Log analysis and digital forensics

M4 – Threats, attacks and countermeasures

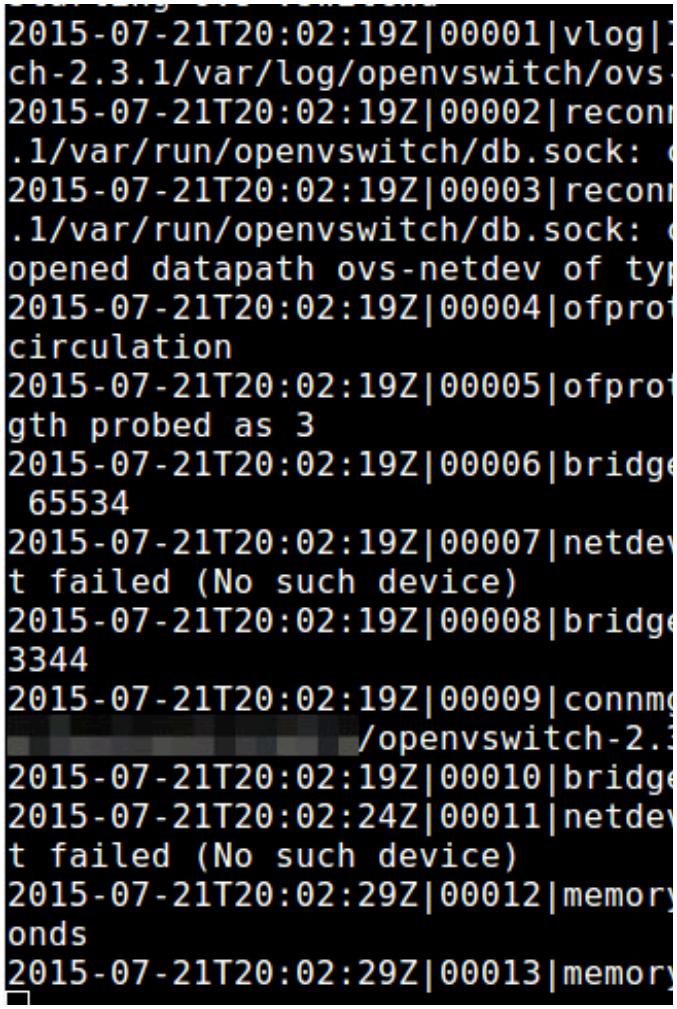
Management de la Sécurité des Systèmes d'Informations

In flagrante delicto…

or "caught in the act" is the ideal situation for every security officer, be it in the physical or in the electronic world. Why is it so **utopian** ?

- Good **monitoring**, **logging**, and **data capture** systems (SIEM, honeynet) that can provide needed information *for a catch* in real-time are **not often implemented** or used.
- Further and more important issues are the **legal aspects**, especially in Europe, where strong privacy and data protection regulations **disallow monitoring** of employees' activities.

➢ The grab to **forensic analysis** tools is **quasi inevitable**. Computer systems are huge and complex, changing very rapidly and even on well monitored environments things can hide, alarms can be miss leading, etc.



Part 1

Logging

Log management & analysis (SIEM)

In a nutshell

- logging is good, log everything
- define the scope of coverage
- define what events constitute a threat
- detail what should be done about them in what time frame
- document when they occurred and what was done
- document where both the events and follow up records can be found
- document how long events and tickets are kept

