

BOTSv3 Security Report: Frothly Brewing Company Attack Investigation

Your Name: Tang Pak Chun

Date: February 2025

GitHub Link: <https://github.com/Patrick-cybersec/COMP3010HK-Security-Operations-Incident-Management/edit/main/comp3010/README.md>

Video Presentation: <https://www.youtube.com/watch?v=cexqLYRQPZM>

1. Introduction

BOTSv3 is a free training dataset created by Splunk. It simulates a real cyber attack on a fake company called **Frothly**, which makes beer. The company has offices, computers, servers, email, and uses Amazon AWS cloud services.

The dataset contains many different kinds of logs:

- Windows computer logs
- Linux server logs
- Network traffic
- Email messages
- AWS CloudTrail (who did what in the cloud)
- AWS S3 access logs (who read or uploaded files to storage)

In this assignment, I worked on the **200-level questions**. These questions focus mostly on problems in the AWS cloud and one question about Windows computers. My main goals were:

- Find out which people (IAM users) were using AWS
- Check if they used extra security (MFA) when logging in
- Discover when someone accidentally made a storage bucket public
- See what secret text file got uploaded after that mistake

- Find which computer was running a different version of Windows compared to others

What I included in this report: Only data from the BOTSV3 dataset. I used Splunk on my own computer. All events are from August 2018 (the time range in the dataset).

This exercise is very useful because it shows exactly how real Security Operations Center (SOC) analysts work every day: reading logs, asking questions, and finding problems before attackers cause big damage.

2. SOC Teams and How to Handle Security Incidents

A Security Operations Center (SOC) is like a control room that watches for cyber attacks 24/7. It has three main levels of people:

- **Tier 1 analysts:** They look at many alerts every day. They decide if something is real or just noise. They do the first quick check.
- **Tier 2 analysts:** They do deeper work. They search logs, connect different clues, and find out what really happened.
- **Tier 3 analysts:** They are experts. They hunt for hidden threats, write new detection rules, and help fix very difficult problems.

In the BOTSV3 scenario:

- Tier 1 might see alerts about strange AWS changes or many failed logins.
- Tier 2 would use Splunk to search for the exact event (like PutBucketAcl) and find user “bstoll” did it.
- Tier 3 would look at the whole picture and suggest long-term fixes like better AWS settings or new alerts.

Most SOCs follow the **NIST incident response steps**:

1. **Preparation** — Buy good tools, train people, make playbooks
2. **Detection and Analysis** — Find bad activity fast using logs and alerts
3. **Containment** — Stop the attack from spreading (example: make bucket private again)
4. **Eradication** — Remove the problem completely
5. **Recovery** — Bring systems back to normal

6. Post-Incident — Write a report and improve everything

From BOTSV3 I learned important lessons:

- Many security problems start because normal employees make mistakes (like forgetting to secure a bucket).
- If logs are missing or hard to search, it takes much longer to find the fact.
- Companies should use automatic tools (such as SOAR) to react faster and reduce human work.
- Prevention is better than reaction — strong rules like force MFA and never allow public buckets can stop most accidents.

3. How I Installed Splunk and Added the Data

I installed Splunk directly on my Windows computer (no virtual machine needed). This made setup faster and simpler for me.

Step-by-step what I did:

1. Downloaded the free Splunk Enterprise installer from the official Splunk website.
2. Ran the installer and followed the wizard (chose default options).
3. Started Splunk and changed the default password for safety.
4. Opened the Splunk web page in my browser.
5. Downloaded the BOTSV3 dataset zip file from GitHub.
6. Unzipped the file.
7. Used Splunk's "Add Data" feature or copied the extracted files into the correct Splunk apps folder (following the GitHub instructions).
8. Restarted Splunk if needed.
9. Tested with a simple search: `index=botsv3 | stats count by sourcetype` — it showed many sourcetypes like aws:cloudtrail, winhostmon, etc.

Why I chose this setup:

- It is quick and easy — no need to set up a separate virtual machine.
- Works well on my personal computer without extra RAM or disk space.

- Still gives full access to all BOTSV3 data and Splunk features.
- In a real small team or student lab, many people use Windows for simplicity.

Screenshots:

- Splunk welcome screen after login

The screenshot shows the Splunk Enterprise application dashboard. The left sidebar lists several applications: Search & Reporting, Audit Trail, Discover Splunk Observability Cloud, Splunk Secure Gateway, and Upgrade Readiness App. The main content area is titled "哈囉, Administrator" and displays a search bar and a table with columns for Type, Application, Name, Owner, Last Updated, Application, and Shared. A message at the bottom states "您尚未建立知識物件." (You have not created any knowledge objects yet.) and a link to "Create a 儀表板吧" (Create a dashboard).

