**Boss of the SOCv3 WriteUp**

Scenario

Fresh off the back of the BOTSv1 CTF I attempted to do this challenge. However, beyond the data they give you and instructions to load it, I couldn’t find what was the context behind the challenge. Also, I suspect the pre-requisite software was written for an earlier version of Splunk as I got all sorts of errors upon restarting my instance. The scoreboard app just wouldn’t work which meant I resorted to using a spreadsheet with the answers in one blacked out column, the attempted answer in another column and a calculation that worked out if I was correct. Bless Excel’s functional weird software.

**Question 200**

|  |  |
| --- | --- |
| Question | List out the IAM users that accessed an AWS service (successfully or unsuccessfully) in Frothly's AWS environment? Answer guidance: Comma separated without spaces, in alphabetical order. (Example: ajackson,mjones,tmiller) |
| Working | Looking at the sourcetypes we see 2 that seem specifically used for AWS. aws:s3:accesslogs and aws:cloudtrail. If you look at the accesslogs sourcetype you get a null user. Looking into cloudtrail you get fields with IAM and actual usernames. |
| Command | index=botsv3 sourcetype="aws:cloudtrail" "userIdentity.type"=IAMUser  | stats values(userName) |
| Answer | bstoll,btun,splunk\_access,web\_admin |

**Question 201**

|  |  |
| --- | --- |
| Question | Question: What field would you use to alert that AWS API activity have occurred without MFA (multi-factor authentication)? Answer guidance: Provide the full JSON path. (Example: iceCream.flavors.traditional) |
| Working | So, I first searched for AWS events on google without MFA and came across this site: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-event-reference-aws-console-sign-in-events.html  I then found the field additionalEventData.MFAUsed = No and tried this but got told it was wrong. I realised this was only for Console Sign In Events. Looking down the list of fields again I came across this field userIdentity.sessionContext.attributes.mfaAuthenticated |
| Command | index=botsv3 sourcetype="aws:cloudtrail" |
| Answer | userIdentity.sessionContext.attributes.mfaAuthenticated |

**Question 202**

|  |  |
| --- | --- |
| Question | What is the processor number used on the web servers? Answer guidance: Include any special characters/punctuation. (Example: The processor number for Intel Core i7-8650U is i7-8650U.) |
| Working | I looked into hardware originally and saw all 3 events had “E5-2676 v3”. Including the v3 was not getting me anywhere so I tried looking elsewhere. Looking at the hint tells you to specifically look at the hardware. Tried it without the v3 and its correct. Gosh I love having to be hyper exact. |
| Command | index=botsv3 sourcetype="hardware" |
| Answer | E5-2676 |

**Question 203**

|  |  |
| --- | --- |
| Question | Bud accidentally makes an S3 bucket publicly accessible. What is the event ID of the API call that enabled public access? Answer guidance: Include any special characters/punctuation. |
| Working | Going back to the cloudtrail sourcetype. Looking at the event types there’s only one that updates the ACL for an S3 bucket “PutBucketAcl” and then as part of the ACL grantee’s there is a group called <http://acs.amazonaws.com/groups/global/AllUsers> that appears to be given READ and WRITE permissions. Only one event shows up for this. |
| Command | index=botsv3 sourcetype="aws:cloudtrail" eventName="PutBucketAcl" "requestParameters.AccessControlPolicy.AccessControlList.Grant{}.Grantee.URI"="http://acs.amazonaws.com/groups/global/AllUsers" |
| Answer | ab45689d-69cd-41e7-8705-5350402cf7ac |

**Question 204**

|  |  |
| --- | --- |
| Question | What is the name of the S3 bucket that was made publicly accessible? |
| Working | In the fields we see requestParameters.bucketName |
| Command |  |
| Answer | frothlywebcode |

