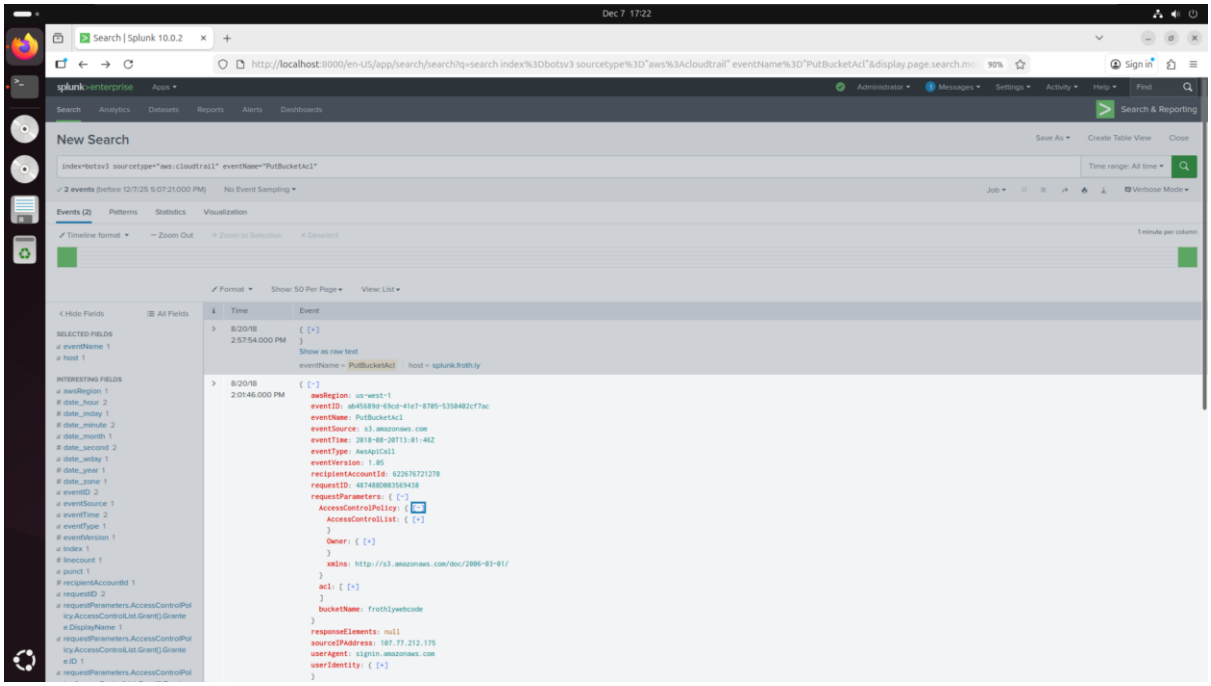
I changed the filter back to sourcetype=”aws:cloudtrail”, and added a filter for eventName=”PutBucketAcl” based on the provided hints. From here, I noticed 2 separate events with different event IDs. I assumed that one event would be the user making the S3 bucket public, and the other event would be the user making the S3 bucket private again. Taking the ID of the newest event, this gave me the answer of ab45689d-69cd-41e7-8705-5350402cf7ac.

Question 5:



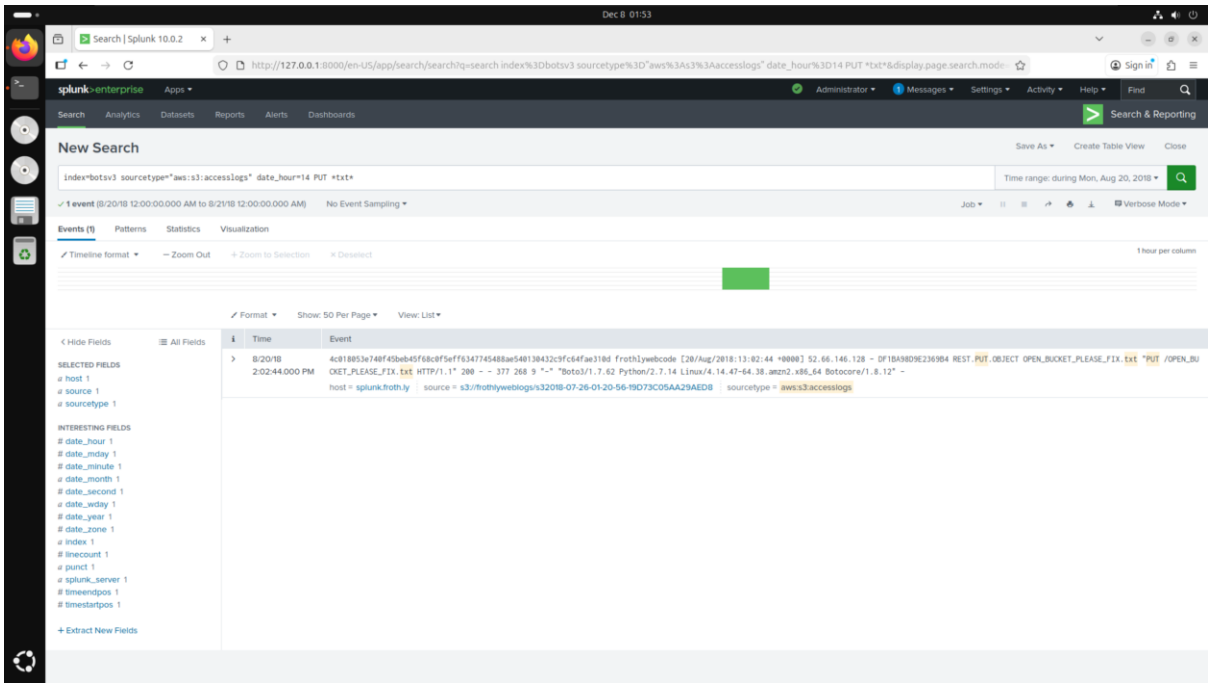
I applied the same filters as question 4. From there, I opened the userIdentity field within the relevant event and found that the username was bstoll.

