Alexei Hooks

Cyber 440

Professor Moeller

October 12, 2022

Individual Analytic Lab – Windows Security Log Analysis

Bottom Line Up Front:

Analyzing and reviewing the log file "SecurityLog-rev2.xml" proved very informational in my learning process. I was able to use different operation systems (Kali Linus and Windows 10) to analyze two separate identical files and understand an all-encompassing view of the data. From what I found, I was able to lay out the simpler details of logs and identify possible indicators of compromise that are located within the dataset. Through means of locating and coding, I was able to better analyze the data and review it within this report. Using the terminal in Kali, I was able to better my skills in the command line along with ease of parsing through large data files. I was also able to create a frequency chart about the most successful log on events from one user. Throughout the rest of this report you will be able to find out important information regarding the data from "SecurityLog-rev2.xml."

Data Overview:

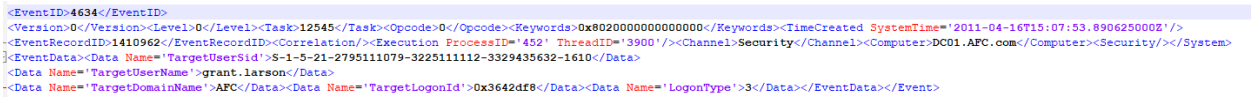
Stage One:

Part A:

When was the first event?

First Event – Event ID = 4634 – An account was logged off

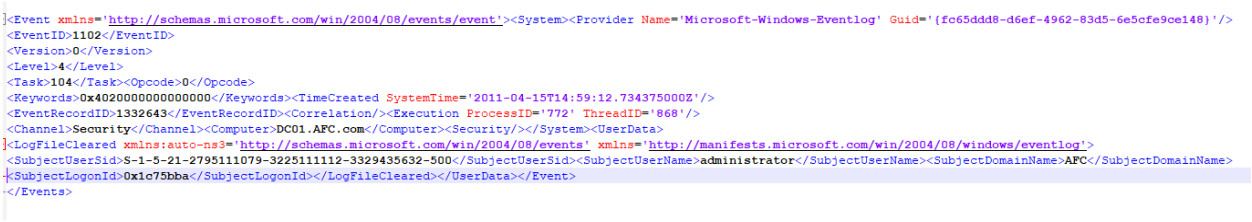
The account was Grant Larson



<EventID>4634</EventID>
<Version>0</Version><Level>0</Level><Task>12545</Task><Opcode>0</Opcode><Keywords>0x8020000000000000</Keywords><TimeCreated SystemTime='2011-04-16T15:07:53.890625000Z'/>
<EventRecordID>1410962</EventRecordID><Correlation/><Execution ProcessID='452' ThreadID='3900'/><Channel>Security</Channel><Computer>DC01.AFC.com</Computer><Security/></System>
<EventData><Data Name='TargetUserSid'>S-1-5-21-2795111079-3225111112-3329435632-1610</Data>
<Data Name='TargetUserName'>grant.larson</Data>
<Data Name='TargetDomainName'>AFC</Data><Data Name='TargetLogonId'>0x3642df8</Data><Data Name='LogonType'>3</Data></EventData></Event>

When was the last event?

Last Event – Event ID = 1102 – The audit log was cleared

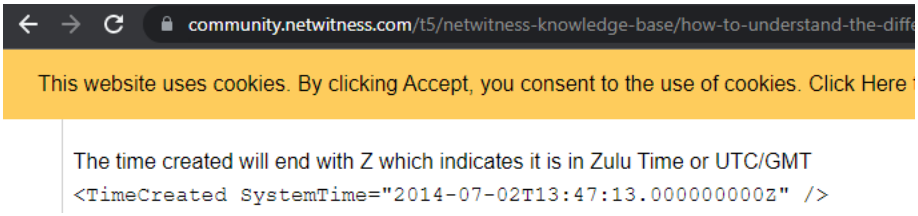


<Event xmlns='http://schemas.microsoft.com/win/2004/08/events/event'><System><Provider Name='Microsoft-Windows-Eventlog' Guid='{fc65ddd8-d6ef-4962-83d5-6e5cfe9ce148}'/>
<EventID>1102</EventID>
<Version>0</Version>
<Level>4</Level>
<Task>104</Task><Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords><TimeCreated SystemTime='2011-04-15T14:59:12.734375000Z'/>
<EventRecordID>1332643</EventRecordID><Correlation/><Execution ProcessID='772' ThreadID='868'/>
<Channel>Security</Channel><Computer>DC01.AFC.com</Computer><Security/></System><UserData>
<LogFileCleared xmlns:auto-ns3='http://schemas.microsoft.com/win/2004/08/events' xmlns='http://manifests.microsoft.com/win/2004/08/windows/eventlog'>
<SubjectUserSid>S-1-5-21-2795111079-3225111112-3329435632-500</SubjectUserSid><SubjectUserName>administrator</SubjectUserName><SubjectDomainName>AFC</SubjectDomainName>
<SubjectLogonId>0x1c75bba</SubjectLogonId></LogFileCleared></UserData></Event>
</Events>

What time zone are the time stamps in?