**Question 205**

|  |  |
| --- | --- |
| Question | What is the name of the text file that was successfully uploaded into the S3 bucket while it was publicly accessible? Answer guidance: Provide just the file name and extension, not the full path. (Example: filename.docx instead of /mylogs/web/filename.docx) |
| Working | There’s 3 hints already provided by the question. The first is the time period which is any time after the bucket was made publicly accessible:    The second is that we’re searching for the frothlywebcode bucket and the third is we’re searching for a text file. Looking at the sourcetypes shows us that 2 are available of which aws:s3:accesslogs would be the most suitable. Searching on the PUT method which is for when you’re uploading to or updating a server leads to one event and shows the text file we are looking for. |
| Command | index=botsv3 sourcetype="aws:s3:accesslogs" "frothlywebcode" "\*txt" http\_method=PUT |
| Answer | OPEN\_BUCKET\_PLEASE\_FIX.txt |

**Question 206**

|  |  |
| --- | --- |
| Question | What is the size (in megabytes) of the .tar.gz file that was successfully uploaded into the S3 bucket while it was publicly accessible? Answer guidance: Round to two decimal places without the unit of measure. Use 1024 for the byte conversion. Use a period (not a comma) as the radix character. |
| Working | Searching for “\*.tar.gz” gives us 3 results and one of which has “access\_denied”. So we narrow down the results to status=200 and there’s 2 events one of which was generated by a user-agent-string that seems to be generated by a python script:    All thats left is to convert the object\_size to MB and round to 2 decimals places |
| Command | 1. index=botsv3 sourcetype="aws:s3:accesslogs" bucket\_name=frothlywebcode http\_method=PUT http\_status=200 "\*.tar.gz" → Shows us the file 2. index=botsv3 sourcetype="aws:s3:accesslogs" bucket\_name=frothlywebcode http\_method=PUT "\*.tar.gz" status=200 http\_user\_agent="Boto3/1.7.61 Python/2.7.14 Linux/4.14.47-64.38.amzn2.x86\_64 Botocore/1.8.12"  | eval SizeInMB=ROUND((object\_size/(1024\*1024)),2) | table SizeInMB |
| Answer | 2.93 |

**Question 208**

|  |  |
| --- | --- |
| Question | A Frothly endpoint exhibits signs of coin mining activity. What is the name of the first process to reach 100 percent CPU processor utilization time from this activity on this endpoint? Answer guidance: Include any special characters/punctuation. |
| Working | Moving over to the cloudwatch sourcetype, as its the service that monitors the resource itself and not the API activity, you can find a field called metric\_name with a useful value of “CPUUtilization”. None of these events showed 100% utilization. Looking through the sourcetypes we see “performonk:process”. A web search suggests it’s used to give performance monitoring on the processes. There’s a field process\_cpu\_used\_percent where a few of them have 100%. The first result of MicrosoftEdgeCP#2 is wrong.  Looking at the hint it asks “Which browser was in use when this endpoint visited the coin mining site(s)?”. Doing a search for bitcoin shows 1 result where cloudwatch appears to have a logged a DNS request to bitcoin subdomain of brewertalk.com and this has a time 1:02am at 21/8/2018 but all of the performonk events are before this time.  Randomly going through the processes leads us to our answer. I wish I understood this a bit more. |
| Command | index=botsv3 sourcetype=perfmonmk:process process\_cpu\_used\_percent=100  | reverse |
| Answer | chrome#5 |

**Question 209**

|  |  |
| --- | --- |
| Question | When a Frothly web server EC2 instance is launched via auto scaling, it performs automated configuration tasks after the instance starts. How many packages and dependent packages are installed by the cloud initialization script? Answer guidance: Provide the number of installed packages then number of dependent packages, comma separated without spaces |
| Working |  |
| Command | index=botsv3 sourcetype="cloud-init-output" "depend\*"  | rex field=\_raw "(?<Installations>Install [\s]+[0-9]+ Packages \(\+[0-9]+ Dependent packages\))" |
| Answer | 7,13 |

