**Capture the Flag (CTF)**

Challenge on IoT Forensics • Instructor • Challenge 2

**Index**

**Assumptions/Needs** 3

**Objective** 3

**Problem** 3

**Task 1** 4

**Task 2** 5

**Task 3** 6

**Objective**

The user’s objective will be to use a filesystem debugger tool to read a baseline timestamp on a few files. After establishing the baseline timestamps, a malicious actor will make modifications to some of the files. The user will have to determine which files were affected, when the attack took place, and what was done.

**Problem**

You suspect that you have experienced a breach. Your job is to determine the extent of the breach and when the breach occurred.

**Assumptions/Needs**

* Hyper-V software for virtual Rasperry Pi
* Raspbian Stretch OS
* debugfs enabled in Linux.

**Task 1**

**Description**

The first task is to create 3 (three) files (eg.: **file1.txt**, **file2.txt**, and **file3.txt**).

2 (two) files with the ‘**nano’** and ‘**echo’** commands and collect baseline timestamps, and 1 (one) file with the **‘cat’** command. You can assume that these timestamps were taken before the breach. Identify the **inode** and **timestamps** of each file and upload screenshots of the **inode** number**, timestamps** and the metadata viewer.

**Expected Answer**

Create time (c-time) = Access time (a-time) = Modify time (m-time) = Change time (c-time) when a file is created using ‘nano’ or ‘echo’ commands. But if a file is created using ‘cat’ command, it updates the create time and access time to the time of creation whereas it updates the modify time (m-time) and change time (c-time) to the time of modification across Raspberry Pi and Ubuntu 18.04.

**Task 2**

**Description**

The baseline timestamps for a system you are in charge of are given. Since then, however, the system has been breached. In an effort to cover his tracks, the attacker made new files by **compressing** his files with the **‘tar’** and **‘gzip’** commands and by moving them into your folder under the same original names . Identify the **inode** and **timestamps** of each file and upload screenshots of the **inode** number**, timestamps** and the metadata viewer. Write a brief analysis of your findings.

**Expected Answer:**

When a file is compressed using ‘tar ’ command, all timestamps are changed to the time of compression in both Raspberry Pi and Ubuntu 18.04 system.

When a file is compressed using ‘gzip ’ command, the change time (c-time) and the create time (cr-time) are changed to the compression time, whereas the access time (a-time) and modify time (m-time) are not changed in both Raspberry Pi and Ubuntu 18.04 system.

**Task 3**

**Description**

You discovered which file the attacker deleted and replaced, but you still suspect that the attacker has seen a file that he shouldn’t have, even if no changes were made. For this scenario, let us assume you as the attacker. Open one of the files with the **‘cat’** commands, and close it without any modification. Now, as an analyst, provide screenshots for evidence of how you can conclude that a file has been viewed but not modified.

**Expected Answer:**

When a file is accessed using both GUI and the ‘cat’ command, there is no change in the timestamps of the file in Raspberry Pi system, but the access time (a-time) of the file is updated to the latest time of access in Ubuntu 18.04 system.