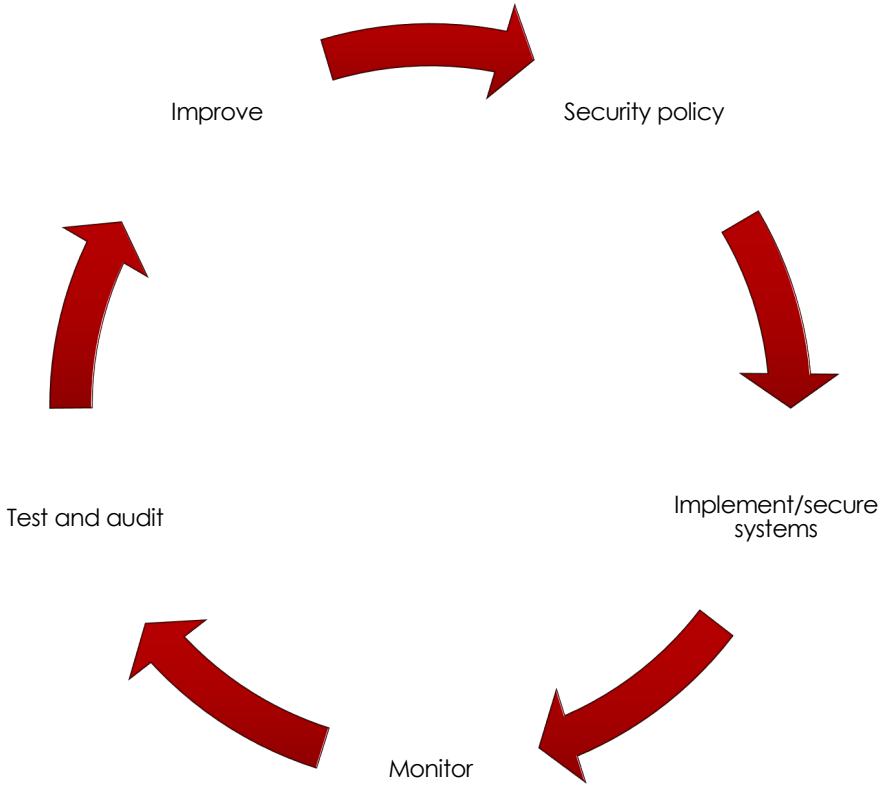
What are logs ?

- **Historical** record of events that happened.
- Record of events and status of systems in a **time sequential** format.
- Record of **activity** on a system/network.
- **Objective**: to provide an audit trail of who done what, where, when and why (**5 Ws**)

Why are logs important ?

- Logs can assist you in:
  - Determining what happened (audit trail)
  - Intrusion detection
  - Incident containment
  - Forensic analysis
  - Proactive protection
  - Real-time alerts
  - Providing a network baseline
  - Determining the health of the network
    - Troubleshooting issues
    - Proactive maintenance