**Question 210**

|  |  |
| --- | --- |
| Question | What is the short hostname of the only Frothly endpoint to actually mine Monero cryptocurrency? (Example: ahamilton instead of ahamilton.mycompany.com) |
| Working | The hints suggest finding the common browser based crypto miners and then searching for it in the DNS records. A search of google for browser based miners around 2017-2018 (when BOTSv3 was released shows Coinhive as a very popular miner which is now defunct. Searching through the DNS sourcetypes gives quite a few events but there’s only 1 src\_ip value    I tried searching for DNS records for this IP address as the destination but didn’t get anything. After searching through a few sourcetypes I found the wineventlog gives the answer in the ComputerName field. |
| Command | 1. index=botsv3 sourcetype="stream:dns" "coinhive" "query\_type{}"=A 2. index=botsv3 sourcetype="wineventlog" "192.168.247.131" |
| Answer | BSTOLL-L |

**Question 211**

|  |  |
| --- | --- |
| Question | How many cryptocurrency mining destinations are visited by Frothly endpoints? |
| Working | You can get this by counting the values in the query{} field from the previous questions DNS events. |
| Command |  |
| Answer | 6 |

**Question 212**

|  |  |
| --- | --- |
| Question | Using Splunk's event order functions, what is the first seen signature ID of the coin miner threat according to Frothly's Symantec Endpoint Protection (SEP) data? |
| Working | Looking through the sourcetypes it can be seen that there are 9 sourcetypes to do with Symantec Endpoint Protection. Doing a wildcard search for the keyword of “signature” leads to only 1 sourcetype showing up symantec:ep:security:file. Using this and reversing the order of events leads to our answer. |
| Command | 1. index=botsv3 sourcetype="symantec:ep\*" "signature" 2. index=botsv3 sourcetype="symantec:ep:security:file" signature=\* | reverse |
| Answer | 30358 |

**Question 213**

|  |  |
| --- | --- |
| Question | According to Symantec's website, what is the severity of this specific coin miner threat? |
| Working | Search google for the answer leads us to Broadcom’s website: https://www.broadcom.com/support/security-center/attacksignatures/detail?asid=30358. Apparently, Broadcom bought Symantec back in 2019. |
| Command |  |
| Answer | Medium |

**Question 214**

|  |  |
| --- | --- |
| Question | What is the short hostname of the only Frothly endpoint to show evidence of defeating the cryptocurrency threat? (Example: ahamilton instead of ahamilton.mycompany.com) |
| Working | Looking at the symantec logs from Question 212 we can see the action taken on the cryptominer was that it was blocked. And if you have a look at the Host\_Name field, we can see the user. |
| Command |  |
| Answer | BTUN-L |

**Question 215**

|  |  |
| --- | --- |
| Question | What is the FQDN of the endpoint that is running a different Windows operating system edition than the others? |
| Working | The hint mentions that we should take a look at any information from the Cisco NVM clients on the mobile workstation. A search on google leads me to believe that there should be a few sourcetypes along the lines of **cisco:nvm:\*** but there’s only cisco:asa, a firewall product, in the environment. Searching for the keyword of Cisco leads to a few events with the following information:    However. a search for AnyConnect doesn’t provide much information. After searching through quite a few sourcetypes for a good while I just search “operatingsystem” and the following is shown on the WinHostMon sourcetype.    Next, we just search for on who has which windows version and we get BSTOLL-L. Combine this with the froth.ly domain and we get out FQDN of the host.  This felt like a rather unrefined way of obtaining the OS information. I would love to find out what the expert method is. |
| Command | index=botsv3 "operatingsystem"  | stats values(ComputerName) by vendor\_product |
| Answer | BSTOLL-L.froth.ly |

