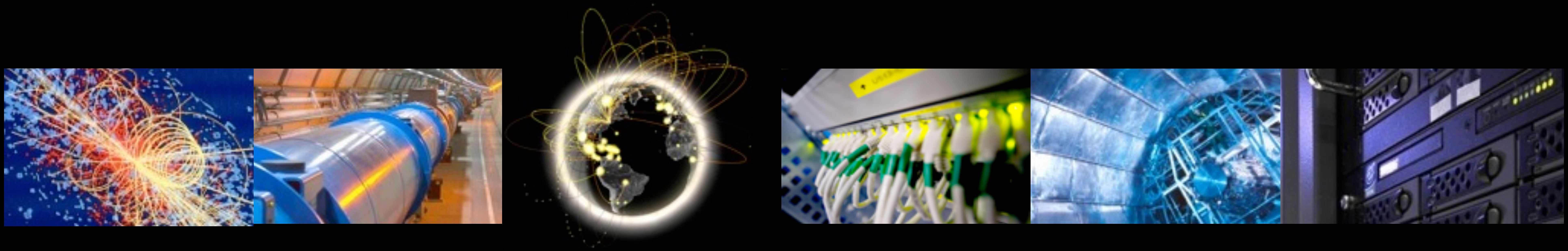
Introduction to Linux Forensics

18-04-2024 / ALICE Workshop



https://indico.cern.ch/event/1355985

Introduction

- Goals of this presentation:
  - Go through the basics **steps previous to the forensics analysis**
    - Don’t panic, don’t tamper evidences.
  - Show the different roles
    - Data Collection
    - Data Analysis
    - Communication/Reporting

Importance of Data Collection

- Create the incident **timeline**
- Search for other **infected systems**
- Assess the **impact** and review the activity
- Guide **mitigation** efforts
- Retrieve non-tampered **evidences**
  - "An independent third party should be able to examine the processes and achieve the same result."
- Identify Indicators of Compromise (**IOCs**) -> Threat Intelligence
- Attribution

Live Collection

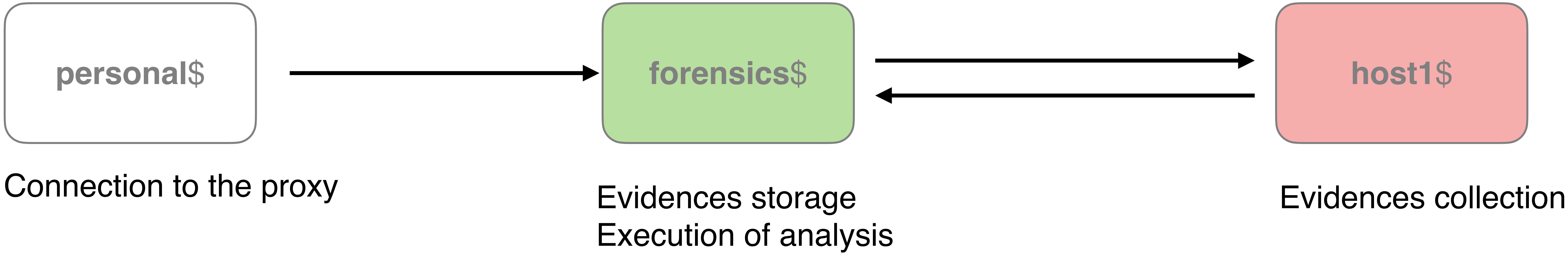
- Pros:
  - Captures the **live state** of the system, including processes and network connections.
  - Allows normal **operations to continue** uninterrupted.
  - Size matters, **reduces the volume** of data generated and bandwith of transfer.
- Cons:
  - **Alters the state** of the machine during the collection process.
  - **May compromise the integrity** of the data collected.
  - Requires **precise knowledge** of what to look for and what to collect.
  - Typically, it is a **one-shot** opportunity before the machine is isolated; data not collected at this time will be lost.

Challenges

- Incident Response challenges:
  - Status of different **work streams** and sharing **findings**
    - Which Investigator did what/when?
  - **Workload** distribution
  - Lack of **communication**
- … and WLCG:
  - Large-scale incident response investigations
    - Geographical distribution, time zone issues
    - Coordination with other security teams
- How can you help as site admins?
  - Provide **clear, concise and detailed information.**
  - If you are not sure how to proceed, ask. **Don't tamper data.**

Preparation

- Snapshots and remote backup
- Disable central services log rotation such as DNS and firewall
- Deploy forensic instance and toolkit
  - **Do not use your device for forensics!** It might be seized for LE.