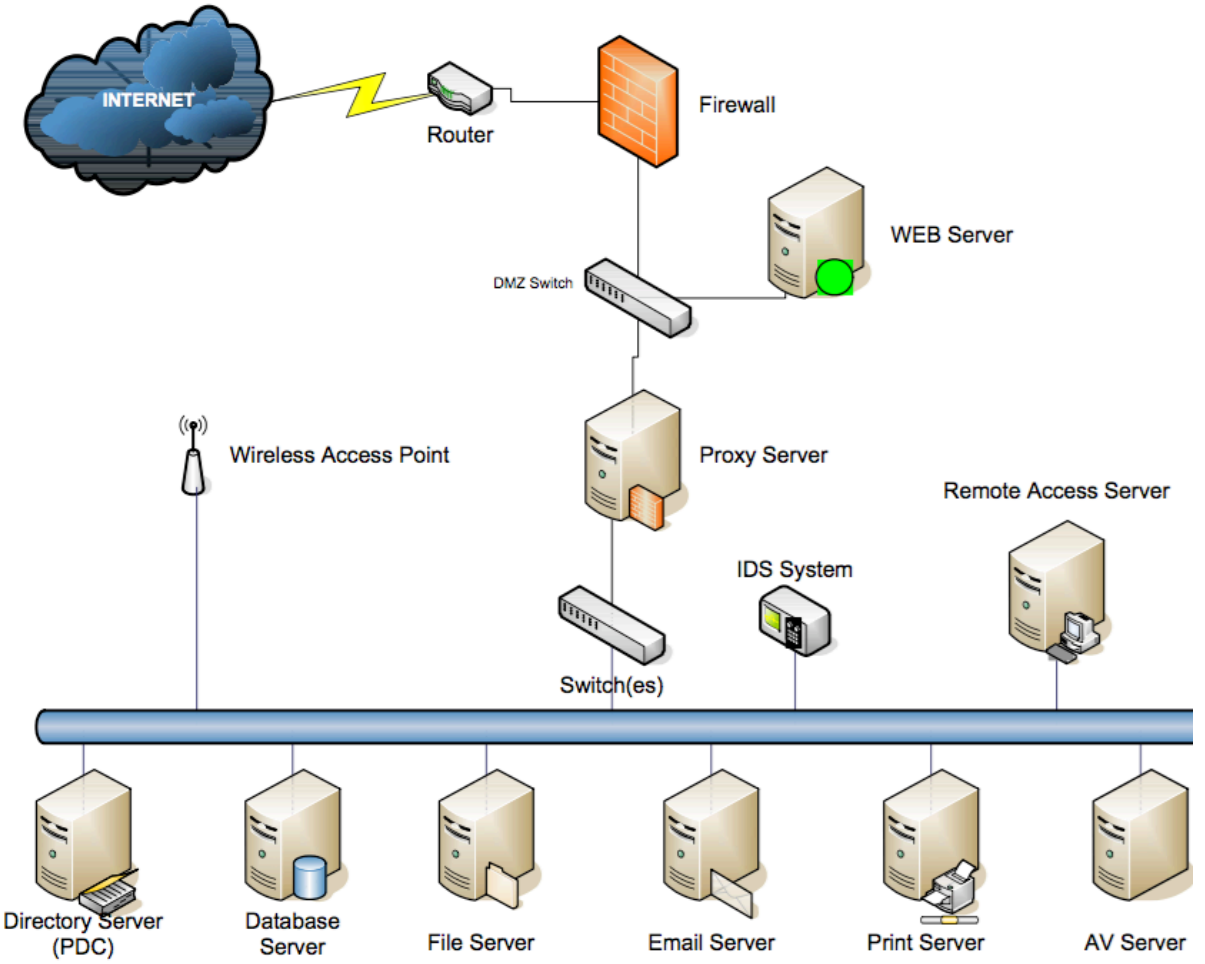
Monitoring as part of an ISMS



Logs are everywhere

All this information should make our jobs easier.

Right?



Challenges

- Logs contain enormous amount of information
- Logs are “written” by developers
  - Format is not easy to read
  - Messages can be obscure
- Different vendors different log formats
- Identifying anomalies can be difficult
  - Probes over time
- Regulatory requirements

Challenges (2)

- Managing logs can be expensive
  - Looking at all events takes time
  - Logs can consume a lot of disk space
  - Log analysis is a quite unique skill
- Volume of information is huge
- No one size fits all
  - Each network is unique

Log analysis

HTTP access example

```
118.78.23.83 - - [18/Oct/2012:11:05:45
    +0200] ``GET /favicon.ico HTTP/1.0''
    200 4531 ``-'' ``Safari/6534.57.2
    CFNetwork/454.12.4 Darwin/10.8.0 (
    i386) (MacBookPro5%2C1)''
```

What can go wrong ? an example

```
118.78.23.83 - - [18/Oct/2012:11:05:45
    +0200] ''GET /favicon.ico HTTP/1.0''
    200 4531 ''-'' ''Safari/6534.57.2
    CFNetwork/454.12.4 Darwin/10.8.0 (
    i386) (MacBookPro5%2C1)''
```

TCP source port of the client is missing.

Without the source port, the IP address and an adequate time-stamp an ISP won't be able to find back the client if this is behind a NAT device (e.g. Internet mobile access is usually behind a carrier grade NAT device)

Destroying evidence without knowing it

- Aggregation, filtering, correlation or alerting might mean destroying evidences
- As an example, you ask for the raw HTTP logs from a proxy
- It's logged but then aggregated to produce usage report
- In incident analysis, you need the raw logs not the high-end report or simplified alert

Never logging the required evidences

- A chain of 4 reverse proxies (WAF) logging at each state but missing a single X-Forwarded-For in the chain
- If you need to rebuild an end-to-end session from logs, you need to keep track of the source IP address for each records

Time synchronization
good practice

- Have an NTP server with an external source
- Every device shall be synced against that server
- How do you check if your NTP server is really at the correct time?
- A 3 minutes drift when merging 200GB of one hour logs is a big . . .

