**Capture the Flag (CTF)**

Challenge on Cloud Forensics • Instructor • Challenge 1

**Index**

**Assumptions/Needs** 3

**Objective** 3

**Problem** 3

**Question 1** 4

**Question 2** 5

**Question 3** 7

**Objective**

The basis of this challenge will be that the user will be viewing timestamps using **FTK Imager** and **debugfs**. The user will also transfer files using **“pscp”** and use debugfs to see file timestamps on an Ubuntu Linux Machine.

**Problem**

This CTF challenge is replicating a scenario in which a malicious attacker has remotely modified private and publicly stored files on the user’s machine.

**Assumptions/Needs**

* Hyper-V and/or VirtualBox software for virtual machines
* Microsoft OneDrive free account (https://onedrive.live.com/about/en-us/)
* Windows and Ubuntu Linux machine
* FTK Imager software (<http://marketing.accessdata.com/ftkimager3.4.2>) and debugfs enabled in Linux.

**Question 1**

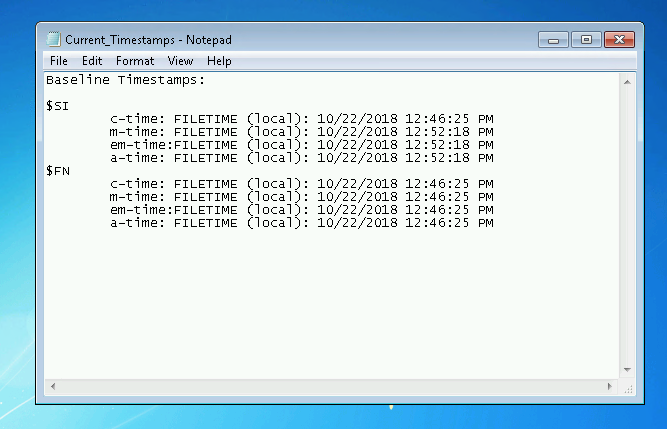
Was the modify-time unchanged? Or was it updated? Please upload a screenshot.

**Description**

The first stage will be a simple check of file timestamps using "Properties".

**Evidence**

The user will simply compare the file’s “Properties” timestamps (by right clicking and using Properties) with a set of given baseline timestamps. The baseline timestamps should have times that are earlier than the actual timestamps of the file (this way the user will see that “someone” has modified the file).



**Analysis**

The user will notice that the tampered file (which had been opened and modified by hackers) has a more recent “Modify time” when looking in the file’s properties.

**Expected Answer**

The participant should upload a screenshot of the **modify-time** as shown in “File Properties”. The **modify-time** should be new/updated.

**Question 2**

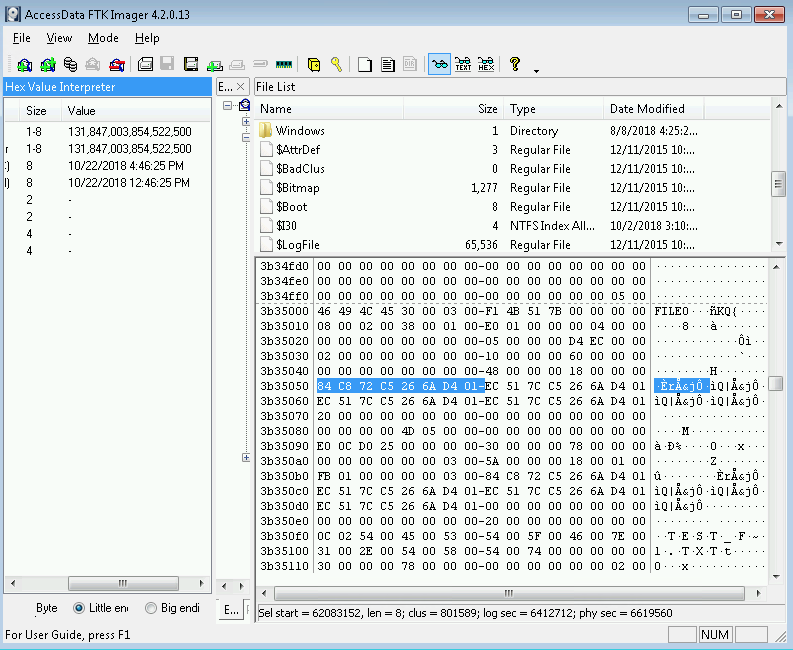
Was the **entry-modify** time unchanged? Or was it updated? Please upload a screenshot.

