

Analyse-Methoden für forensische Daten

Anmerkungen Tutorium

- WAL , wear leveling
 - bei Flash Speicher werden iNodes durch-rotiert
 - es kann also X-tausend SQLite DBs bei etwa 50 Apps
 - Smartphones haben 30-40 Partitionen
- Autopsy für Smartphone ist erlaubt
 - Sleuthkit für Win Image: da Nintendo nicht ok

Fokus

- Win
- Android

APL Ausgabe

2.12

Win 11

- Sleuth `mm1s`
- verschlüsselt

Logfile

- Logic Analyzer 20 GB
- `sync` nach Kopieren

Linux prep

Die Forensik VM braucht Ressourcen. Es dauert sonst zu lange.

- `losetup -a` : keine SquashFS loop Devices = einfacher
 - bauen von Loop Devices (manuell ?)
 - `--partscan` / `--find` / `--show`
- `log2timeline`
 - optional, für die Generierung für Super-Timeline
- `sleuthkit` ist sehr versions-abhängig
 - 4.12 hat LVM Unterstützung (unwichtig für APL)
- Auswertungen beginnen mit der Listung der Versions-Nummern der eingesetzten Forensik-Programme, sonst sind 2 Punkte weg.
 - nicht alles wie `grep` etc., sondern `sleuthkit` usw.
- ZFS - Filesystem der 3. Generation. FAT16 z. B. ist schwer zu reparieren, da es kein Journal hat. 2. Generation: ext3 / ext4, HFS(+), NTFS - haben ein Journal. 3. Generation: ZFS
- `binwalk` und `yafs` werden eher nicht benötigt werden
- `bulkextractor` sucht Crypto-Currency Kram und `.onion` Darknet URLs
 - nur die alte Version mit Python 2 funktioniert für uns. Version 2 nicht nutzen, nur 1.6
- `scraper` Modul kann *nur* Crypto-Spuren suchen (es reicht für uns, und braucht kein Python 2)

APL Theorie

• Prefetch

- `sccainfo` für das Lesen von *.pf Files
- gehören zu den "Windows-spezifischen Artefakten"
 - HW Wallet (?)
 - Indizien für Crypto Kram (USB -> ?)

• Volume Shadow Copies

- unallocated & allocated - in der VSS Kopie sind auch gelöschte Dateien.
- wenn es sie nicht gibt, dokumentieren -> Screenshot
- Anzahl der Files vom Windows = Anzahl der VSS Dateien
 - plaso erzeugt eine SQLite wo alles drin steht, deshalb dauert es so lange. `log2timeline` erkennt VSS Kopieen automatisch und fragt. Die letzte könnte ausreichen: also "die neueste...".
 - `vshadowinfo` : Sucher
 - `vmdk` : Mounter

• Registry

- `sleuth fsstat` - zeigt Win XP, kann aber 11 sein. XP ist die Versionsnummer von NTFS (!)
 - OS Infos werden ermittelt: aus der Registry
- Angedockte USB Devices erscheinen in der Registry
 - USB Kennung emulierbar . Zeitstempel & Mount Point können evtl. abweichen. Kann man aufzeigen, aber ist semi-relevant ("sieht manipuliert aus").
- RegRipper (`rip.pl` oder so)
- Log2Timeline
 - Geht alle logisch vorhandene Dateien durch, und macht einen Zeitstrahl
 - Empfehlung: ohne VSS Kopieen
 - Timeline: fix, nur Meta-Infos aus Datei
 - Super-Timeline: lange, weil auch inhaltlich