Question 6:



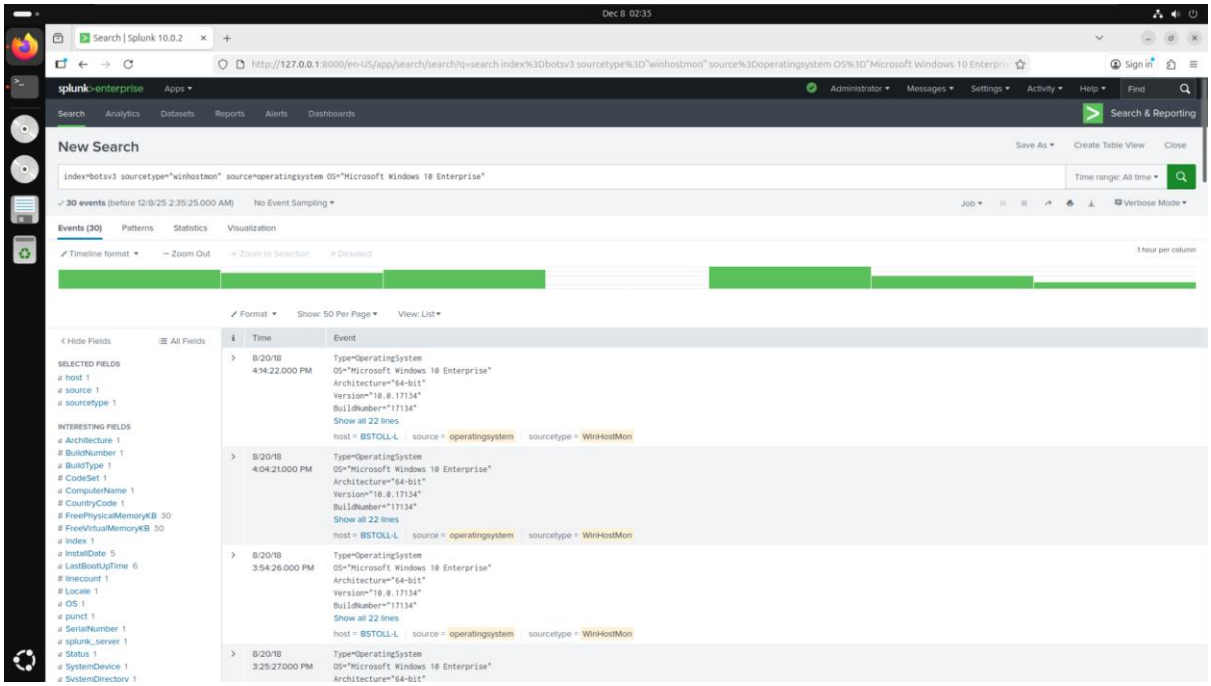
I applied the same filters as question 4. From there, I opened the requestParameters field within the relevant event and found that the bucketName was frothlywebcode.