When analyzing the events, each has a time stamp in the variable SystemTime. The time entries are in Zulu time, meaning that it is using UTC/GMT time zones. I know this as each time stamp is followed with a Z, indicating Zulu time. An example is linked below:

<TimeCreated SystemTime='2011-04-16T15:07:53.890625000Z'/>



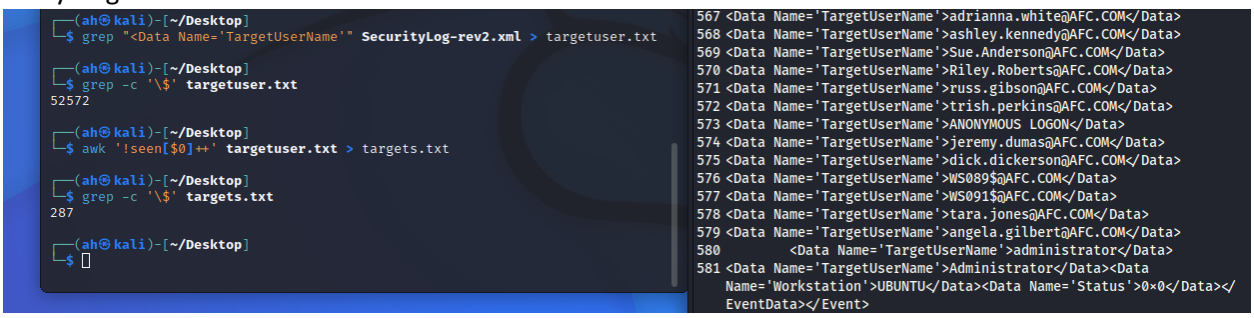
How many total events reported?

There are a total of 78,320 events within the file. I did this by looking at the count for “EventID” which was 156,640. As “EventID” shows up twice for each event, I divided 156,640 by 2 to gain the total number of events.

Part B:

How many different users (as opposed to computers) log on to the network?

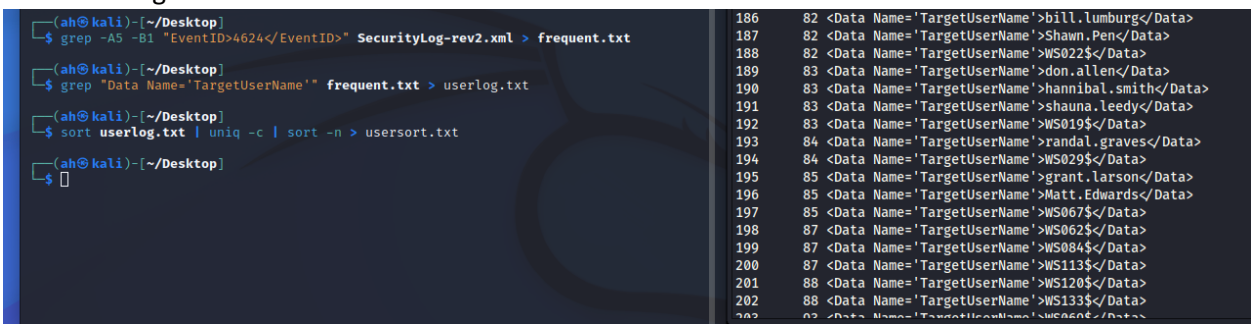
The amount of users within the security logs is 294. I found this by finding the total amount of users including the computers, then searched for how many $ were included in the list. That number came out to 287. As the total list of users and computers was 581. Subtract the amount of computers from this list and you get 294 users.



