

# IR Timeline Analysis

-

Walkthrough of the challenge room

Paweł Murdzek, Piotr Szewczyk

Warsaw University of Technology

January 26, 2026

## Contents

<b>Introduction</b>	2
<b>1. Lab Description</b>	2
<b>2. Timelines with Log2Timeline</b>	2
2.1. Section Description	2
2.2. Analysis and Answers	2
<b>3. Timeline Analysis with Timesketch</b>	2
<b>4. Timeline Analysis Practical</b>	2
4.1. Conclusions	3

## Introduction

This document constitutes a walkthrough for the "IR Timeline Analysis" challenge category on the Try-HackMe platform.

## 1. Lab Description

The lab aims to teach how to use tools for incident timeline analysis.

## 2. Timelines with Log2Timeline

### 2.1. Section Description

The task involved using the Log2Timeline tool to recreate basic information and the incident timeline.

### 2.2. Analysis and Answers

1. **What option can be used with Log2Timeline to indicate the timeline output file?**  
*Answer: -storage-file*
2. **Based on the Jimmy\_timeline.plaso file, how many event sources are parsed after running pinfo.py against the storage file?**  
Command used: `pinfo.py Jimmy_timeline.plaso | more`  
*Answer: 4982*
3. **On the same timeline file, how many events were generated for the firefox\_history?**  
*Answer: 50*
4. **Based on the B4DM755 timeline, what time was the interview.txt file created? (hh:mm:ss)**  
Commands:  
— `log2timeline.py -storage-file Forensic_Image_b4dm755.plaso Forensic_Image_b4dm755.E01`  
— `psort.py -o dynamic -w Forensic_Image_b4dm755.csv Forensic_Image_b4dm755.plaso`  
— `grep -i "interview.txt" Forensic_Image_b4dm755.csv | grep "Creation Time"`  
*Answer: 14:02:34*

## 3. Timeline Analysis with Timesketch

- **How many data types were in the Jimmy Supertimeline sketch?**  
*Answer: 48*
- **How many entries were in the EVTX Gap Analysis under the Jimmy Supertimeline?**  
The EVTX analyzer report gives us:  
*Answer: 34870*
- **Which search engine did Jimmy Wilson use to search for "how to disappear without a trace?"**  
*Answer: Bing*
- **What is the path of the program that was called to initiate Microsoft Antimalware Service?**  
*Answer: C:\Program Files\Microsoft Security Client\MsMpEng.exe*

## 4. Timeline Analysis Practical

- **How many event sources were identified?**  
Command: `pinfo.py Timeline_Challenge.plaso | more`  
*Answer: 189100*
- **How many events were generated from the dpkg parser?**  
*Answer: 14718*

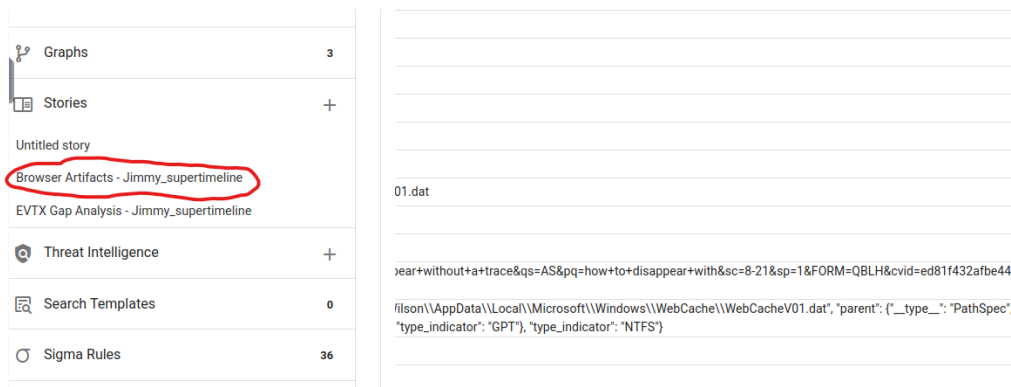
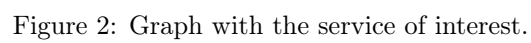


Figure 1: Browsing report.