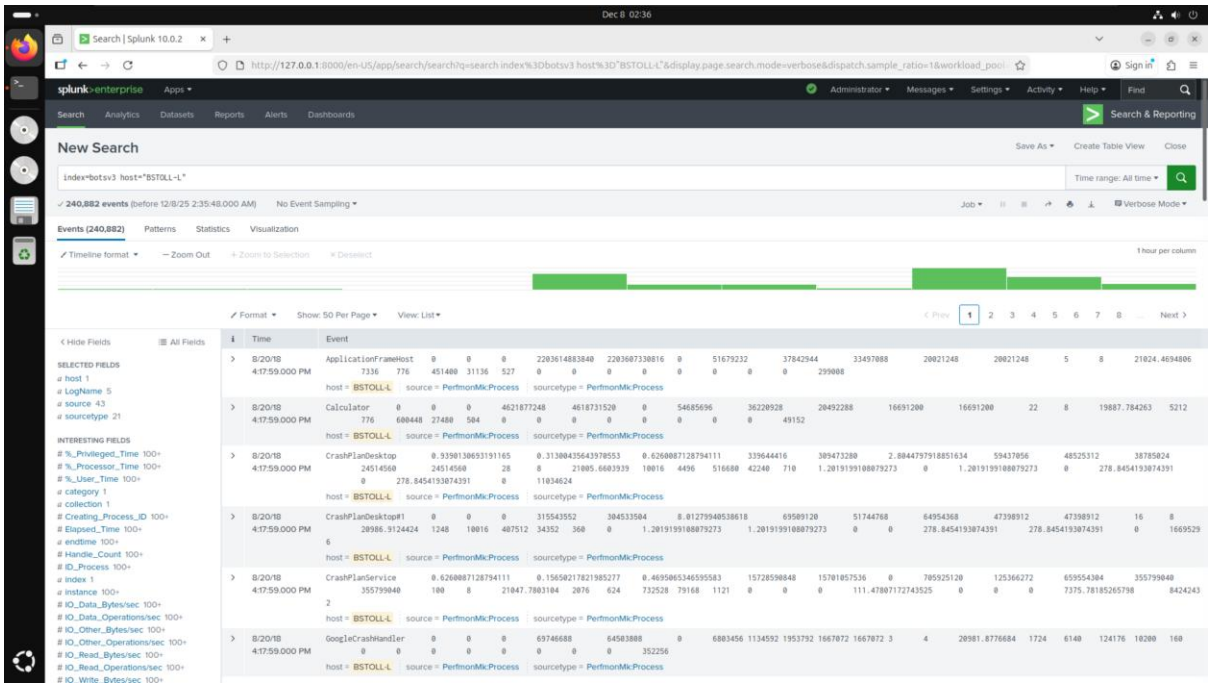
Question 7:

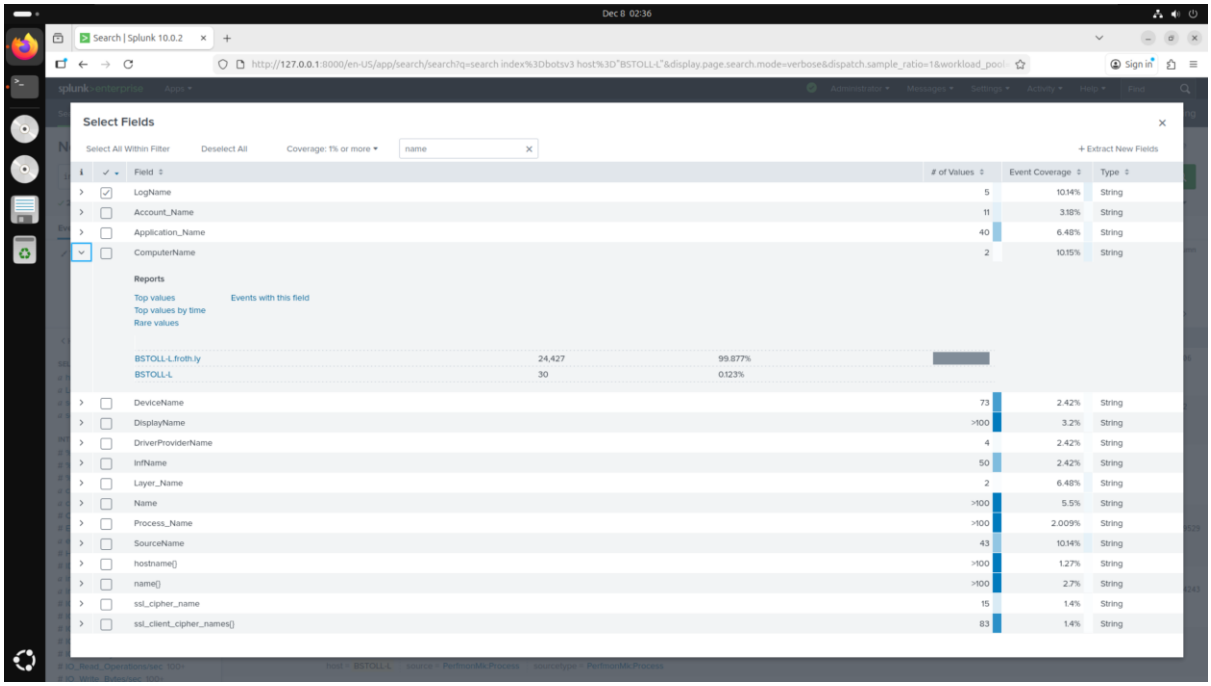


I changed the filter to sourcetype=”aws:s3:accesslogs” based on the provided hint. Then, knowing that the events in questions 4 to 6 occurred between 2pm and 3pm, I added a filter for date_hour=14 to check specifically between these times. Next, I added another filter for HTTP PUT requests, since the question involved uploading a file to the S3 bucket. I lastly added a filter for *txt* since the question mentioned a text file being uploaded which hinted to me that it would be a .txt file. This led me to find the uploaded text file “OPEN_BUCKET_PLEASE_FIX.txt”.

Question 8:


Dec 8 02:35
Search | Splunk 10.0.2
splunk>enterprise
Search & Reporting
New Search
index=botsv3 sourcetype="winhostmon" source=operatingsystem OS="Microsoft Windows 10 Enterprise"
30 events (before 12/8/25 2:35:25.000 AM)
Events (30)
Patterns
Statistics
Visualization
Time
Event
Type=OperatingSystem
OS="Microsoft Windows 10 Enterprise"
Architecture="64-bit"
Version="10.0.17134"
BuildNumber="17134"
Show all 22 lines
host = BSTOLL-L source = operatingsystem sourcetype = WinHostMon


Dec 8 02:36
Search | Splunk 10.0.2
splunk>enterprise
Search & Reporting
New Search
index=botsv3 host="BSTOLL-L"
240,882 events (before 12/8/25 2:35:48.000 AM)
Events (240,882)
Patterns
Statistics
Visualization
Time
Event
host = BSTOLL-L source = PerfmonMk:Process sourcetype = PerfmonMk:Process



I started by filtering for sourcetype=”winhostmon” using the provided hint. I then searched through the filters to find a filter for “operatingsystem” and an additional filter for OS=”Microsoft Windows 10 Enterprise”. This operating system only had 30 uses, which all came from one user – BSTOLL-L. Once I had this user, I then removed my operatingsystem and OS filters, and instead added a filter for host=”BSTOLL-L”. I searched the filters list again for anything relating to FQDN, however I could not find anything. I instead searched the additional filters for anything relating to “name”, where I found the filter ComputerName which when inspected had the value of BSTOLL-L.froth.ly.

Conclusion:

This report has demonstrated the real-world application of SOC tiers when analysing the BOTSv3 dataset within Splunk. I have mainly applied tier 2 of SOC, however, have also applied other incident handling methodologies to investigate and identify security incidents within the dataset. Overall, this report has highlighted the importance of real-world SOC operations in identifying and analysing security incidents.

References:

BOTSv3 GitHub download: https://github.com/splunk/botsv3

ChatGPT: https://chatgpt.com/

Splunk Enterprise download: https://www.splunk.com/en_us/download/splunk-enterprise.html

SOC roles explanation: https://www.paloaltonetworks.com/cyberpedia/soc-roles-and-responsibilities#:~:text=A%20security%20operations%20center%2C%20or,security%20solutions%2C%20tools%20and%20products.

NIST incident handling methodology: https://www.nist.gov/cyberframework/getting-started/online-learning/five-functions

AI Declaration:

Solo Work	**S1 - Generative AI tools have not been used for this assessment.**	☐
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.	☐
	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.	☐
	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.	☐
	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.	☐
	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.	☐
	A6 – Code Review 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.	☐
	A7 - Code Generation for Learning Purposes 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.	☐
	A8 - Technical Guidance & Debugging Support 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.	☐
	A9 - Testing and Validation Support 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.	☐
	A10 - Data Analysis and Visualization Guidance 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.	
	A11 - Other uses not listed above Please specify:	☐
Partnered Work	**P1 - Generative AI tool usage has been used integrally for this assessment** Students can adopt approaches that are compliant with instructions in the assessment brief. Please Specify: - Locating reports - Summarising reports - Fixing grammar - More precise language	☒

Please provide details of AI usage and which elements of the coursework this relates to:
Partnered Work – Used to locate and summarise reports for SOC and incident handling, Used to make language more concise and in proper English, including fixing grammar

I understand that the ownership and responsibility for the academic integrity of this submitted assessment falls with me, the student.	☒
I confirm that all details provide above are an accurate description of how AI was used for this assessment.	☒