Basic Investigation

- Roles and procedures
- Investigation Timeline
  - – Notepad or collaborative editing to **track events**
  - – **History of the investigator** commands

A. Interactive (connecting to the host):

- Loads the user's shell profile (~/.bash_profile, ~/.bashrc)
- Affected logs: ~/.bash_history, auth.log, secure, syslog, messages, lastlog, wtmp, btmp

```
forensics$ script ~/evidences/investigation-host1.txt
forensics$ ssh myuser@host1
host1$ export PS1='[\D{%FT%T%z}] \u@\h \w\$ '
host1$ unset HISTFILE
```

B. Non-Interactive (remote execution):

- Useful for scripts and automation

Data Forwarding

- Locally
  - Mounted disks (ssd, usb, ..)
  - Partitions
- Network storage
  - Evidences forwarding:

```
host1$ scp -r /tmp/mypath [USER]@forensics:~/evidences
host1$ tar -zv /tmp/mypath | nc forensics [PORT]
```

  - Per-command forwarding:
    - Smaller blocks of data.
    - Lower risk of loosing data
    - Storage available

Volatile data

- Most of the times it contains the **key indicators** (using much less space)
- **System** info (date, kernel, packages)
- **Network** State (open ports, program2port, interfaces, routing tables, pcap)
  - Firewall status, not only to allow but also to remove competence.
  - Connections to other hosts
- **Processes** and connected users
  - Masquerade processes (impostor syndrome)
  - scheduled tasks -> cronjobs
  - User/group responsible
- **Files**: Open, deleted, memory
- **Kernel modules**
- Mounted **filesystems**

Memory Collection

- Full memory
  - AVML

```
host1$ avml /tmp/mypath/memory.dmp
```

  - LiME
    - Kernal dependency
    - Requires module
- Single processes

```
host1$ export PID=12345; kill -STOP $PID
host1$ cp /proc/$PID/exe $PID.exe
(A) host1$ gcore $PID
(B) host1$ gdb -p $PID # Type gcore, detach, exit
```

Non-volatile data

- Containers: last layer
- Instances: Disk images
  - Local Capture

```
host1$ dd if=/dev/sdX bs=4M | gzip -c > /tmp/mypath/
```

  - Remote Capture

```
host1$ dd if=/dev/sdX bs=4M | gzip -c | nc forensics [PORT]
forensics$ nc -v -l -p [PORT] > ~/evidences/image.dd.gz
```

  - Other tools:
    - Dc3dd and dcfldd
    - Forensics extras
    - Hashing logs

Artifacts Collector

- Unix-like Artifacts Collector (UAC):
  - Profiles allow customisation
  - ir_triage is complete but may trigger errors. Progressive capture reduces risks.

```
host1$ git clone https://github.com/tclahr/uac
(A)host1$ cd uac; ./uac -p ir_triage /tmp/mypath
(B)host1$ ./uac -a live_response/network,live_response/process /tmp/mypath/
```

Network Capture

- Remote logging should be enabled
- If there’s no remote logging available

```
host1$ tcpdump -G [SECONDS] -W 1 -w /tmp/mypath/host1.pcap -I [INTERFACE]
```

  – Memory and logs also contain network data.

Other Sources

- Remote Logs
  - System Logs
    - Logins, Commands executed, etc.
  - Firewall Logs
  - Network traffic
- Extra Logging
  - What we do extra in interactive/batch nodes:
    - Commands + network: Auditbeat (levereage auditd)
      - 32 bit syscalls: execve,execveat. connect,accept
      - 64 bit syscalls: execve,execveat. connect,accept4
  - We also have packetbeat (data transmitted etc, much more heavy and noisy)

Forensics Analysis

- Access
  - Backdoors and persisting malware
- Users
  - Who has interactive logins? non expected sudo accounts? bash history
- Filesystem
  - Hidden files, Deleted files, Owner/group Changes, mac times
- Working with binaries
  - Use a sandbox (better one that you don't run: https://www.joesandbox.com/, any.run)
  - Reverse engineering (Ghidra, IDA, Binary Ninja, Cutter), file, strings, hex editor
- Network
  - Connections and DNS resolutions

Post-exploitation

- Backdoors
  - Account modification
    - New (sudo) accounts added
    - Password changes
    - Service accounts roles
    - $HOME/.ssh/authorized_keys entries added
  - New or replacement binaries (or libraries) i.e. RATs
  - Web shells
- Persisting malware
  - Service start-up scripts
  - Scheduled tasks
- More ideas:
  - Mittre Att&ck https://attack.mitre.org/tactics/TA0003/

Analysis Tools

- Volatility
  - Parse and extract in-memory kernel structures from a memory dump
  - Requires building a profile from the specific kernel

- The Sleuth Kit (TSK)
  - Parse and extract files and other fs objects from raw data disk images or partitions
  - Data carving

```
forensics$ mmls image.dd
forensics$ fls -o 20992 -r -m / image.dd > fls_20992.txt
forensics$ for file in fls_*.txt; do mactime -b $file >
```

File Timestamps

- Mactime or MAC(b), depends on the file system type and mount options
- Modify (content), Access (content), Change (metadata), Birth
- Modify and Access time can be changed at will by file owner
- Reading a file will modify access time
  - With caveats: relatime (default in most systems), noatime, ro

Demo

Questions?
Suggestions are welcomed!

Jose.Carlos.Luna@cern.ch

Pau.Cutrina@cern.ch