**Question 216**

|  |  |
| --- | --- |
| Question | According to the Cisco NVM flow logs, for how many seconds does the endpoint generate Monero cryptocurrency? Answer guidance: Round to the nearest second without the unit of measure. |
| Working | According to the hint we should be using the cisconvmflowdata. I guess we now know how to get NVM data. It also shows that its a good idea to know what the sources are, in addition to, the sourcetypes in your data.  Searching the events for connections to the coinhive servers we see 6 events and many fields that are shortened. A search of the syslog sourcetype on google leads to this website https://community.cisco.com/t5/security-knowledge-base/cisco-network-visibility-nvm-collector/ta-p/4309825  Which shows the fields fss and fes represent the timestamps for when the network flow was initiated and completed respectively. Obtaining the minimum and maximum values for fss and fes and then calculating the difference gives the time in seconds.  NOTE: It should be noted that I initially tried to search for just “coinhive” but this gives the connection to the top level domain and gives a wrong duration answer. To get the right answer you the need the connections to just the final endpoints. |
| Command | index=botsv3 source=cisconvmflowdata "\*.coinhive"  | stats min(fss) as starting, max(fes) as ending  | eval duration=ending-starting |
| Answer | 1666 |

**Question 217**

|  |  |
| --- | --- |
| Question | What kind of Splunk visualization was in the first file attachment that Bud emails to Frothly employees to illustrate the coin miner issue? Answer guidance: Two words. (Example: choropleth map) |
| Working | Looking at the sourcetypes a good place to search would be the stream:smtp sourcetype since it deals with the mail protocol. Next to narrow down the emails sent by bud, it seems we need to regex extract the sender field. This shows some .jpg attachments    However. neither this nor the body of the email mentions the visualisation type. The hint says we need to grab the base64 of the attachment and decode it. Using cyberchef we get the image below:    And thus, we get our answer of Column Chart |
| Command | index=botsv3 sourcetype="stream:smtp"  | rex field=\_raw "(?<senderemail>[A-Za-z]+@froth\.ly)" | search senderemail="bstoll@froth.ly" | table \_time, "content{}" |
| Answer | Column Chart |

**Question 218**

|  |  |
| --- | --- |
| Question | What IAM user access key generates the most distinct errors when attempting to access IAM resources? |
| Working | I don’t actually know how to answer this question. I tried filtering on iam as eventsource and user’s type being IAMUser and then tried searching multiple different combinations of things but just kept getting the wrong answer. I even looked at this guys write-up  https://ellisstannard.medium.com/boss-of-the-socs-bots-v3-part-2-971476c4cb1  Which suggests to look at the errorMessage instead of the errorCode but I still don’t get the same answer. |
| Command |  |
| Answer |  |

**Question 219**

|  |  |
| --- | --- |
| Question | Bud accidentally commits AWS access keys to an external code repository. Shortly after, he receives a notification from AWS that the account had been compromised. What is the support case ID that Amazon opens on his behalf? |
| Working | Doing a general search for “git” leads us to a the sourcetype of “stream:smtp”, which gives us 2 results. Narrowing it down with a general search for Amazon gives us 1 email where the answer is present in the content body. |
| Command | index=botsv3 "\*git\*" "\*amazon\*" sourcetype="stream:smtp" |
| Answer | 5244329601 |

**Question 220**

|  |  |
| --- | --- |
| Question | AWS access keys consist of two parts: an access key ID (e.g., AKIAIOSFODNN7EXAMPLE) and a secret access key (e.g., wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY). What is the secret access key of the key that was leaked to the external code repository? |
| Working | In the previous question the content body has a link to the person’s github where the secret access key is store. Following this link gives the key |
| Command | index=botsv3 "\*git\*" "\*amazon\*" sourcetype="stream:smtp" |
| Answer | Bx8/gTsYC98T0oWiFhpmdROqhELPtXJSR9vFPNGk |

