Forensics, Malware and Penetration Testing

Disk Forensics

David Oswald and Andreea Radu

University of Birmingham

d.f.oswald@bham.ac.uk

Outline

1. Disk forensics* ←

2. Log file forensics

3. Network forensics

4. Memory forensics

5. Mobile devices (Android)

* May need RAM forensics, e.g., in case of full-disk encryption

Books? (without warranty)

- File System Forensic Analysis by Brian Carrier
- NTFS Forensics: A Programmers View of Raw Filesystem Data Extraction by Jason Medeiros
- Forensics Wiki: https://forensicswiki.xyz/page/Main_Page

Disk and file forensics what?

Block devices vs filesystems

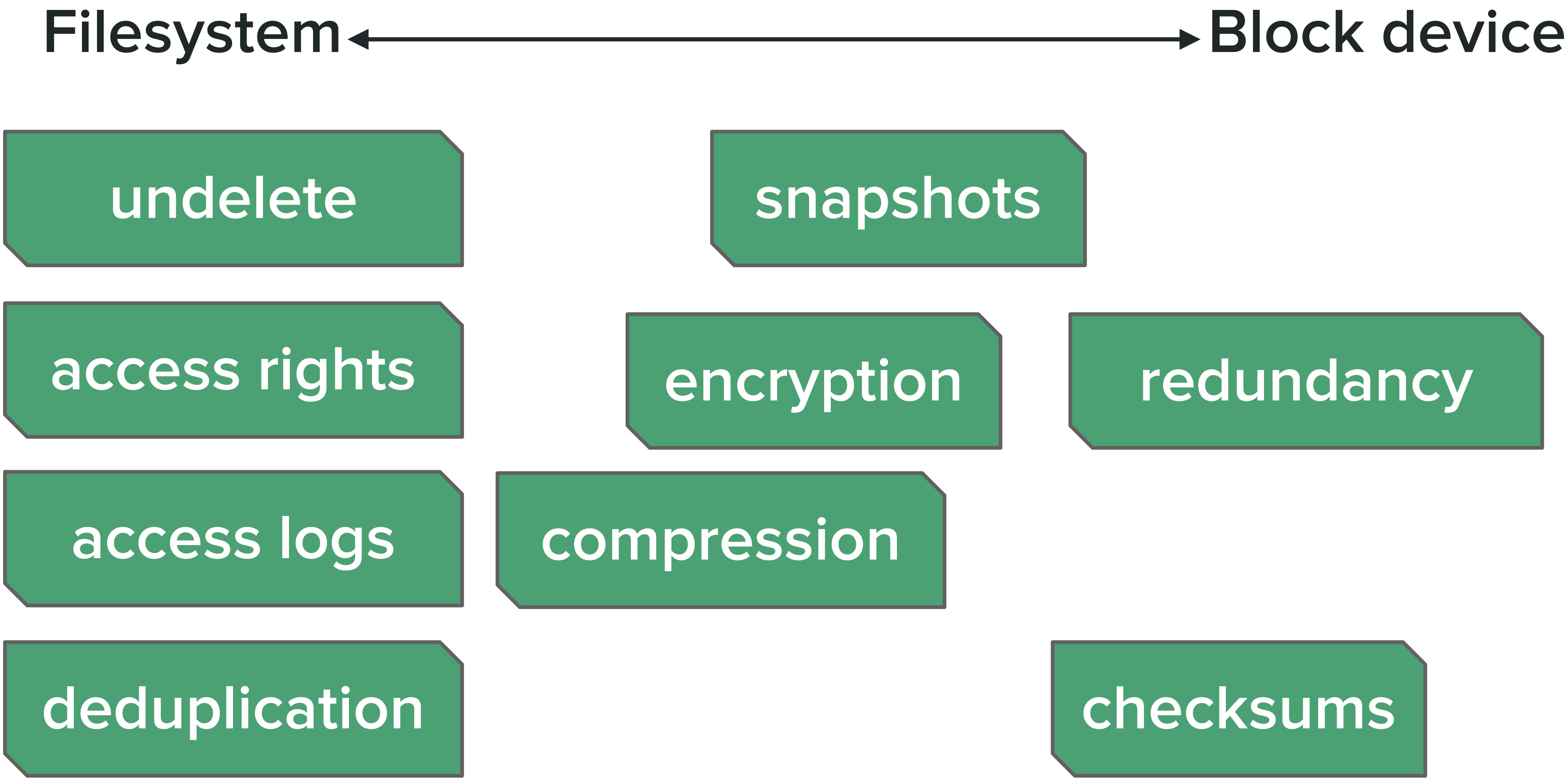
Block device

- Visible as a list of blocks to the operating system
- Usually supports random access
- May have “weird“ properties for access times