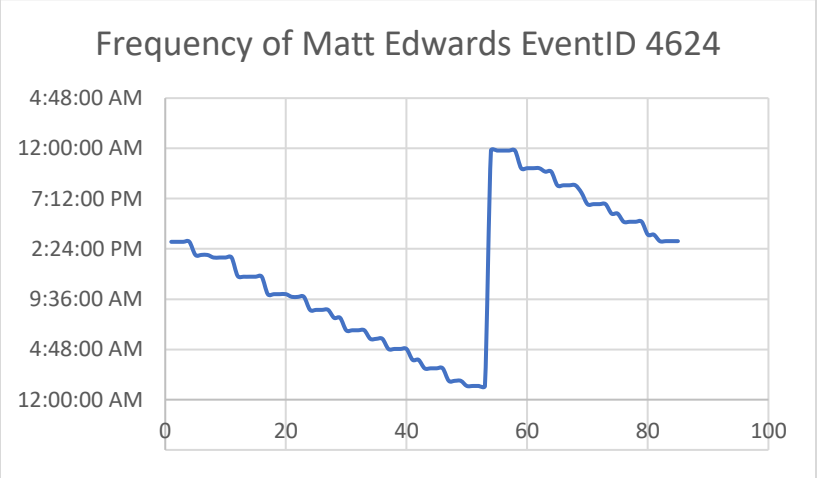
Which user log on more times than other users?

I reworked the console to only output the EventID of 4624, being that it means there was a successful log in. I then searched for the user names in that file, and sorted the file into a secondary file after that. Then I was able to look at how many times each user had been seen successfully logging on. It seemed as if there were only 316 unique usernames that successfully logged on. Of these, the most popular was a computer “DC01$.” The most successful log on attempts was done by “Matt.Edwards” with 85

successful log ins.



Make a frequency chart of when this user logs in.



I made this table by isolating the dates and times of EventID 4624 triggering by hand of Matt Edwards, and organizing it into a scatterplot in excel.

Part C:

How many times is the EventID 4625 reported?

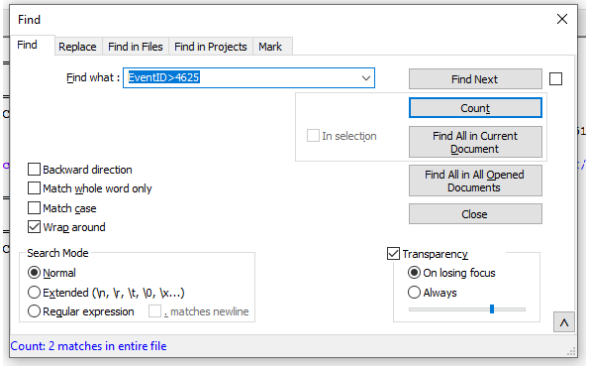
The event ID “4625 – An account failed to Log on” occurred twice throughout the entire file. This event happened separately at:

<TimeCreated SystemTime='2011-04-15T15:05:01.875000000Z'/>

<TimeCreated SystemTime='2011-04-15T15:04:57.671875000Z'/>

```
<Event xmlns='http://schemas.microsoft.com/win/2004/08/events/event'>
<System>
    <Provider Name='Microsoft-Windows-Security-Auditing' Guid='{54849625-5478-4994-A5BA-3E3B0328C30D}'/>
    <EventID>4625</EventID>
    <Version>0</Version>
    <Level>0</Level>
    <Task>12544</Task>
    <Opcode>0</Opcode>
    <Keywords>0x8010000000000000</Keywords>
    <TimeCreated SystemTime='2011-04-15T15:04:57.671875000Z'/>
    <EventRecordID>1332958</EventRecordID>
    <Correlation/>
    <Execution ProcessID='452' ThreadID='3224'/>
    <Channel>Security</Channel>
    <Computer>DC01.AFC.com</Computer>
    <Security/>
</System>
<EventData>
```

```
<Event xmlns='http://schemas.microsoft.com/win/2004/08/events/event'>
<System>
    <Provider Name='Microsoft-Windows-Security-Auditing' Guid='{54849625-5478-4994-A5BA-3E3B0328C30D}'/>
    <EventID>4625</EventID>
    <Version>0</Version>
    <Level>0</Level>
    <Task>12544</Task>
    <Opcode>0</Opcode>
    <Keywords>0x8010000000000000</Keywords>
    <TimeCreated SystemTime='2011-04-15T15:05:01.875000000Z'/>
    <EventRecordID>1332961</EventRecordID>
    <Correlation/>
    <Execution ProcessID='452' ThreadID='3224'/>
    <Channel>Security</Channel>
    <Computer>DC01.AFC.com</Computer>
    <Security/>
</System>
```



Describe each of these events focusing on the user accounts and computers that were involved.