**Description**

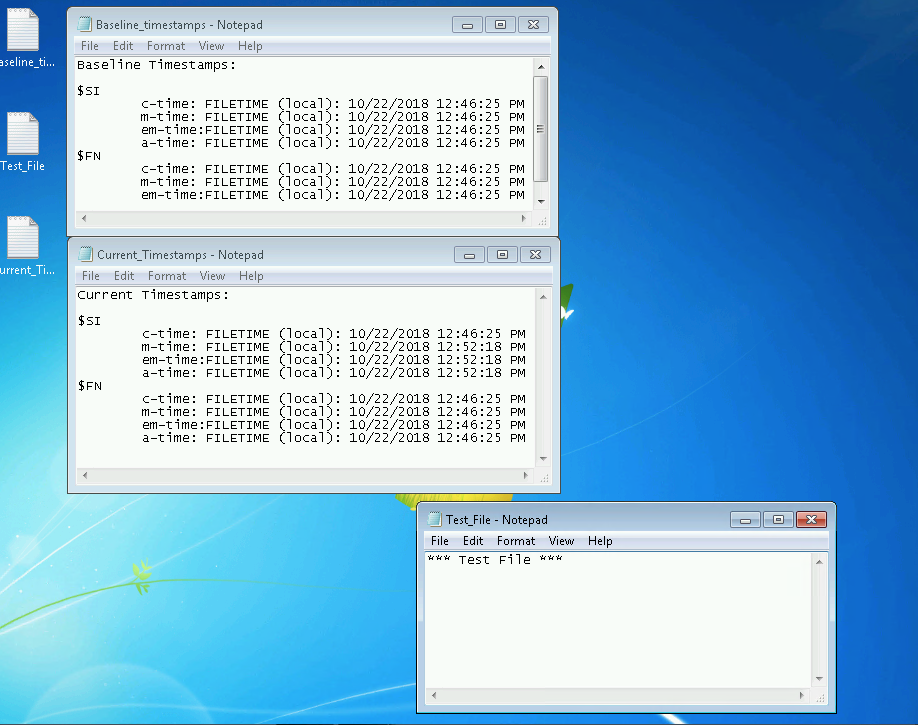
The second stage will utilize FTK Imager to see the $SI and $FN timestamps and to compare them with baseline timestamps.

**Evidence**

The user will use FTK Imager to uncover the 8 timestamps contained within $MFT.



Here are the recorded differences in the timestamps.



**Analysis**

After viewing the timestamps in the $MFT using FTK Imager, the user will see that not only was the modify time changed but so was the entry-modify time. The user has now truly verified that the file has been modified by an outside source (attacker).

**Expected Answer**

The participant should upload a screenshot of the entry-modify time (em-time) as shown in FTK Imager. The em-time should be newer than the baseline timestamps (it should have been updated). The participant needs to upload the $SI em-time NOT the $FN em-time.

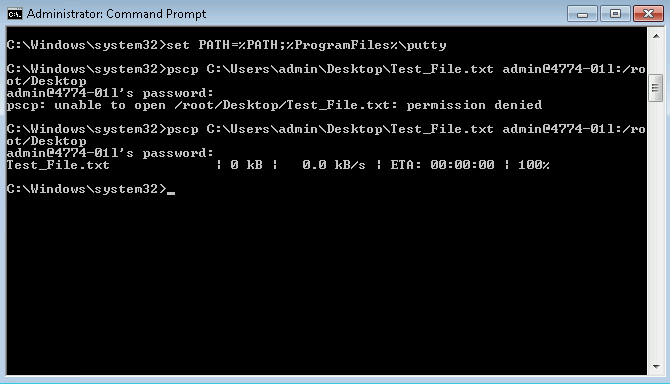
**Question 3**

What are the new timestamps for the newly transferred file? Did they remain unchanged or stay the same. Please upload a screenshot.

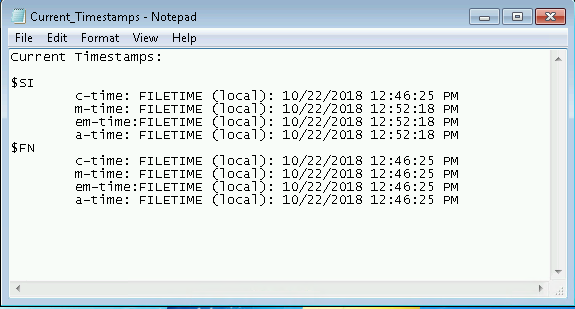
**Description**

The third stage will be transferring a file to a Linux machine to compare the timestamps and to determine if timestamps change if a hacker were to steal a file from the user’s Windows machine (by pscp-ing it to their Linux machine).

