**Splunk BOTSv1 Write Up**

**Scenario 1 – Web Site Defacement**

**Question 101**

|  |  |
| --- | --- |
| Question | What is the likely IPv4 address of someone from the Po1s0n1vy group scanning imreallynotbatman.com for web application vulnerabilities? |
| Working | Looking into the stream:http sourcetype and then finding the highest count of communications between the source and destination IP addresses we find the likely source IP of the attacker |
| Command Used | sourcetype="stream:http" | stats count by src\_ip, dest\_ip | sort -count |
| Answer | 40.80.148.42 |

**Question 102:**

|  |  |
| --- | --- |
| Question | What is the vulnerability scanner company? |
| Working | If you have a look at the some of the events where the source IP address is set to the probable attacker in the previous question you can see in the src\_headers there’s mention of an Acunetix Web Vulnerability Scanner – Free Edition |
| Command Used | sourcetype="stream:http" src\_ip="40.80.148.42" |
| Answer | Acunetix |

**Question 103:**

|  |  |
| --- | --- |
| Question | What content management system is imreallynotbatman.com likely using? |
| Working | Much like the previous question, if you have a look at some of the source headers you can see mention of Joomla. Plugging this into google shows the first result as a CMS company |
| Command Used |  |
| Answer | Joomla |

**Question 104:**

|  |  |
| --- | --- |
| Question | What is the name of the file that defaced the imreallynotbatman.com website? Please submit only the name of the file with extension? |
| Working | The hints says to look at the stream:http, fgt\_utm and suricata and that the web server is the source.  If we have look at stream:http and use the webserver as the source. There are 2 events where it is sending data to 23.22.63.114 and the file being sent is poisonivy-is-coming-for-you-batman.jpeg |
| Command Used | sourcetype="stream:http" src\_ip="192.168.250.70" dest\_ip="23.22.63.114" |
| Answer | poisonivy-is-coming-for-you-batman.jpeg |

**Question 105:**

|  |  |
| --- | --- |
| Question | This attack used dynamic DNS to resolve to the malicious IP. What fully qualified domain name (FQDN) is associated with this attack? |
| Working | This was partially answered by the last question where we can see the site that is hosting the .jpeg file that is defacing the site. |
| Command Used |  |
| Answer | prankglassinebracket.jumpingcrab.com |

**Question 106:**

|  |  |
| --- | --- |
| Question | What IPv4 address has Po1s0n1vy tied to domains that are pre-staged to attack Wayne Enterprises? |
| Working | The previous question asks you to find the FQDN of the site as the attack uses dynamic DNS. In answering this question we probably have to look through the events tied to the “http:dns” sourcetype. Doing a general string search for the domain listed in the previous question, shows you 3 events. Of those 3 there’s one with a reply code of “No Error” which means the request completed successfully and we can see our answer in the host\_addr field |
| Command Used | sourcetype="stream:dns" "prankglassinebracket.jumpingcrab.com" "reply\_code{}"=NoError |
| Answer | 23.22.63.114 |

**Question 108**

|  |  |
| --- | --- |
| Question | What IPv4 address is likely attempting a brute force password attack against imreallynotbatman.com? |
| Working | This was revealed in question 101. It’s the second highest count of packets to the server |
| Command Used |  |
| Answer | 23.22.63.114 |

**Question 109**

|  |  |
| --- | --- |
| Question | What is the name of the executable uploaded by Po1s0n1vy? |
| Working | The first hint mentions that we want to look at HTTP POST method and the second hint mentions looking specifically for executables. Searching on the POST method using both of the attackers IP addresses comes up with a lot of results that were either to do with the initial vulnerability scan or the brute force password attack and thus we should filter them out. Additionally making sure we search for “\*.exe” will give us one event and looking in the raw data we can find out file highlighted |
| Command Used | sourcetype="stream:http" (src\_ip="23.22.63.114" OR src\_ip="40.80.148.42") http\_method=POST NOT ("passwd" OR "Acunetix") "exe" |
| Answer | 3791.exe |

**Question 110**

|  |  |
| --- | --- |
| Question | What is the MD5 hash of the executable uploaded? |
| Working | The hint mentions we should go look at the Sysmon sourcetype which is XmlWinEventLog:Microsoft-Windows-Sysmon/Operational. Searching for “3791.exe” leads us to 69 events and both hash fields listing more than 50 MD5s. If you set the ImageLoaded field to the 3791.exe then this reduces to one. |
| Command Used | sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" ImageLoaded="C:\\inetpub\\wwwroot\\joomla\\3791.exe" |
| Answer | AAE3F5A29935E6ABCC2C2754D12A9AF0 |