- **How many total tags were set?**  
 Command: `psort.py -o null -analysis tagging -tagging-file tag_linux.txt Timeline_Challenge.plaso`  
 Answer: *5408*
- **What is the highest tagged element?**  
 Answer: *login\_failed*
- **Under which username does the cronjob that executes app.py run?**  
 Answer: *smokey*
- **What is the hash of the successful SSH login with the PID 1669?**  
 Answer: *a2407e0f3c80d01d2369f15e2b8aa279e790eaa0b1d20ab71cd35c2c7f5ace71*

#### 4.1. Conclusions

During this lab, we learned how to read artifacts from a timeline using Log2Timeline, as well as other information using the Timesketch tool. Then we approached the challenge, where we applied the acquired skills in practice and found some information about SSH login.



```

analyst@ip-10-67-128-154: ~/Desktop/Timelines/Task6
File Edit View Search Terminal Help

Total
824802
0 0 0 0

Identifier PID Status Memory Events
Tags Reports
Main 2975 finalizing 189.3 MiB 824802 (0)
5408 (0) 1 (0)
tagging 2978 completed 88.6 MiB 824802 (549)
5408 (6) 1 (1)

Processing completed.

***** Analysis report: tagging *****
Date and time : 2026-01-19T22:44:26.000000+00:00
Event filter : N/A
Results : application_execution: 82
: application_install: 1990
: boot: 26
: device_connection: 144
: device_disconnection: 16
: event_tags: 5408
: groupadd: 24
: groupdel: 4
: login: 190
: login_failed: 2454
: logout: 220
: runlevel: 24
: session_start: 120
: session_stop: 88
: shutdown: 2
: useradd: 20
: userdel: 4

analyst@ip-10-67-128-154:~/Desktop/Timelines/Task6$ ^C

```

Figure 3: Result of Linux tags analysis.

Saved Searches 0

**Data Types 8**

- fs:stat (360.8K)
- syslog:line (21.9K)
- systemd:journal (20.3K)
- linux:dpkg\_log:entry (7.2K)
- linux:utmp:event (1.7K)
- linux:apt\_history\_log:entry (36)
- syslog:ssh:login (29)
- syslog:cron:task\_run (19)**

Tags 0

Graphs 3

Rows per page: 40 1-19 of 19

message

T03:26:29.000Z Cron ran: /usr/bin/python3 /var/opt/... Timeline\_Challenge

(smokey) CMD (/usr/bin/python3 /var/opt/app/app.py)

/usr/bin/python3 /var/opt/app/app.py

syslog:cron:task\_run

2022-03-02T03:26:29.000000+00:00

EXT:/var/log/syslog.1

☒ biblioteca

Cron ran: /usr/bin/python3 /var/opt/app/app.py for user: **smokey** pid: 730

{\_\_type\_\_: "PathSpec", "inode": 19857, "location": "/var/log/syslog.1" "parent":  
 {\_\_type\_\_: "PathSpec", "volume\_index": 0, "location": "/lvm1", "parent":  
 {\_\_type\_\_: "PathSpec", "location": "/p3", "parent": {\_\_type\_\_: "PathSpec",  
 "parent": {\_\_type\_\_: "PathSpec", "location": "/home/securitynomad/Desktop/  
 Forensics-Timeline/Timeline\_Challenge.dd", "type\_indicator": "OS"},  
 "type\_indicator": "RAW"), "type\_indicator": "GPT"), "type\_indicator": "LVM")

Figure 4: Finding the app.py call.

Timelines

1

Saved Searches

0

Data Types

8

fs:stat (360.8K)

syslog:line (21.9K)

systemd:journal (20.3K)

linux:dpkg\_log:entry (7.2K)

linux:utmp:event (1.7K)

linux:apt\_history\_log:entry (36)

syslog:ssh:login (29)

syslog:cron:task\_run (19)

Tags

0

Graphs

3

Stories

+

Threat Intelligence

+

471 days

2024-03-27T06:15:47.000Z

Successful login of user: smokey from 10.10.147.234:39802 using authentication method: password ssh pid: 1669

authentication_method	password
body	Accepted password for smokey from 10.10.147.234 port 39802 ssh2
data_type	syslog:ssh:login
datetime	2024-03-27T06:15:47.000000+00:00
display_name	EXT:/var/log/auth.log
hostname	biblioteca
ip_address	10.10.147.234
message	Successful login of user: smokey from 10.10.147.234:39802 using authentication method: password ssh pid: 1669
path_spec	({"__type__": "PathSpec", "inode": 6633, "location": "/var/log/auth.log", "parent": {"__type__": "PathSpec", "volume_index": 0, "location": "/lvm1", "parent_security_nomad/Desktop/Forensics-Timeline/Challenge.dd", "type_indicator": "OS"}, "type_indicator": "RAW"}, {"type_indicator": "GPT"}).
pid	1669
port	39802
protocol	ssh2
reporter	sshd
sha256_hash	62407ed0d680501d2369f15e2b8a5c99e790aa0b1d20b71cd35c2c7f8ae7f
source_long	SSH Log
source_short	LOG
tag	

Figure 5: Finding the SSH login hash with PID 1669.