Filesystem

- View on / namespace for storage resources
- Stores contents of objects (files, dirs) with associated metadata

Supported features



Filesystem

Block device

undelete

snapshots

access rights

encryption

redundancy

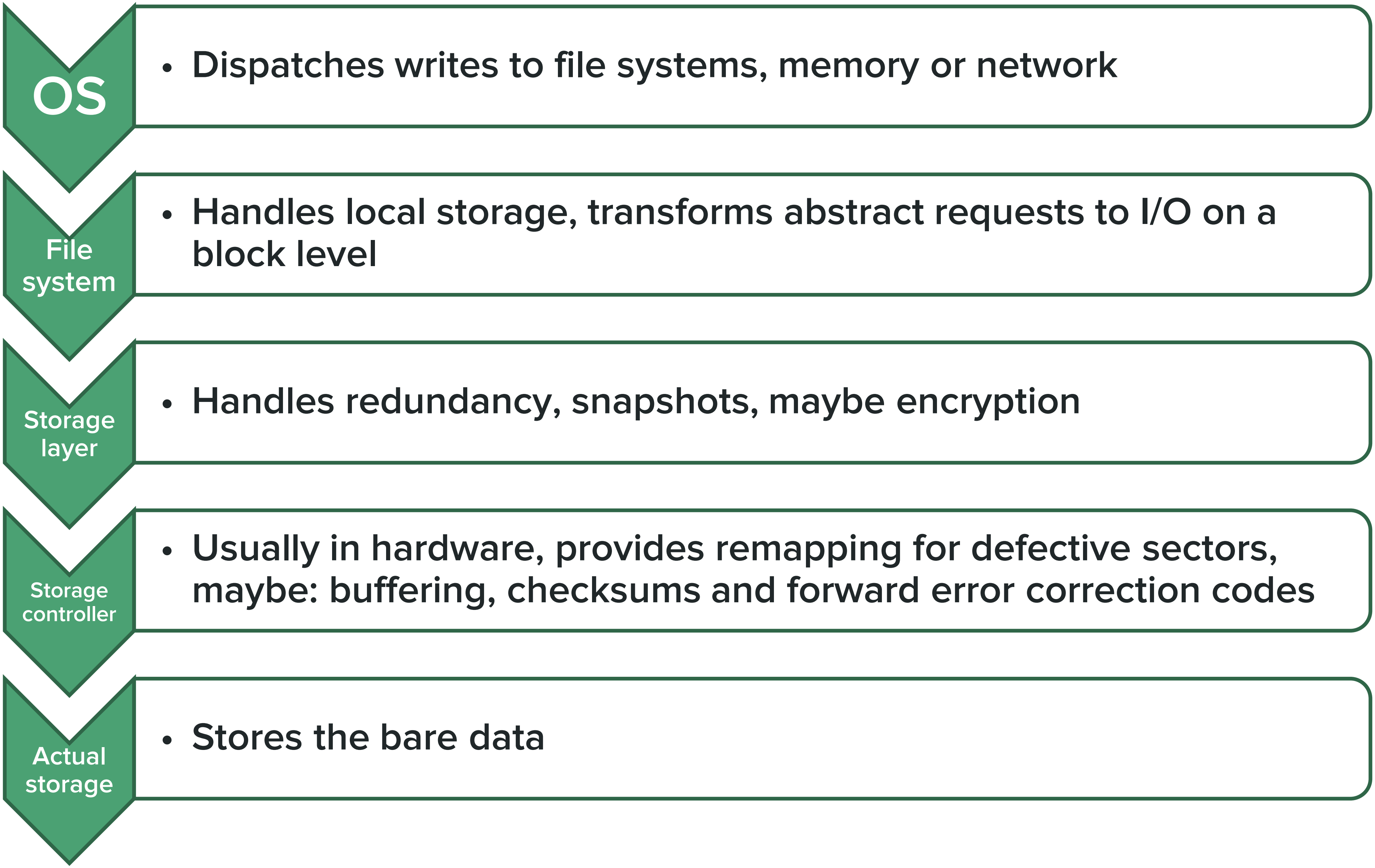
access logs

compression

deduplication

checksums

Storage architecture



OS

- Dispatches writes to file systems, memory or network

File system

- Handles local storage, transforms abstract requests to I/O on a block level

Storage layer

- Handles redundancy, snapshots, maybe encryption

Storage controller

- Usually in hardware, provides remapping for defective sectors, maybe: buffering, checksums and forward error correction codes

Actual storage

- Stores the bare data

Quick note: Every layer loses information

- Storage controller:

  Number of writes per sector, contents of defective blocks, contents of backup blocks

- Storage layer:

  Contents of faulty drive, contents of spare sectors

- File system:

  Contents of deleted files, previously used filenames, fragmentation pattern

File system forensics

- **Main goal:** Determine what is *currently* stored in a filesystem
- Determine how and when that state was created
- Possibly determine *previous* contents of the storage
- Make sure this state was not fabricated
- Look for anomalies in the system
- Under *no circumstances*: MODIFY ANYTHING!