**Question 111**

|  |  |
| --- | --- |
| Question | Using research techniques, provide the SHA256 hash of this malware |
| Working | To be honest I couldn’t figure this out even when I tried to use the IP address that the hints suggested because virustotal and any other malware stats site like abuseipdb, hybrid-analysis, criminalip.io pretty unanimously require sign ups to get anything out of them. I then caved in and searched for other write ups in Splunk and found this one: <https://www.cyberdonald.com/post/boss-of-the-soc-splunk> which provided the answer. However, I later realised I was an idiot and didn’t fully look at all the tabs that Virus Total. Looking at the Relations tab shows some files provided by the server with one of the files being “MirandaTateScreensaver.scr.exe” which also leads to the answer |
| Command Used | sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" ImageLoaded="C:\\inetpub\\wwwroot\\joomla\\3791.exe" |
| Answer | 9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8 |

**Question 112**

|  |  |
| --- | --- |
| Question | What special hex code is associated with the customized malware discussed in question 111? |
| Working | Did the same thing as the previous question and just looked at the previous write up. |
| Command Used | sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" ImageLoaded="C:\\inetpub\\wwwroot\\joomla\\3791.exe" |
| Answer | 53 74 65 76 65 20 42 72 61 6e 74 27 73 20 42 65 61 72 64 20 69 73 20 61 20 70 6f 77 65 72 66 75 6c 20 74 68 69 6e 67 2e 20 46 69 6e 64 20 74 68 69 73 20 6d 65 73 73 61 67 65 20 61 6e 64 20 61 73 6b 20 68 69 6d 20 74 6f 20 62 75 79 20 79 6f 75 20 61 20 62 65 65 72 21 21 21 |

**Question 114**

|  |  |
| --- | --- |
| Question | What was the first brute force password used? |
| Working | Splunk sorts events so the latest event is first. Also all form\_data events with a password have the text “passwd”, knowing this I used the below command |
| Command Used | sourcetype="stream:http" src\_ip="23.22.63.114" passwd | sort +\_time | table \_time, form\_data | where isnotnull(form\_data) |
| Answer | 12345678 |

**Question 115**

|  |  |
| --- | --- |
| Question | One of the passwords in the brute force attack is James Brodsky's favorite Coldplay song. We are looking for a six-character word on this one. Which is it? |
| Working | With this question we need to extract the passwords that are used in the form\_data field using the rex command. We then need to get the length of the passwords and filter on passwords with a length of six characters.    Next, I got ChatGPT to give me a list of all Coldplay songs of 6 characters. I then added an additional condition to the where command with guessed\_pwd={input}. In tidying up this writeup I tried to go back and get the AI to provide the answers for me again and it provided me with the output below:    In My Place? Paradise? For anyone else reading this, just parse the list of songs using PowerShell or Python. It’s faster and you don’t need to “Prompt Engineer” them. |
| Command Used | sourcetype="stream:http" src\_ip="23.22.63.114" passwd | rex field=form\_data "passwd=(?<guessed\_pwd>[^&]+)"| eval pwd\_length=len(guessed\_pwd) | table form\_data, guessed\_pwd, pwd\_length | where isnotnull(form\_data) and pwd\_length=6 |
| Answer | Yellow |

**Question 116**

|  |  |
| --- | --- |
| Question | What was the correct password for admin access to the content management system running "imreallynotbatman.com"? |
| Working | The hint mentions that there are more than one IP address submitting passwords. Modifying the command in the previous question to get rid of the src\_ip requirement passwords, if you have a look at the stats for the extracted field you can see that batman was used twice. |
| Command Used | sourcetype="stream:http" passwd | rex field=form\_data "passwd=(?<guessed\_pwd>[^&]+)"| eval pwd\_length=len(guessed\_pwd) | where isnotnull(guessed\_pwd) and pwd\_length=6 |
| Answer | batman |

**Question 117**

|  |  |
| --- | --- |
| Question | What was the average password length used in the password brute forcing attempt? |
| Working | Modify the previous question to get rid of the password length requirement and use the stats avg command |
| Command Used | sourcetype="stream:http" passwd | rex field=form\_data "passwd=(?<guessed\_pwd>[^&]+)"| eval pwd\_length=len(guessed\_pwd) | table src\_ip, form\_data, guessed\_pwd, pwd\_length | where isnotnull(form\_data) | stats avg(pwd\_length) |
| Answer | 6 |

