IFS4102 LAB
WEEK 6

NO LAB TASKS FOR THIS WEEK!

REGISTER GROUPINGS BY 19 FEB

OBJECTIVES

1. Hide, execute, discover, delete files in NTFS **(Task 1-1)**
2. Inspect Windows Event Log File **(Task 2)**
3. Analyze prefetch files **(Task 3)**
4. Analyzing Windows Shortcut Files **(Task 4)**
5. (Optional) Analyzing Jump List Files **(Task 5)**
6. (Optional) Analyzing Thumbnail Caches **(Task 6)**

1. HIDE, EXECUTE, DISCOVER, DELETE FILES IN A NTFS DISK **(TASK 1-1&2)**

- To **hide** file in an existing file, use command line.
  - Command: `type malware.exe > file.txt:malware.exe`
  - `Can hide more than 1 stream`

- To **run** executables or hidden files in ADS
  - For normal files, use appropriate application to run
    - Command: `mspaint file.txt:secret.jpg`
  - For executables, use powershell
    - Command: `.\file.txt:hw.exe`

1. HIDE, EXECUTE, DISCOVER, DELETE FILES IN A NTFS DISK **(TASK 1-1&2)**

- To **discover** ADS,
  - Using Cmdline: `dir/R`
  - Using sysinternal streams: `streams <file>`
  - Using powershell: `Get-Item –path <file> -stream *`

- To **delete** streams
  - Using powershell: `Remove-Item –path <path> -stream <name>`
  - Using sysinternal streams: `streams –d <file>`
    - Sysinternal streams remove all streams

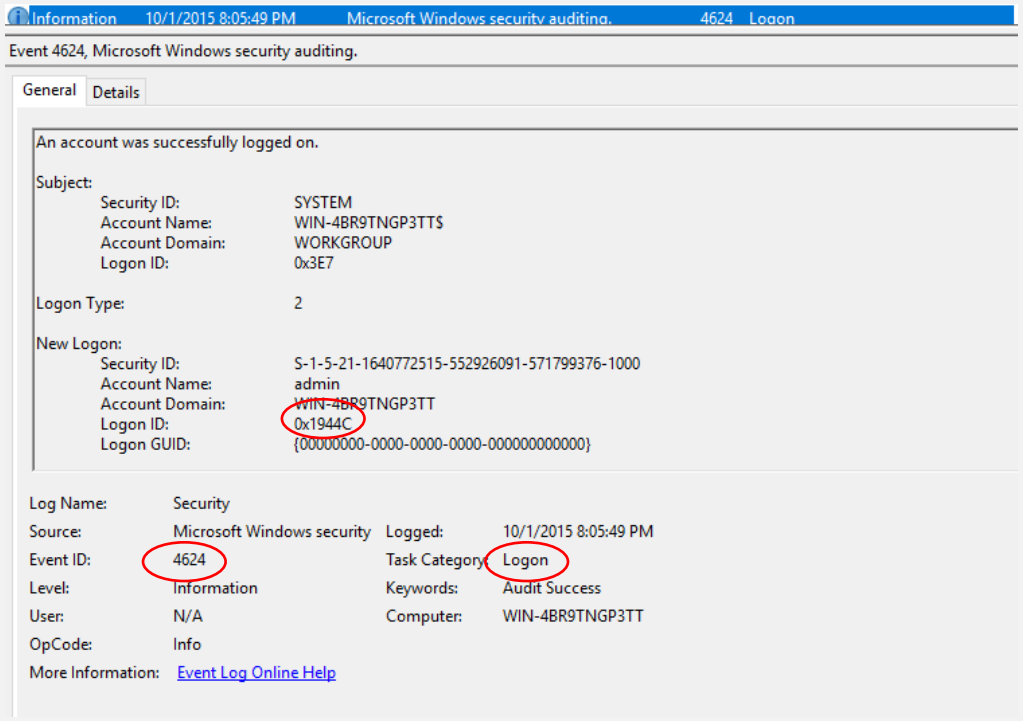
2. INSPECT WINDOWS EVENT LOG FILE **(TASK 2)**

- Use sample file
- Security Log records events related to security such as logon attempts and resource access.
- Event ID correspond to a particular event. Too many to list.
  - https://www.xplg.com/windows-server-security-events-list/
  - In this lab, we focus on Event ID 4624 and 4634/4647

2. INSPECT WINDOWS EVENT LOG FILE **(TASK 2)**

- Event ID 4624 and 4634/4647 (you will see and use this mainly)
  - 4624 = Logon
  - 4634/4647 = Logoff
  - Both events may be correlated with each other using LOGON ID values in the log
- Scenario: Suspect claims he did not use the computer. Nobody touched it.
  - Can find matching logon and logoff time, create a timeline?
  - Can use other information such as registry analysis showing only got **one** user account and **password protected**
  - **The evidence contradicts his statement**

2. INSPECT WINDOWS EVENT LOG FILE **(TASK 2)**



Information 10/1/2015 8:05:49 PM Microsoft Windows security auditing. 4624 Logon

Event 4624, Microsoft Windows security auditing.