Proprietary log-formats

- Do you need special tools to extract raw logs? Are those accessible to you ?
- Are the raw logs viable as evidence?
- How long does it take to extract the logs? (e.g. 10 days to extract 5 days)
  - Can you wait 10 days if the "remediation" solution can be found within those logs?

Everything can go wrong with logging

- Log rotation (e.g. what's on going while the rotation is performed?)
- Threshold for limiting logs size (e.g. Many vendors have hard coded values)
- Log analysis requires context (e.g. Is the access from that time period valid? or is it coming from a Malware?)
- Are you just logging denied access? or just accepted access? or both?


More Security Doesn't Make You More Secure
Better Management Does.

Good practices

- **Develop logging Policy**
- **Determine what information is relevant to you.**
  - What devices are important?
  - What events are important?
  - Don't forget to turn on logging!
  - Timing of events, e.g. user logons in morning.
  - What reports you and the business want/need?
  - Group servers into zones based on their function or criticality and prioritise events accordingly.

Good practices (2)

- **Baseline your systems & network.**
  - Determine how your network normally behaves
  - Repeat at regular intervals
  - Define context (e.g. is the event time period valid? or is it coming from a malware?)
- **Secure log files on all devices**
  - Encrypt logs
- **Ensure all devices use same time source.**
  - If using more than one time zone use UTC.
  - Use NTP protocol from a secure source to synchronise time.

Good practices (3)

- **Centralise log collection**
  - Dedicated server to collect all logs.
    - Be careful of network traffic volumes.
    - Be aware of limitations of server to process number of events.
  - Configure all devices send logs to central log server.
  - Make sure central server is secure. Secure transmission of logs.
    - e.g. Syslog uses UDP by default. Consider using IPSec or next generation Syslog (Syslog-NG)

Good practices (4)

- **Normalise the data**
  - All events should be normalised into same format.
- **Review the Logs**
  - Ensure logs are regularly reviewed
    - Manually
    - Automatically

Good practices (5)

- **Log Rotation**
  - Determine time schedule
  - Based on volume of data
  - Develop meaningful naming convention
  - Move data to rotated file
- **Log Retention**
  - May be regulatory requirements
  - Archive onto WORM (Write-Once-Read-Many) type devices and store in secure area

Good practices (6)

- **Using logs for investigation**
  - List and train people to extract logs (and also to read those)
  - Try to manually reconstruct a chain of events from logs
  - Test your log exports (e.g. you ask your local CSIRT/CERT to make a test request)
  - Storing raw logs is usually simpler than creating "aggregated" logs
    - Make sure raw logs are readable by standard text-processing tools
  - Use beacon logs to ensure the effectiveness of your log processing (eg. SIEM)

Part 2

Digital Forensics

Definition

The "art" of (application of science) **identifying, collecting, examining and analysing data** from a **"digital crime-scene"** whilst **preserving the integrity** (*free from distortion or bias*) of the information and maintaining a strict **chain of custody** (*determine and reconstruct what has happened*).

Modus operandi

- Identification & Collection
  - *Anything you do on a system disturbs it*
- Examination & Analysis
  - *Never work on the original data*
- Integrity preservation
  - *Trust is your enemy*
- Chain of custody
  - Consider to comply to legal framework

The Battle plan

- What has happened and when ?
  - Identify the modus operandi and its timeline
- Why didn't security systems block the intrusion ?
  - Identify valuable logs and alarms
- Which are the affected systems ?
  - Machines & persons involved or victimized
  - Also think about « collateral damage » !
- What data did change on the systems ? What is normality ?
  - Stolen data, modified data or traces left by the offender
- What were the rogue's intentions ?
  - For legal implications it's important to determine the real impact

Identification & Collection

- Abundance of data sources
  - from most obvious: *desktops, servers, firewalls…*
  - to least obvious: *packet captures, netflow, hidden data streams…*
- Collecting/acquiring data, what first?
  - Likely value
  - Volatility
  - Effort required
  - Integrity
  - Preservation