**Question 118**

|  |  |
| --- | --- |
| Question | How many seconds elapsed between the time the brute force password scan identified the correct password and the compromised login? |
| Working | I was originally going to subtract the times but then looked at the hint and it suggests using the transaction command. The transaction command is great because it gives an additional field called duration which will give the length of time between the first and last event in a transaction. Filtering to the two events where passwd=batman and you have your duration |
| Command Used | sourcetype="stream:http" "passwd=batman" | transaction dest\_ip | eval roundedDuration=round(duration,2) | table roundedDuration |
| Answer | 92.17 |

**Question 119**

|  |  |
| --- | --- |
| Question | How many unique passwords were attempted in the brute force attempt? |
| Working | Just need to use a distinct count on the passwords that were used |
| Command Used | sourcetype="stream:http" passwd | rex field=form\_data "passwd=(?<guessed\_pwd>[^&]+)"| table src\_ip, form\_data, guessed\_pwd | where isnotnull(form\_data) | stats dc(guessed\_pwd) |
| Answer | 412 |

**Scenario 2 - Ransomware**

**Question 200**

|  |  |
| --- | --- |
| Question | What was the most likely IPv4 address of we8105desk on 24AUG2016? |
| Working | Searching for we8105desk leads us to a qualified name of we8105desk.waynecorpinc.local. Next I looked at the sourcetypes and used Sysmon since it was the most used (looking back the “stream:dns” would probably have been more useful) and then looked at all of the src\_ip’s being used. This gives you 4 values.    Of these 0.0.0.0 is a null route, 127.0.0.1 is the localhost and the last address is the IPv6 address. |
| Command Used | index=botsv1 sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" src\_host="we8105desk.waynecorpinc.local" | stats values(src\_ip) |
| Answer | 192.168.250.100 |

**Question 201**

|  |  |
| --- | --- |
| Question | Amongst the Suricata signatures that detected the Cerber malware, which one alerted the fewest number of times? Submit ONLY the signature ID value as the answer. |
| Working | Searching for the compromised workstation’s IP address in the suricata sourcetype shows 4 events and a field called alert.signature\_id    In order to narrow this down I added “cerber” to the search which filters outs 2020716 which provides the answer. |
| Command Used | index=botsv1 sourcetype="suricata" src\_ip="192.168.250.100" "cerber" event\_type=alert | stats count by alert.signature\_id |
| Answer | 2816763 |

**Question 202**

|  |  |
| --- | --- |
| Question | What fully qualified domain name (FQDN) does the Cerber ransomware attempt to direct the user to at the end of its encryption phase? |
| Working | The first hint suggests we look at the stream:dns sourcetype for A queries coming from the infected workstation and to narrow the search period.  To narrow down the search period I got the time of the last event from the previous question which was:    Using a search period between 17:15 and 17:16 and then searching the stream:dns sourcetype gives us 6 events. If we look at the query field we are given 3 values, one of which contains the string “cerber” |
| Command Used | 1. index=botsv1 sourcetype="suricata" event\_type=alert src\_ip="192.168.250.100" | stats latest(\_time) as LastTime | eval LastTime=strftime(LastTime,"%Y/%m/%d %H:%M:%S") 2. index=botsv1 sourcetype="stream:dns" src\_ip="192.168.250.100" |
| Answer | cerberhhyed5frqa.xmfir0.win |

**Question 203**

|  |  |
| --- | --- |
| Question | What was the first suspicious domain visited by we8105desk on 24AUG2016? |
| Working | Searching the DNS sourcetype for A queries and the infected workstation during the date in question gives us quite a few domains. Searching through each of them leads to solidaritedproximate being alerted as very suspicious. |
| Command Used | index=botsv1 sourcetype="stream:dns" src\_ip="192.168.250.100" "query\_type{}"=A | table \_time, query{} | sort +\_time |
| Answer | solidaritedeproximite.org |

**Question 204**

|  |  |
| --- | --- |
| Question | During the initial Cerber infection a VB script is run. The entire script from this execution, pre-pended by the name of the launching .exe, can be found in a field in Splunk. What is the length of the value of this field? |
| Working | The first hint suggests to look through Sysmon data. Searching through Sysmon events for the visual basic scripts gives us 7 events and if we have a look at the command line field gives 1 event where there is a huge command that looks like it’s being obfuscated. |
| Command Used | index="botsv1" sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" Computer="we8105desk.waynecorpinc.local" "vbs" | eval LengthOfCmd=len(CommandLine) | table CommandLine, LengthOfCmd |
| Answer | 4490 |