General Details

An account was successfully logged on.

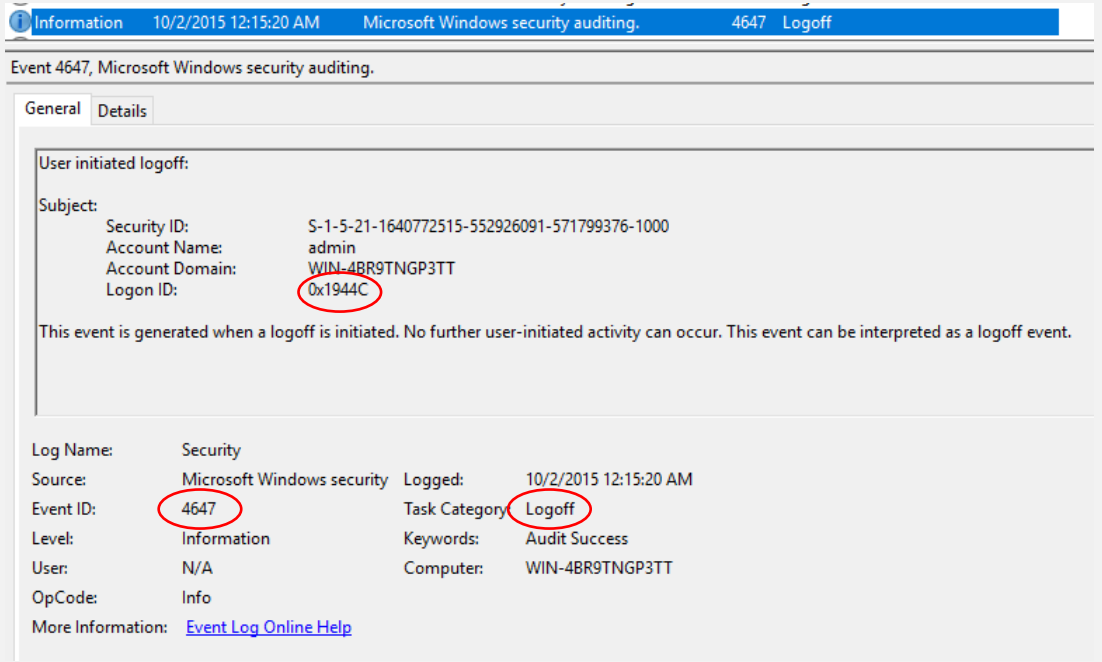
Subject:
- Security ID: SYSTEM
- Account Name: WIN-4BR9TNGP3TT$
- Account Domain: WORKGROUP
- Logon ID: 0x3E7

Logon Type: 2

New Logon:
- Security ID: S-1-5-21-1640772515-552926091-571799376-1000
- Account Name: admin
- Account Domain: WIN-4BR9TNGP3TT
- Logon ID: 0x1944C
- Logon GUID: {00000000-0000-0000-0000-000000000000}

Log Name:	Security		
Source:	Microsoft Windows security	Logged:	10/1/2015 8:05:49 PM
Event ID:	4624	Task Category:	Logon
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	WIN-4BR9TNGP3TT
OpCode:	Info		

More Information: Event Log Online Help



Information 10/2/2015 12:15:20 AM Microsoft Windows security auditing. 4647 Logoff

Event 4647, Microsoft Windows security auditing.

General Details

User initiated logoff:

Subject:
- Security ID: S-1-5-21-1640772515-552926091-571799376-1000
- Account Name: admin
- Account Domain: WIN-4BR9TNGP3TT
- Logon ID: 0x1944C