**Question 221**

|  |  |
| --- | --- |
| Question | Using the leaked key, the adversary makes an unauthorized attempt to create a key for a specific resource. What is the name of that resource? Answer guidance: One word. |
| Working | Initially searching using the secret part of the access key gives no results. Searching for the general access key however gives a few cloud trail results. Looking at the command field we see a 1 value that would definitely be of interest:    Filtering on this field and value and we get an event where it says the user unsuccessfully to create a nullweb\_admin. |
| Command | index=botsv3 "AKIAJOGCDXJ5NW5PXUPA" sourcetype="aws:cloudtrail" command=CreateAccessKey |
| Answer | nullweb\_admin |

**Question 222**

|  |  |
| --- | --- |
| Question | Using the leaked key, the adversary makes an unauthorized attempt to describe an account. What is the full user agent string of the application that originated the request? |
| Working | Similar to the above question if we change the command to “DescribeAccountAttributes”, the answer can be found in the userAgent field. |
| Command | index=botsv3 "AKIAJOGCDXJ5NW5PXUPA" sourcetype="aws:cloudtrail" command=DescribeAccountAttributes |
| Answer | ElasticWolf/5.1.6 |

**Question 223**

|  |  |
| --- | --- |
| Question | The adversary attempts to launch an Ubuntu cloud image as the compromised IAM user. What is the codename for that operating system version in the first attempt? Answer guidance: Two words. |
| Working | Working: Doing a search for ubuntu across the dataset provides us with 13 datasources to choose from. Looking through the top choice of package we see Ubuntu but different versions    The next one to try might be OSQuery as stream:smtp and ms:o365 don’t seem useful. Searching for information on OSQuery gives this site: https://www.uptycs.com/blog/threat-research-report-team/osquery-what-it-is-how-it-works-and-how-to-use-it –which says that OSQuery standardises information collection on operating systems.  There’s a field called columns.build\_distro which contains 1 value “xenial”. Searching for xenial on google leads to an Ubuntu distribution called Xenial Xerus. Apparently Xerus is a type of African ground squirrel which makes sense considering Ubuntu is usually known for naming their distributions with the naming convention of [Adjective] [Animal]. |
| Command | index=botsv3 "\*ubuntu\*" sourcetype="osquery:results" |
| Answer | xenial xerus |

**Question 224**

|  |  |
| --- | --- |
| Question | Frothly uses Amazon Route 53 for their DNS web service. What is the average length of the distinct third-level subdomains in the queries to brewertalk.com? Answer guidance: Round to two decimal places. (Example: The third-level subdomain for my.example.company.com is example.) |
| Working | Similar to question 216 with the NVM logs being in the source. If we have a look for DNS by Amazon sourcetypes we find the following results with lambda:dns being an option we can    Searching through the events we get one-liners that don’t have the site visited automatically extracted for us in the field. Using the rex command, we extract the third level domain and deduplicate them. Next, we get the length of the field and find the average length, rounded to 2 decimal places. It should be noted that this question was mildly annoying as the answer was rounded without the zero, so I was attempting 8.10 instead of 8.1 and was getting told I was wrong each time until I tried it without the zero. |
| Command | 1. | tstats count WHERE index=botsv3 AND sourcetype="aws:\*" by source 2. index=botsv3 sourcetype="aws:cloudwatchlogs" | rex field=\_raw "(?<thirdlevel>[\$\-\_+!\*\'(),A-Za-z0-9]+)\.brewertalk\.com"  | dedup thirdlevel | eval ThirdLevelLength = len(thirdlevel)  | stats avg(ThirdLevelLength) as AVGLENGTH | eval AVGLENGTH=ROUND(AVGLENGTH,2) |
| Answer | 8.10 |

**Question 225**

|  |  |
| --- | --- |
| Question | Using the payload data found in the memcached attack, what is the name of the .jpeg file that is used by Taedonggang to deface other brewery websites? Answer guidance: Include the file extension. |
| Working | To be honest I’m not entirely sure what’s going on with this question and will attempt later. |
| Command |  |
| Answer |  |