- List of all sourcetypes found

The screenshot shows a search results page in Splunk. The search query is "index=botsv3 | stats count by sourcetype". The results table has columns for sourcetype and count. The data is as follows:

sourcetype	count
access_combined	3967
alternatives	4
amazon-ssm-agent	965
amazon-ssm-agent-too_small	1408
apache_error	63
aws:cloudtrail	6571
aws:cloudwatch	4936
aws:cloudwatch:guardduty	1
aws:cloudwatchlogs	115145
aws:cloudwatchlogs:vpcflow	97448
aws:config:rule	255
aws:description	522
aws:elb:accesslogs	2051
aws:rds:audit	35192

- Simple search result showing data is working

i	時間	事件
>	19/09/20 2:10:50.000	<error><message>Splunkd daemon is not responding: ('Error connecting to /services/shcluster/config/config: [Errno 111] Connection refused.'){</message></error> host = OD-FM-NA-1-0xd2d665d4bdace22.amazonaws.com source = aws_content_importer sourcetype = aws_content_importer
>	18/08/20 23:27:09.296	[{ bytes: 50637 bytes_in: 30 bytes_out: 50607 dest_ip: 172.16.0.178 dest_mac: 02:A0:38:1B:3:70 endtime: 2018-08-20T15:27:09.296Z flow_id: c240fdab-997d-4285-a221-1080976a6b8a fragment_count: 0 packets_in: 1 packets_out: 36 protocol: UDP protocol_stack: ip:udp:unknown protoid: 17 src_ip: 11.125.33.118 }

4. Answers to the 200-level Questions

Question 1: Which IAM users used AWS services?

Answer: bstoll,btun,splunk_access,web_admin

Search used: `index=botsv3 sourcetype=aws:cloudtrail | stats count by userIdentity.userName | sort userIdentity.userName`

```
> 18/08/20      { [-]
  23:15:04.000
    awsRegion: us-west-1
    eventID: 51b2664e-dd61-4db3-ab77-31b...
    eventName: DescribeSecurityGroups
    eventSource: ec2.amazonaws.com
    eventTime: 2018-08-20T15:15:04Z
    eventType: AwsApiCall
    eventVersion: 1.05
    recipientAccountId: 622676721278
    requestID: 11a2f0a0-78b1-4968-8887-4...
    requestParameters: { [+]
    }
    responseElements: null
    sourceIPAddress: 107.77.212.175
    userAgent: signin.amazonaws.com
    userIdentity: { [-]
      accessKeyId: ASIAZB6TMXZ7FYCAEHN...
      accountId: 622676721278
      arn: arn:aws:iam::622676721278:use...
      invokedBy: signin.amazonaws.com
      principalId: AIDAJUFKXZ44LV4EN4MG...
      sessionContext: { [+]
      }
      type: IAMUser
      userName: bstoll
    }
  }
```

Pay attention to userName

The screenshot shows the Splunk Enterprise search interface. The top navigation bar includes the 'splunk>enterprise' logo and a '應用套件' dropdown. Below the navigation is a horizontal menu with tabs: 搜尋 (Search), 分析 (Analytics), 資料集 (Data Sets), 報告 (Reports), 警示 (Alerts), and 儀表板 (Dashboards). The '搜尋' tab is highlighted with a green underline. The main title '新搜尋' (New Search) is displayed above the search bar. The search bar contains the command: `index=botsv3 sourcetype=aws:cloudtrail | stats count by userIdentity.userName | sort userIdentity.userName`. Below the search bar, a message indicates `✓ 6,571 個事件 (10/01/01 0:00:00.000 至 20/01/01 0:00:00.000)` and a dropdown menu for '無事件取樣' (No sample events). A secondary search bar below shows the query `userIdentity.userName` with a dropdown arrow. The results section lists four users: bstoll, btun, splunk_access, and web_admin. At the bottom left, there are buttons for '顯示: 20 每頁' (Display: 20 pages), '格式' (Format) with a dropdown arrow, and a toggle switch for '預覽: 開' (Preview: On).

Why it matters: In a real company, SOC must watch who is using cloud accounts. Strange users or too many actions can be signs of hacking.

Question 2: Which field shows if someone did NOT use MFA?