- USB Sick Mount Event hat Zeitstempel

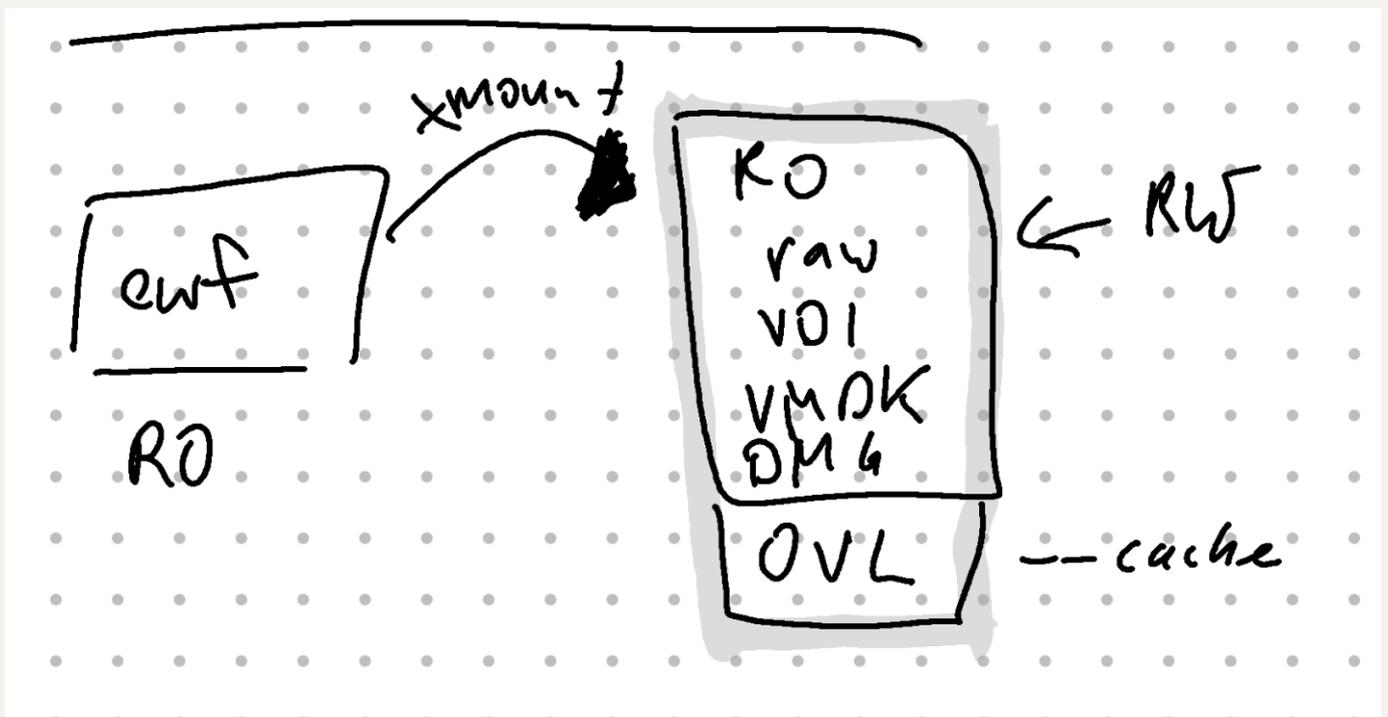
- **Outlook**

- APL irrelevant

- **\$MFT**

- DB-artige Struktur, 4 Zeitstempel (Meta-Infos)
 - Anti-Forensik Tools können idR die Zeitstempel in der MFT nicht manipulieren

Xmount Skizze



Zeitstempel

Anpassung der Zeitstempel unter Linux. Mindestens 3 Stempel, oft 4.