Being that each of the two events are describing a user or computer failing to log on, the situation should be taken with a decent bit of caution. As the channel of each event is labeled security, I believe that this could be in line for the security team to make sure this is legit activity. The source of this login failure is “DC01.AFC.com” which could be a company domain that is being targeted by malicious actors.

What do you think should be done to solve this issue?

I believe that the only real thing to be done here is to follow up on where the source is coming from, and try to understand who might be trying to access the domain. From there, I don’t believe any modifications should be made to the security logs, as it reported the activity as it should have.

Part D:

Report which software tools you used.

I used two different virtual machines to complete my analysis of the windows security log file “SecurityLog-rev2.xml.” The first was a Windows 10 VM, which allowed me to access Notepad++ to

analyze the entire .xml file. The second was a Kali Linux VM which I used to parse through the file in the terminal emulator. Both software tools allowed me a good amount of control over the data, which gave me a better idea how to manipulate code to more easily find pertinent information within the file.

Identify the methods you used to find the information. Report and functions, scripts or semi-automated methods you applied in the tools.

When using Notepad++, I mostly used the control + F key to access the find mechanism. This solution allowed me to quickly search and find different keywords that I was looking for, along with an option called "count" that allowed to see how many times a keyword was used in the file.

In the Kali Linux terminal, I was able to use the commands of grep, awk and sort to better sift through data from the .xml file. The ability to quickly queue data that was important to find was a great advantage, as it saved time along with the ability to write the sought after data to a new text file. This allowed me to better analyze and report my findings. I also used excel to create the frequency chart for Matt Edwards Event ID 4624 to show how many successful login attempts were made by him.

Specific scripts I used are examples such as:

grep "Data Name= 'TargetUserName'" frequent.txt > frequentuser.txt

- Allowed for me to write data from one file to the other, with the goal of having a list only regarding target user names. This allows me to better analyze the data I wanted to find.

grep -A5 -B1 "<EventID>4624</EventID> SecurityLog-rev2.xml > frequent.txt

- The idea of this line allowed me to see additional info about the EventID 4624 – successful log on. The -A5 allowed for more lines to be included for each found match, giving me more context about each event. The -B1 allowed for me to have the number of lines printed before the output. This allows for easy counts.

sort frequentuser.txt | uniq -c | sort -n

- This command allowed me to sort the file frequencyuser.txt to be able to see how many times each unique variable was used, along with being sorted in the most popular of these variables. This was helpful to see which users logged on the most.

Stage Two:

Part A:

Your program should read the file as input and write an output file.

grep SecurityLog-rev2.xml > partA.txt

Part B:

Modify your program to only duplicate lines into your output file that are associated with a specified EventID.

grep "<EventID>4624</EventID>" SecurityLog-rev2.xml > partB.txt

Part C:

Modify your program to generate a count of the number of times that the event ID occurs. Run your program and generate and output file for Event ID 4625.

grep -c "<EventID>4624</EventID>" SecurityLog-rev2.xml > partC.txt

Part D:

Modify your program to report the number of times that the given event ID occurs over time. Your program should report the number of times each EventID occurs during each hour.

grep -A6 -B1 "<EventID>4624</EventID>" SecurityLog-rev2.xml > partD.txt

grep 'TimeCreated SystemTime' partD.txt > partDi.txt

sort partDi.txt | uniq -c | sort -n > partDii.txt

Indicators of Compromise

Within the data file "SecurityLog-rev2.xml" there are possible IOCs that reside hidden to the untrained eye. I was able to locate an amount of three separate known IOCs that are included on sites such as: Important Windows Event IDs: Which Events You Should Monitor and Why | BeyondTrust

The four indicators of compromise I found existed in the system were EventIDs: 1102, 4625, 4672.

1102: "Audit Log was cleared"

- Occurred 1 time.
- This IOC is actually the last event in the file, and is commonly used for intruder in the system to cover their tracks, meaning they are able to delete logs that exist within the dataset before they are able to be seen, so no history of the attack was able to be seen.

4625: "An account failed to log on"

- Occurred 2 times.
- With any system, a possible foreign logon attempt should be taken with a caution. In the case of the two EventIDs that included 4625, the threat trying to access the system was "DCO1.AFC.com" and was labeled into a security collection. Further analysis would have to be done to see if the threat was real or benign.

4672: "Special privileges assigned to new logon"

- Occurred 4361 times.
- When looking through logs, it needs to be known who has special rights, and who does not. With EventID 4672, there is a possibility that the wrong user gets granted admin rights, allowing them access to company data they should not have. This could also be from an outside threat, granting themselves admin access that could lead to the whole company's data being targeted.