**Question 205**

|  |  |
| --- | --- |
| Question | What is the name of the USB key inserted by Bob Smith? |
| Working | Searching google on how to identify USB evens in registry lead me to these very useful links:   * <https://community.splunk.com/t5/Getting-Data-In/How-do-I-receive-events-whenever-someone-plugs-unplugs-a-USB/m-p/38240> * <https://lantern.splunk.com/Security/UCE/Foundational_Visibility/Incident_management/Investigating_a_ransomware_attack/Removable_devices_connected_to_a_machine>   Which suggest to look for registry entries mentioning the USBSTOR and grabbing the friendly name |
| Command Used | index="botsv1" sourcetype="winregistry" host="we8105desk" "USBSTOR" friendlyname |
| Answer | MIRANDA\_PRI |

**Question 206**

|  |  |
| --- | --- |
| Question | Bob Smith's workstation (we8105desk) was connected to a file server during the ransomware outbreak. What is the IPv4 address of the file server? |
| Working | The hint suggests looking at the SMB traffic from the infected device. Using this information and finding the destination with the highest count leads us to our answer |
| Command Used | index="botsv1" sourcetype="stream:smb" src\_ip="192.168.250.100" | stats count by dest\_ip |
| Answer | 192.168.250.20 |

**Question 207**

|  |  |
| --- | --- |
| Question | How many distinct PDFs did the ransomware encrypt on the remote file server? |
| Working | The hint says that the windows events logs are the place to look at and in order to search through the event logs we need to find the name of the server which can do by searching the DNS requests.    We then search all the event logs and narrow it down by pdfs that the malware masquerading as Bob accessed as there’s one event with the user of Administrator.    Finally distinct count the PDFs |
| Command Used | 1. index="botsv1" sourcetype="suricata" "dns.rdata"="192.168.250.20" | stats values(dns.rrname) 2. index="botsv1" sourcetype="wineventlog\*" ComputerName="we9041srv.waynecorpinc.local" "\*.pdf" user="bob.smith" | stats dc(Relative\_Target\_Name) as "Number of PDFs" |
| Answer | 257 |

**Question 208**

|  |  |
| --- | --- |
| Question | The VBscript found in question 204 launches 121214.tmp. What is the ParentProcessId of this initial launch? |
| Working | The hint suggests to use the Sysmon sourcetype and that you should use the process\_id, ParentProcessId, CommandLine and ParentCommandLine.  Initially I wasn’t thinking clearly since I just set process = 121214.tmp and tried the parentID of this but was promptly told I was wrong. Having a look at the question again I realised that the question is looking for the very first initial process.    So 121214.tmp was launched by cmd.exe which in turn was launched by 20429.vbs but looking at cmd.exe’s ParentProcessID gives us the ID that we are looking for |
| Command Used | index="botsv1" sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" Computer="we8105desk.waynecorpinc.local" "121214.tmp" | table process, process\_id, ParentProcessId, CommandLine, ParentCommandLine | reverse |
| Answer | 3968 |

**Question 209**

|  |  |
| --- | --- |
| Question | The Cerber ransomware encrypts files located in Bob Smith's Windows profile. How many .txt files does it encrypt? |
| Working | The hints suggest to look at the Sysmon sourcetype, focus on the user’s profile, use EventCode=2 and that we should look out for duplicates. Initially tried to solve this by counting the file\_names only to table the file\_path and file\_name and realised the hint is warning about duplicates as the file name is the same across many different folders. |
| Command Used | index="botsv1" sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" file\_name="\*.txt" EventCode=2 file\_path="C:\\Users\\bob.smith\*" | stats dc(file\_path) |
| Answer | 406 |

**Question 210**

|  |  |
| --- | --- |
| Question | The malware downloads a file that contains the Cerber ransomware cryptor code. What is the name of that file? |
| Working | The hint suggests to start with the suricata sourcetype but I found that this didn’t really help. I went randomly through a few of the indicators compromise until I finally got some data which seemed useful from using the solidaritedeproximite site which was the first suspicious domain the user’s computer visited. It downloads a jpg which the fgt\_utm claims is malware. |
| Command Used | index="botsv1" "mhtr.jpg" sourcetype=fgt\_utm |
| Answer | mhtr.jpg |

**Question 211**

|  |  |
| --- | --- |
| Question | Now that you know the name of the ransomware's encryptor file, what obfuscation technique does it likely use? |
| Working | Steganography is a common technique used with images to hide data |
| Command Used | N/A |
| Answer | Steganography |