Answer: `userIdentity.sessionContext.attributes.mfaAuthenticated`

The screenshot shows the Splunk Enterprise search interface. The top navigation bar includes the 'splunk>enterprise' logo and a '應用套件' dropdown. Below the navigation is a horizontal menu with tabs: 搜尋 (Search), 分析 (Analytics), 資料集 (Data Sets), 報告 (Reports), 警示 (Alerts), and 儀表板 (Dashboards). The '搜尋' tab is highlighted with a green underline. The main title '新搜尋' (New Search) is displayed above the search bar. The search bar contains the command: `index=botsv3 sourcetype=aws:cloudtrail earliest=0`.

```
> 18/08/20      { [-]
23:15:20.000    awsRegion: us-west-1
                  eventID: 97c6bfcb-c3cf-437c-8b05-4043635ce306
                  eventName: DescribeInstanceStatus
                  eventSource: ec2.amazonaws.com
                  eventTime: 2018-08-20T15:15:20Z
                  eventType: AwsApiCall
                  eventVersion: 1.05
                  recipientAccountId: 622676721278
                  requestID: f4bd4e9b-e27c-4a52-93fa-fab7a76d3639
                  requestParameters: { [+] }
                  responseElements: null
                  sourceIPAddress: autoscaling.amazonaws.com
                  userAgent: autoscaling.amazonaws.com
                  userIdentity: { [-]
                      accountId: 622676721278
                      arn: arn:aws:sts::622676721278:assumed-role/AWS
                      invokedBy: autoscaling.amazonaws.com
                      principalId: AROAIOHK7E4SHKYSVYLM:AutoScaling
                      sessionContext: { [-]
                          attributes: { [-]
                              creationDate: 2018-08-20T15:09:21Z
                              mfaAuthenticated: false
                          }
                          sessionIssuer: { [+] }
                      }
                  }
                  type: AssumedRole
              }
}
```

Pay attention at mfaAuthenticated

splunk>enterprise 應用套件 ▾

搜尋 分析 資料集 報告 警示 儀表板

新搜尋

```
index=botsv3 sourcetype=aws:cloudtrail earliest=0  
| table userIdentity.sessionContext.attributes.mfaAuthenticated
```

✓ 6,571 個事件 (10/01/01 0:00:00.000 至 20/01/01 0:00:00.000) 無事件取樣 ▾

事件 樣式 統計資料 (6,571) 視覺化

顯示: 20 每頁 ▾ 格式 ▾ 預覽: 開

userIdentity.sessionContext.attributes.mfaAuthenticated ^

userIdentity.sessionContext.attributes.mfaAuthenticated
false

MfaAuthentication: false

Why it matters: Without MFA, stolen passwords are very dangerous. SOC should make an alert for any important action done without MFA.

Question 3: What CPU model is on the web servers?

Answer: E5-2676

The screenshot shows a Splunk search interface with the following search bar query: `index=botsv3 sourcetype=hardware earliest=0`. Below the search bar, it says "3 個事件 (26/02/12 19:00:11.000 之前) 無事件取樣". The main area displays a table with three columns: "時間" (Time), "事件" (Event), and "欄位" (Fields). The first event is timestamped at 18/08/20 22:26:25.000. The event details are as follows:

欄位	KEY	VALUE
	CPU_TYPE	Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz
	CPU_CACHE	30720 KB
	CPU_COUNT	2
	HARD_DRIVES	xvda 8 GB;

Below the table, there are buttons for "顯示全部 9 行" (Show all 9 rows), "source = hardware", and "sourcetype = hardware". On the left sidebar, under "所選欄位", there are entries: host 3, source 1, and sourcetype 1. Under "關注欄位", there is an entry: index 1.

Why it matters: Knowing normal hardware helps SOC notice strange things, for example if CPU usage suddenly goes very high because of malware.

Question 4: What is the Event ID that made the S3 bucket public?

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

The screenshot shows a Splunk search interface with the following search bar query: `index=botsv3 sourcetype=aws:cloudtrail eventName=PutBucketAcl earliest=0`. The main area displays a table with three columns: "時間" (Time), "事件" (Event), and "欄位" (Fields). The first event is timestamped at 18/08/20 22:26:25.000. The event details are as follows:

欄位	KEY	VALUE
	eventName	PutBucketAcl

Below the table, there are buttons for "source = aws:cloudtrail" and "sourcetype = aws:cloudtrail". On the left sidebar, under "所選欄位", there are entries: host 3, source 1, and sourcetype 1. Under "關注欄位", there is an entry: index 1.

```

{ [-]
  Grantee: { [-]
    URI: http://acs.amazonaws.com/groups/global/AllUsers
    xmlns:xsi: http://www.w3.org/2001/XMLSchema-instance
    xsi:type: Group
  }
  Permission: READ
}
{ [-]
  Grantee: { [-]
    URI: http://acs.amazonaws.com/groups/global/AllUsers
    xmlns:xsi: http://www.w3.org/2001/XMLSchema-instance
    xsi:type: Group
  }
  Permission: WRITE
}
]
}

```

Pay attention at /AllUsers

Why it matters: This ID proves exactly when and how the security mistake happened.