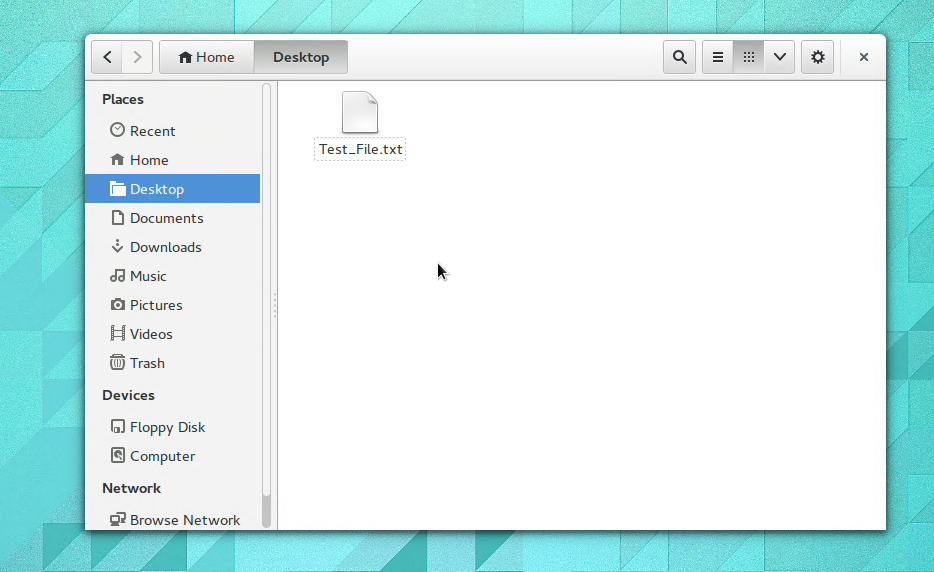
**Evidence**

The user will pscp the Windows file to a Linux machine.

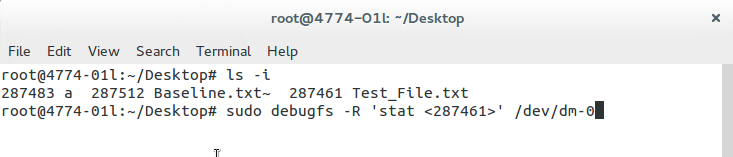
The user will check the transferred file’s timestamps in the Linux machine and compare them with the original timestamps (below).



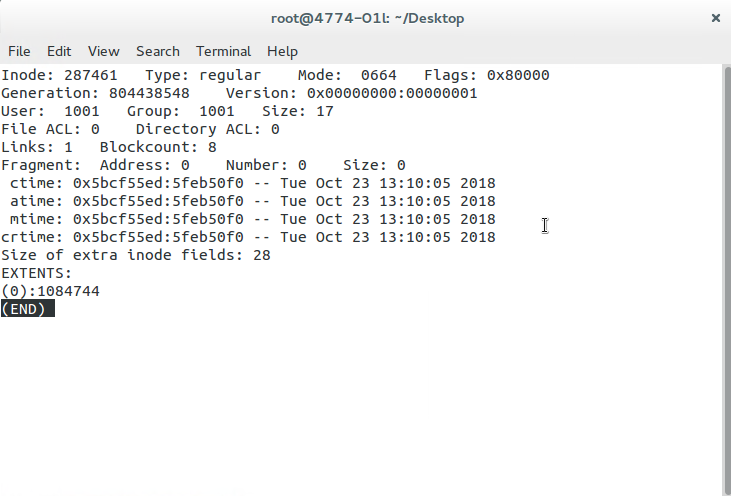
Here the file has been successfully transferred using pscp.



Using debugfs in the Linux terminal, the user can view the new timestamps for the transferred file.



Here are the results:



**Analysis**

As can be seen above, the user will see that when a file is transferred from their Windows machine to a hacker’s Linux machine, the timestamps change and become the exact time and date that the file arrived on the Linux machine.

Overall, what the user/participant in this challenge should take away is that timestamps are very important tools that can be used to see and/or verify that an outside source has accessed their files. This challenge teaches the user how to inspect file timestamps using debugfs, FTK Imager, and pscp.

**Expected Answer**

The participant should upload a screenshot of the debugfs output for the file on the Ubuntu Linux machine. The ctime, atime, mtime, and crtime should all be the same and should be newer (updated) than the baseline timestamps.