Order of volatilty

Type of Data	Life Span
Registers or cache	Nanoseconds
Main Memory	Ten Nanoseconds
Network State	Milliseconds
Running Processes	Seconds
Disk	Minutes to Hours
Backup Medias	Years
CD-ROMS or printouts	Tens of years and more

Examination & Analysis

- Filter, filter, filter
- Keywords & context
- Timeline
- Correlation
- Chain of custody

Integrity preservation

The analysis of a "crime-scene" or "situation" must be handled with great care. A fixed and precise procedure has to be followed :

1. **Secure and isolate**
   - This first step is somehow converse. Should the network cable be immediately plugged out, or not ? Experts minds are split on this.
2. **Record the scene**
   - Respect the order of volatility of data storage.
3. **Conduct a systematic search of "evidence"**
   - Think, define goals, targets (based on logs etc. of unaffected systems). Assume the worst, but move carefully and most important log (write down) every action.
4. **Collect and package evidence**
   - Spot and get data that can be used as evidence, by correlating various redundant sources, to be able to draw coherent conclusions.
5. **Maintain the chain of custody**
   - Keep the parent-child persistence and reconstruct the time line, to ensure traceability and reproducibility

Integrity preservation (2)

- Anything you do to a system disturbs it
- Keep data as original (unbiased) as possible
- Don't work on the original
- Speed is of the essence - but don't overdo it
- Consider the policies, conform to the legal framework
- Prepare to be surprised

Chain of custody

- *Being able to clearly describe how the evidence was found, how it was handled and everything that happened to it.*

Document, document, document:

- Where, when, and by whom was the evidence discovered and collected.
- Where, when and by whom was the evidence handled or examined.
- Who had custody of the evidence, during what period. How was it stored.
- When the evidence changed custody, when and how did the transfer occur (include shipping numbers, etc.).

Tools you'll need

- A program for examining processes (e.g., 'ps').
- Programs for examining system state (e.g., 'showrev', ‘ifconfig', 'netstat', 'arp').
- A program for doing bit-to-bit copies (e.g., 'dd', 'SafeBack').
- Programs for generating checksums and signatures (e.g., 'sha1sum', a checksum-enabled 'dd', 'SafeBack', 'pgp').
- Programs for generating core images and for examining them (e.g., 'gcore', 'gdb').
- Scripts to automate evidence collection (e.g., The Coroner's Toolkit / Sleuthkit)).

Case example

Prepare for the battle

- Trust is your enemy
  - Only us “your tools”, for instance statically linked binaries on a CD or other write-protected media
- First collect, analyse later!
  - Best is to prepare a secure and trusted machine to collect all the data.

Planning, documentation & the chain of custody

- **Planning** is King
- **Documentation** is God
  - Documenting all the actions during a forensics analysis is paramount. Sometimes you only have one chance to get an info or respond correctly, especially when working on a hot (on-line) victim.
- Doing **checksums** of all transferred data for later consistency is very important
- Identical ***trusted/untrusted*** execution can be useful to check if the offender did really modify things

Example *logbook*

Time	Command	Trusted	Untrusted	Checksum	comments
15/01/2005 10:32:15	dd < /dev/mem | cryptcat -w 3 key coroner 6969	x		689d65e97da d5b8d1a35b3 600c3f7b8e	
15/01/2005 10:38:27	lsof | cryptcat -w 3 key coroner 6969	x		0c2e968f8560 0d3f33bec543 3ca1345d04	
15/01/2005 10:43:12	lsof | cryptcat -w 3 key coroner 6969		X	32a3a45f4562 de254cc35fe3 62ca51ee	

Memory and volatile data

- Memory: *dd is your friend*, example usage :
  - `dd < /dev/kmem  > out0`
  - `dd < /dev/mem  > out1`
- Date and time info of the victim host, to be compared with non-affected hosts
  - `date`
- Running processes and open files
  - `ps eaxlf`
  - `lsof`
- gcore (dump memory for a given process)
  -
```
dcore 346 | strings > 346.mem
grep `[pattern]' 346.mem
strings 346.mem | less
```
- also check content of /proc (in /proc/PID/exe the current executable can be found)

Network (config and activity)