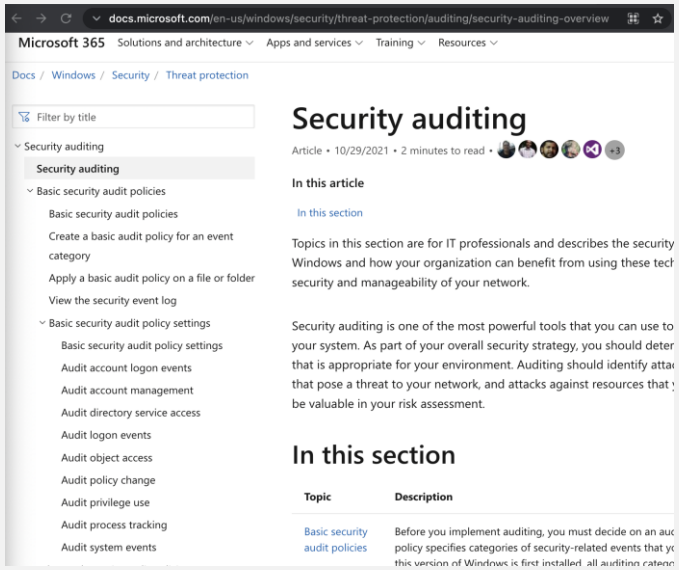
This event is generated when a logoff is initiated. No further user-initiated activity can occur. This event can be interpreted as a logoff event.

Log Name:	Security		
Source:	Microsoft Windows security	Logged:	10/2/2015 12:15:20 AM
Event ID:	4647	Task Category:	Logoff
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	WIN-4BR9TNGP3TT
OpCode:	Info		

More Information: Event Log Online Help

2. INSPECT WINDOWS EVENT LOG FILE **(TASK 2)**

- https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/security-auditing-overview
- https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/default.aspx



3. ANALYZE PREFETCH FILES (**TASK 3**)

- A Prefetch file is created when you open an application on windows system
- Contain information such as:
  - Executable name
  - Hash of executable path
  - Exact Path of Executable
  - Created/Modified Time of file
  - Run count
  - Last run time
  - Files referenced by the executable
- Usage? Prove that a certain executable was run to cover up tracks. Or show evidence of execution, even though program is deleted.

4. ANALYZING WINDOWS SHORTCUT FILES **(TASK 4)**

- .LNK files (labels or windows shortcut files) created automatically by Windows OS whenever a user open their files.
- OS use these files for quick access to certain files
- Can also be created by users themselves to make activities easier.
- Contain important information such as:
  - Source path
  - Time tags (Created, modified, access time)
  - File size
  - S/N of volume
  - MAC address

5. ANALYZING JUMP LIST FILES **(TASK 5)**

- **Similar to shortcuts, contain information about recently access applications and files**
- Demo

6. ANALYZING THUMBNAIL CACHES **(TASK 6)**

- Open a folder that contains pictures, what do you see?
- These small pictures or thumbnails are stored in a special file called thumbnail cache database
- Used mostly to establish whether or not an image file existed on the computer at some point.
- Demo using thumbcache_viewer (not sure why WFA not working for me?)
  - http://thumbcacheviewer.github.io/

QUESTIONS?

MUCH MORE POWERFUL FILTERING IN EVENT VIEWER

```
<QueryList>
  <Query Id="0" Path="file://C:\Users\IEUser\Downloads\Security.evtx">
    <Select Path="file://C:\Users\IEUser\Downloads\Security.evtx">
        *[ System [ ( EventID=4624 ) ] ]
        and
        *[ EventData [ Data [ @Name='LogonType' ] and ( Data = 2 ) ] ]
    </Select>
  </Query>
</QueryList>
```

Selection Criteria

The above XML filtering will select only logs that are event ID=4624 and have LogonType=2.

https://techcommunity.microsoft.com/t5/ask-the-directory-services-team/advanced-xml-filtering-in-the-windows-event-viewer/ba-p/399761

MUCH MORE POWERFUL FILTERING IN EVENT VIEWER

```
<QueryList>
  <Query Id="0" Path="file://C:\Users\IEUser\Downloads\Security.evtx">
    <Select Path="file://C:\Users\IEUser\Downloads\Security.evtx">
        *[ System [ ( EventID=4624 or EventID=4647 ) ] ]
        and
        *[ EventData [ Data [ @Name='TargetLogonId' ] and ( Data = '0x1be71' ) ] ]
    </Select>
  </Query>
</QueryList>
```

Selection Criteria

The above XML filtering find logon and logoff events that have the same logon id of 0x1be71

https://techcommunity.microsoft.com/t5/ask-the-directory-services-team/advanced-xml-filtering-in-the-windows-event-viewer/ba-p/399761

EVENT VIEWER'S WOES

- Troublesome, manual creation of timeline
- XML filtering helps but can be confusing.
- Log2timeline automatic generate forensic timelines (I think we will cover this in a future lab, lab 8?)
- But feel free to try it out https://plaso.readthedocs.io/en/latest/sources/user/Users-Guide.html