MAC

1. m - modified (Inhaltlich)
2. a - accessed
3. c - Win: creation, Lin: changed (Änderung der Meta-Daten)

MAC B ("Birth")

4. b -

Tutorials

Beispiele DELL & VSS

Demo Trace

- APL: Screenshots nicht als alleiniges Merkmal. Ein Satz muss dazu.
- Ein paar Screenshots sind sinnvoll, da es eine Echtheitsprüfung nach der APL geben kann. Nicht alles muss als Screenshot dargestellt werden.
- Es ist zu empfehlen zu zeigen, dass die Verzeichnisse vorher leer sind

```
1 ##### win xp
2
3 mmls dell13.E0* # klassifizierung
4
5 fls -pro 63 dell13.E0* | grep -i prefetch # prefetch dateien
  ansehen
6
7 # bei FF runcount und last used
8
9 icat -o 63 dell13.E0* 13973 > /tmp/ff.pf
10 # im Prefetch nachsehen ob z. B ein HW Wallet da ist
11
12 ##### win 7
13
14 mmls vss.E0*
15 # 206848 / 2
16
17 # Gibt es VSS Kopieen? -> Doku
18
19 # APL bedarf logisch (mount listing) & physisch (auch gelöschte)
20
```

```
21 xmount --int ewf dell3.E0* --cache /tmp/dell.ovl --out vdi
22
23 # VDI in Virtualisierer rein, vorhandene HD
24 # opengates um offline zu bleiben, kann auch SATA Treiber
    reinmachen
25
26 # HW Raids haben den MBR hinten. Stattdessen ein Image
27 # xmount kann das verschieben
28 # RAID Controller kann man evtl. nicht direkt an VBox
    durchreichen
29
30 # in der APL soll das Win 11 Virtualisiert werden
31 # qemu / Vbox
32 # BIOS mit TPM wird gebraucht
33 # Hier wird virtualisiert nach dem Entschlüsseln
34 # Bitlocker Partition kann ersetzt werden
35 # nested virtualisierung muss man aktivieren
36
37 # OVL wird grundsätzlich gemacht
38 xmount --in ewf css.E0* --cache /tmp/vss.ovl --out raw /ewf
39 mmls /ewf/css.dd
40 # VSS werden in der C:, also in der 2. Partition sein
41
42 # ins RAW Image gehen: xmount in raw muss vorher durch sein
43 losetup --partscan --find --show /ewf/vss.dd
44
45 # /dev/loop7 oder so kommt raus, je nachdem
46 ll /dev/loop7*
47 # kommt dann loop7(,p1, p2) - wir brauchen p2
48
49 vshadowinfo /dev/loop7p2
50
51 # VSS Kopieen bereitstellen
52 mkdir /vss
53 vshadowmount /dev/loop7p2 /vss
54 ll /vss # c-time von den erzeugten Stores, innen sind die VSS
    Kopieen integer
55
56 # aus der VSS Kopie alle gelöschten Dateien raus:
```

```
57 tsk_recover /vss/vss5 /tmp/recovered # in der APL die neuste
58
59 mount -o ro /vss/vss5 /mnt/
60 # kann sein, dass gelöschte Dateien hier sind.
61
```