- `ifconfig`

```
eth0 Link encap:Ethernet  HWaddr 01:0C:5F:E3:C2:95
inet addr:192.168.1.18  Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: f480::2dc:6eff:fde3:c295/64 Scope:Link
UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
RX packets:73741 errors:0 dropped:0 overruns:0 frame:0
TX packets:19848 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:21053599 (20.0 MiB)  TX bytes:1738710 (1.6 MiB)
Base address:0xcf80 Memory:fe9e0000-fea00000
```

- netstat ; route ; w (show who is logged and what they are doing)

- `last`

```
th     pts/3  vega              Thu Jan  6 09:21 - 17:30  (08:08)
loic   pts/1  asterix.gaule.fr  Thu Jan  6 09:14 - 11:26  (02:12)
loix   pts/1  asterix.gaule.fr  Thu Jan  6 08:24 - 08:24  (00:00)
ben    pts/1  warp.skynet.be    Thu Jan  6 07:54 - 08:09  (00:14)
andy   pts/1  cpc-ofd2-6-0-c    Thu Jan  6 03:00 - 03:11  (00:10)
bibi   pts/1  coco.internet.lu  Wed Jan  5 22:34 - 22:38  (00:04)
```

- `tcpdump -s0 -i eth0 -w evidence.cap`

Kernel and hardware info

- kernel modules/config/patches
  - `lsmod`
- hardware info
  - `lspci`
  - `dmesg`

Disk capture and analysis

- Not simply a backup! Make an identical copy of the filesystem to a secure location.
  - Deleted files are also needed, copy the whole disk not only the partitions (swap is easily forgotten, data may reside on unpartitioned areas).

- Copying the whole disk to the coroner machine using *dd* and *cryptcat*:

```
[victim] dd id=/dev/hdc | cryptcat -w 2 -k key
coroner 6969
```

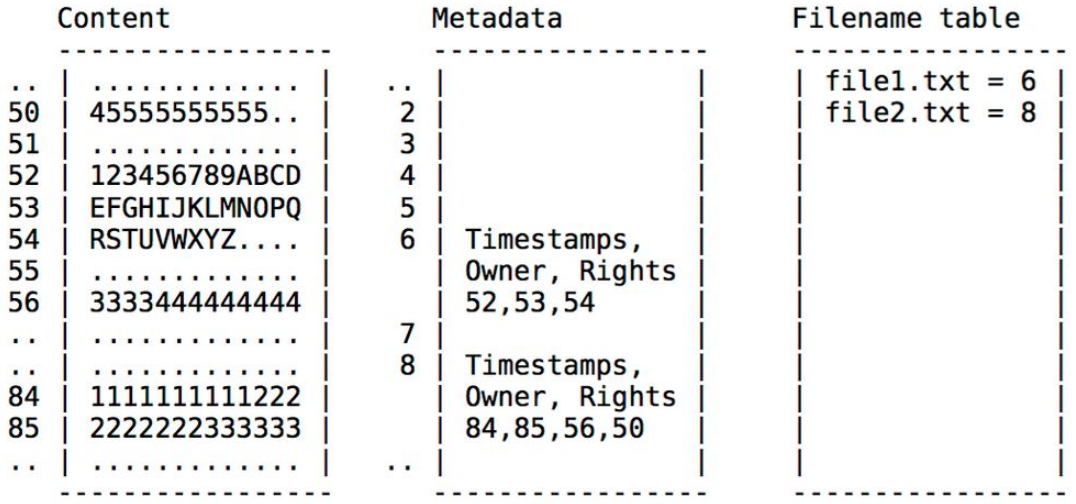
- To reconstruce and work on partitions, separate them into different files:

```
fdisk -lu hdc.dd (to get start/end cycles)
dd if=hdc.dd of=hdc1.dd bs=512 skip=52
count=115119933
```

Exploring the abyss

- Log analysis of trusted systems
  - IDS, firewall, routers, switches, honeynets…
- Recovering deleted files:

```
file1.txt = "123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ"
file2.txt = "11111111112222222222333333333344444444445555555555"
```



Thank you for your attention

Questions ?

Merry Christmas and a Happy New Year