Question 5: What is Bud's username?

Answer: bstoll

The screenshot shows a Splunk search interface with the following details:

- Search Query:** index=botsv3 sourcetype=aws:cloudtrail eventName=PutBucketAcl earliest=0 | table _time eventID userIdentity.userName requestParameters.bucketName requestParameters.AccessControlPolicy
- Results:** 2 個事件 (26/02/10 22:41:52.000 之前) 無事件取樣
- Event 1:** _time: 2018/08/20 21:57:54, eventID: 9a33d8df-1e16-4d58-b36d-8e80ce68f8a3, userIdentity.userName: bstoll, requestParameters.bucketName: frothlywebcode
- Event 2:** _time: 2018/08/20 21:01:46, eventID: ab45689d-69cd-41e7-8705-5350402cf7ac, userIdentity.userName: bstoll, requestParameters.bucketName: frothlywebcode

Why it matters: We can see it was probably an accident by an employee, not an outside hacker.

Question 6: What is the name of the public S3 bucket?

Answer: frothlywebcode

The screenshot shows the Splunk Enterprise search interface. The search bar contains the query: index=botsv3 sourcetype=aws:cloudtrail eventName=PutBucketAcl earliest=0 | table _time eventID userIdentity.userName requestParameters.bucketName requestParameters.AccessControlPolicy. The results section shows 2 events from 2018/08/20 at 21:57:54 and 2018/08/20 at 21:01:46. Both events show the user identity as bstoll and the bucket name as frothlywebcode.

_time	eventID	userIdentity.userName	requestParameters.bucketName
2018/08/20 21:57:54	9a33d8df-1e16-4d58-b36d-8e80ce68f8a3	bstoll	frothlywebcode
2018/08/20 21:01:46	ab45689d-69cd-41e7-8705-5350402cf7ac	bstoll	frothlywebcode

```
        }
        {
        [-]
        Grantee: [-]
            URI: http://acs.amazonaws.com/groups/global/AllUsers
            xmlns:xsi: http://www.w3.org/2001/XMLSchema-instance
            xsi:type: Group
        }
        Permission: WRITE
    }
]
}
Owner: [-]
    DisplayName: bstoll
    ID: 4c018053e740f45beb45f68c0f5eff6347745488ae540130432c9fc
}
xmlns: http://s3.amazonaws.com/doc/2006-03-01/
}
acl: [-]

]
bucketName: frothlywebcode
}
```

Pay attention at bucketName

Why it matters: Knowing the exact bucket name helps understand how much data was at risk.

Question 7: What text file was uploaded when the bucket was public?

Answer: OPEN_BUCKET_PLEASE_FIX.txt

The screenshot shows a Splunk search interface with the following details:

- Search Query:** index=botsv3 sourcetype=aws:s3:accesslogs "frothlywebcode" "PUT" "200" *.txt earliest=-
- Results:** 1 個事件 (18/08/20 21:02:44.000 至 18/08/20 21:02:44.001) 無事件取樣
- Event Details:**

時間	事件
18/08/20 21:02:44.000	4c018653e740f45beb45f68c0fSeff6347745488ae540130432c9fc64fae310d [frothlywebcode] [20/Aug/2018:13:02:44 +0000] 52.66.146.128 - DF1BA98D9E2369B4 REST PUT OBJECT OPEN_BUCKET_PLEASE_FIX.txt *PUT /OPEN_BUCKET_PLEASE_FIX.txt HTTP/1.1 200 - - 377 268 9 -.* "Botocore/1.7.62 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86_64 Botocore/1.8.12" - host = splunk.froth.ly source = s3://frothlyweblogs/s32018-07-26-01-20-56-19D73C05AA29AED8 sourcetype = aws:s3:accesslogs

Why it matters: This shows attackers (or curious people) could download secret files very easily after the mistake.

Question 8: Which computer has a different Windows version?