Registry

- Authentifizierungsverfahren für lokale Accounts in Evolution
- Rainbow Tables für bis zu 7 Zeichen einfach erzeugbar, da unsalted bis Win 7
- SAM file, heute verschlüsselt mit SYSTEM Key
- Am Anfang der Analyse extrahieren:
 - SAM
 - SYSTEM
 - SOFTWARE - **USB device history** (relevant für APL)
 - Service Pack, User, System Information

```
1 mmls dell13.E0*
2 # 63 offset kommt hierher
3
4 # RegEx: $: endet mit SAM
5 fls -pro 63 /dev/loop7p2 | grep SAM$
6 # .. nicht APL relevant
7 # Bei toten Loop Devices in der APL können wir uns Anmerkungen
  schenken, wenn aus loop7 loop7+1 ... wird
8 # WINDOWS /system32/config/SAM - Hive
9 # ntuser.dat - profilbezogener Hive (letzte Dateien geöffnet,
  Drucker benutzt usw.) - wird nicht benötigt (?)
10
11 # Extraktion des Files
12 icat -o 63 dell13.E0* 3667 > /tmp/SAM
13
14 # Cross-check doku,entieren:
15 file /tmp/SAM
16 # da sollte so was wie MS Windows registry file stehen
```

```
17 # nicht ausschweifend beschreiben. Nutzung zeigen, nicht
    wissenschaftlich erklären
18
19 # SYSTEM Hive alles klein, SAM alles groß
20 # -i - case ignorieren
21 fls -pro 63 dell3.E0* | grep system$
22 # es ist die Datei in system32/config/system
23
24 fls -pro 63 dell3.E0* 334 > /tmp/system
25 file /tmp/system
26
27 # Es gibt mehrere ntuser.dat, z. B. bei User Default
28 # Dieser ist ein Template User.
29
30 # Lokale Win Accounts sind heute selten
31 ophcrack &
32 # SAM und SYSTEM per -> Load
33 # Crack per Rainbow table
34
35 # in der APL soll System virtualisiert werden. Als Nachweis:
    eigenes Desktop Bild, Datei abgelegt oder so was (Originalität).
36
37 # Es liegt _k_ein Lokal-Konto vor
38 # * utilmgr.exe durch cmd.exe austauschen, cmd auf beim Login
    Dialog nach dem Boot
39
40 # Regripper plugin listing
41 regripper -l | grep -i ver
42 # zur Bestimmung winver aus dem [Software] Hive
43 # ServicePack, BuildRelease, RegisteredOwner
44 # externe USB Platten sind keine USB Devices bei MS
45 regripper -p winver -r # ? /tmp/system #
46
47 # es gibt verschiedene USB Plugins
48 # Verwendung von EventLogs ist ebenfalls ok
49 # in der Registry stehen auch MAC Adressen <-> Access Point
    Verbindungen, was relevant sein kann
```

Bitlocker Entschlüsselung

Funktionsweise von TPM Sniffing

- per Bus TPM Sniffing mittels Logic Analyzer mit genug MSPS
 - TPM nutzt LPC (7 Signale), SPI (4 Signale) oder I2C
 - Nyquist Shannon: Logic Analyser muss doppelt - 3x so hoch sein
 - PCI Bus 33 MhZ. Also 66 MhZ. 4 - 7 Signale.
 - Ein 500 MSPS LogicAnalyzer reicht für LPC
- FVEK - zum Ver- und Entschlüsseln
- Ent VMK1 - muss entsiegelt werden (Binär-Datei, unter 100k), wird FVEK

APL Vorgehensweise

In der APL wird ein LPC Bus benutzt (also 7 Kanal-System). Die exportierten Werte müssen an "ARNE" übergeben werden. Im Header der Sample-Datei stehen die Spezifikationen.

Saleae Logic 2 herunterladen. -> File -> Open Capture (paar hundert MB *.sal File) -> Export Data -> dauert lange, und wird 20 GB großes CSV.

Tipp: Nach dem DL der Captures sollte der Hash-Wert geprüft werden, sonst droht Zeitverschwendung.

Für den Boot-Vorgang

Hier werden die Daten noch nicht entschlüsselt

```
1 ARNE -i bitlocker_logic_export.csv\ # nicht das .sal File
2 -k "1=LCKL,2=LAD0,3=LAD1,4=LAD2,5=LAD3,6=LFRAME,7=LRESET"\
3 -o wings_apl.vmk
```

Entschlüsselung

`dislocker` : wird gebraucht, nach ARNE. Kann Bitlocker Partitionen unter macOS und Linux einbinden. Hier wird der VMK gebraucht (Version $\geq 0.7.2$).

Partions-Layout:

- sdc1 vfat - EFI Kram
- hidden NTFS am Ende

ARNE sucht nach einer Signatur, und holt einen 32 Byte für den VMK (?); sollte da drin sein wenn man nach dem Sub-String sucht.

```
1  mkdir /tmp/dislocker
2
3  sudo dislocker -h
4
5  # siehe Platform 4n6
6
7  # Eine entschlüsselte physische Partition entsteht wenn
   dislocker funktioniert
8
9  # Hex-Editor:
10 xxd dislocker-file | less
11
12
```

- 0..9615 - Block 1
- 9616 - Block 2 : Bitlocker
- Block 3: NTFS

Block 2 wird ersetzt. `dmsetup` hilft.