Answer: bstoll-l.froth.ly

index=botsv3 sourcetype=winhostmon "operatingsystem" earliest=0

✓ 204 個事件 (26/02/12 19:21:23.000 之前) 無事件取樣 ▾

事件 (204) 樣式 統計資料 視覺化

時間表格式 ▾ -縮小 +縮放至選取範圍 ×取消選擇

格式 ▾ 顯示: 20 每頁 ▾ 檢視: 清單 ▾

隱藏欄位	所有欄位	i	時間	事件
所選欄位 a host 8 a source 1 a sourcetype 1	關注欄位 a Architecture 1 # BuildNumber 1 a BuildType 1 # CodeSet 1 a ComputerName 8 # CountryCode 1 # FreePhysicalMemoryKB 100+ # FreeVirtualMemoryKB 100+ a index 1 a InstallDate 5 a LastBootUpTime 38 # linecount 1 # Locale 1 a OS 2 a punct 1 a SerialNumber 2 a splunk_server 1	>	18/08/20 23:17:23.000	Type=OperatingSystem OS="Microsoft Windows 10 Pro" Architecture="64-bit" Version="10.0.17134" BuildNumber="17134" 顯示全部 22 行 host = FYODOR-L source = operatingsystem sourcetype = WinHostMon
		>	18/08/20 23:16:28.000	Type=OperatingSystem OS="Microsoft Windows 10 Pro" Architecture="64-bit" Version="10.0.17134" BuildNumber="17134" 顯示全部 22 行 host = JWORTOS-L source = operatingsystem sourcetype = WinHostMon
		>	18/08/20 23:14:22.000	Type=OperatingSystem OS="Microsoft Windows 10 Enterprise" Architecture="64-bit" Version="10.0.17134" BuildNumber="17134" 顯示全部 22 行 host = BSTOLL-L source = operatingsystem sourcetype = WinHostMon

Pay attention at OS=" Microsoft Windows 10 Enterprise"

Other hosts using Microsoft Windows 10 Pro, but host: BSTOLL-L is uniquely using Microsoft Windows 10 Enterprise version.

The screenshot shows the Splunk Enterprise search interface. At the top, there's a search bar with the following query:

```
index=botsv3 BSTOLL-L OR bstoll-l earliest=0
| search sourcetype IN ("WinEventLog:Security", "xmlwineventlog:microsoft-windows-sysmon/Operational", "wineventlog")
| table _time host ComputerName
| search ComputerName=*.froth.ly
```

Below the search bar, it says "23,812 個事件 [26/02/10 23:20:00.000 之前]" and "無事件取樣".

The search results table has three columns: "_time", "host", and "ComputerName". The data is as follows:

_time	host	ComputerName
2018/08/20 23:17:58	BSTOLL-L	BSTOLL-L.froth.ly
2018/08/20 23:17:29	BSTOLL-L	BSTOLL-L.froth.ly
2018/08/20 23:17:29	BSTOLL-L	BSTOLL-L.froth.ly
2018/08/20 23:17:28	BSTOLL-L	BSTOLL-L.froth.ly
2018/08/20 23:17:22	BSTOLL-L	BSTOLL-L.froth.ly
2018/08/20 23:16:47	BSTOLL-L	BSTOLL-L.froth.ly

Why it matters: Different versions can be a sign of compromise, or just bad management — SOC should check both.

Overall lesson: One small human mistake can cause big data leaks. Good tools like Splunk + strong rules can catch and stop it early.

5. Conclusion and References

Summary of what I found:

- User bstoll made a storage bucket public by accident.
- After that, a sensitive text file was uploaded.
- Some AWS actions happened without MFA protection.
- One computer (bstoll-l.froth.ly) had a different Windows edition.

What companies should improve:

- Force MFA on all accounts
- Use AWS setting to block public buckets automatically
- Create alerts for dangerous events like PutBucketAcl with public access
- Keep a list of normal computer setups so strange ones are easy to notice

Simple references:

1. NIST. Computer Security Incident Handling Guide. 2012.
<https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf>
2. Splunk. BOTSV3 Dataset on GitHub. <https://github.com/splunk/botsv3>
3. AWS. Explanation of PutBucketAcl.
https://docs.aws.amazon.com/AmazonS3/latest/API/API_PutBucketAcl.html

Extra files:

- Screenshots folder with pictures from Splunk
- Video showing live searches and explanations

Gernerative AI Declaration

I applied AI when i:

- Tell my answer to ai and let it generates a report with professional and neatly format.
- troubleshoot at finding the answer of question8. I asked ai and know that the operating system field is shown as “operatingsystem” at splunk and computer name as “computername” .
- ask ai to check if i am missing component of submitting the coursework.