Zugang ohne PW

- system32: utilmgr.exe <- cmd.exe - Ersetzen. Vorher ein Backup machen.

APL Infos

2 Crypto Spuren `scraper` testen...

1. `mmls bitlocker_image.E01`, Partition mit den meisten Sektoren ist wichtig
2. `xmount --in ewf botlocker_image.E01 --cache /tmp/bl.ovl --out raw /ewf` -> Bitlocker Image 128 GB oder so
3. Physisch & Logisch, also gleich Loop Devices machen mit `losetup --partscan --part --show /ewf/bitlocker_image.dd`
4. `ll /dev/loop7*`, in den 3er rein
5. `fsstat /dev/loop7p3` -> "Encryption detected". Also
6. `head /tmp/digital.csv` - SCLOCK, MOSI, MISO etc. also SPF
 - a. `ARNE -i /tmp/digital.csv -k "1=SCLK,2=CS,3=MOSI,4=MISO" -o hh.vmk`
 - b. Start SPI Decoder -> Läuft :)
 - c. TPM Frames bei umdie 80%
 - d. LPC in der APL wird dauern
7. `hh.vmk` sollte in `/tmp` liegen, wenn erfolgreich extrahiert
8. `dislocker` wird gebraucht
 - a. Copy & paste auf der Platform `mkdir /tmp/bitlocker` (Physisch), das andere logisch.
 - b. diese `$()` Shell Mathe wird gebraucht, wenn loop Devices manuell erzeugt werden.
 - c. `dislocker -K /tmp/hh.vmk -V /dev/loop7p3 /tmp/bitlocker - 127 GB` oder so.
9. `fsstat /tmp/bitlocker/dislocker-file | less` - XP heißt nicht Windows XP, sondern NTFS Version
10. `fls /tmp/bitlocker/dislocker-file`

11. `utilmgr.exe` mit Sleuth backuppen, kopiere dann `cmd.exe` drüber.
12. loopdevice auf dislocker file
13. dmsetup.txt erzeugen gemäß Tabelle
14. Neues loop Devices gem. NTFS Partition, und dann noch was hinten dran kommt. cat Befehl aus dem 4n6
 - a. Basic Data Partition nicht mehr Bitlocker, sondern NTFS
15. `xmout --in raw /dev/mapper/merged --cache /tmp/blohne.ovl --out vdi /ewf2` (virtuelles Device, kann also durch Reboot verschwinden). EWF wird 12 GB oder so. RAW 128 GB.
 - a. `ewf2` muss vorher angelegt werden
16. ...
17. VDI für VBox für die 100% Lösung, also Gast OS bauen

APL Q&A

- `web_yellow`: .onion URLs könnten drin sein. Hidden Services werden nicht per Default installiert.
 - Raspberry Pi
 - Apache2 oder so `/var/www/html`
 - Statische Seite zeigen
 - Ist der Web Auftritt im Darknet on? -> TOR Browser installieren, aufmachen, Screenshot
 - `/etc/torrc` macht Hidden Services verfügbar
 - `/var/lib/tor/hidden_service`
 - `cat hostname` - .onion URL
 - `python3 -m http.server 80` aufmachen aus `/tmp/index.html` um den internen HTTPd an den Hidden Service zu knüpfen
- Session0.sal
- md5 Hashes sollen identisch sein, `-c` ist zu nehmen um es zu zeigen. Für alle Files

- Frage 4: HW Wallet, könnte .onion URL. Keine Fall-Interpretation oder Investigativ-Beschreibung.

foo

Angaben können inkorrekt sein.