Really forbidden!!!

How NOT to do it

- ~~Boot the machine, copy all files to USB disk~~
- ~~Put the disk(s) into second PC, mount filesystem, copy files~~
- ~~Put the disk(s) into second PC (running “normal“ OS), mount, take image~~

How to DO it: Common practice

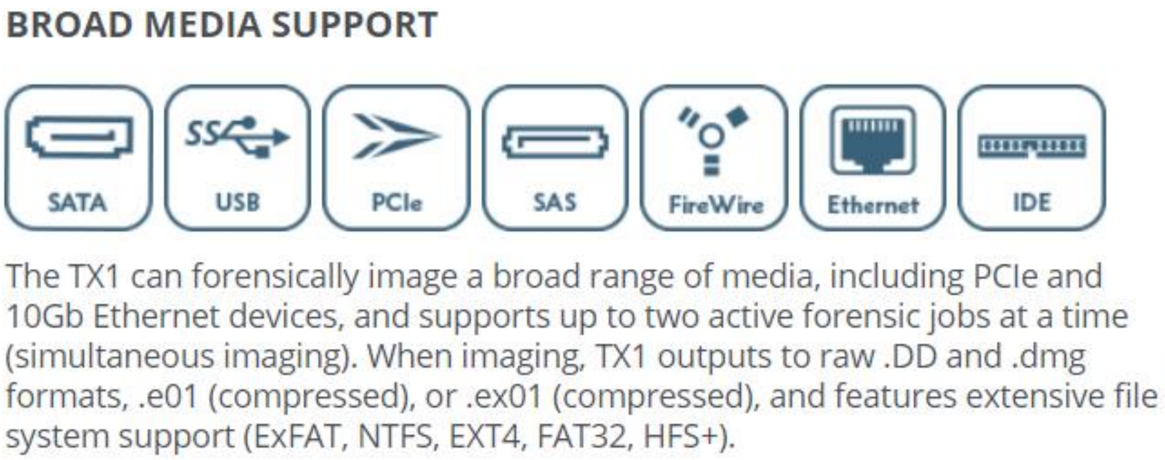
- Document state of system, peripherals, serial numbers, location, date/time, internal connection of disks
- Shut the powered system down
- Remove drives
- Use a hardware write blocker
- Use second system to image with dedicated software
- Take an image of each disk separately
- Store hash of the image, write it down
- Run your full analysis only on the images you took, reassemble RAID or similar storage in software

Read-only access to a filesystem

- Just don’t modify something willingly
  (Your operating system will still modify the access time of the files)
- Mount the filesystem read-only
  (might still trigger something like auto-defragmentation)
- Use write-barriers on the operating system
  (The OS might still change power-management settings on the storage)
- Add a hardware write blocker
  (A hardware RAID controller after the blocker might start a rebuild)
- Bypass hardware RAID controller, take images
  (The disk itself might still start to shuffle around free/unused blocks)
- Bypass storage controller on the hardware
  (That‘s pretty good, but often difficult)

Write blockers

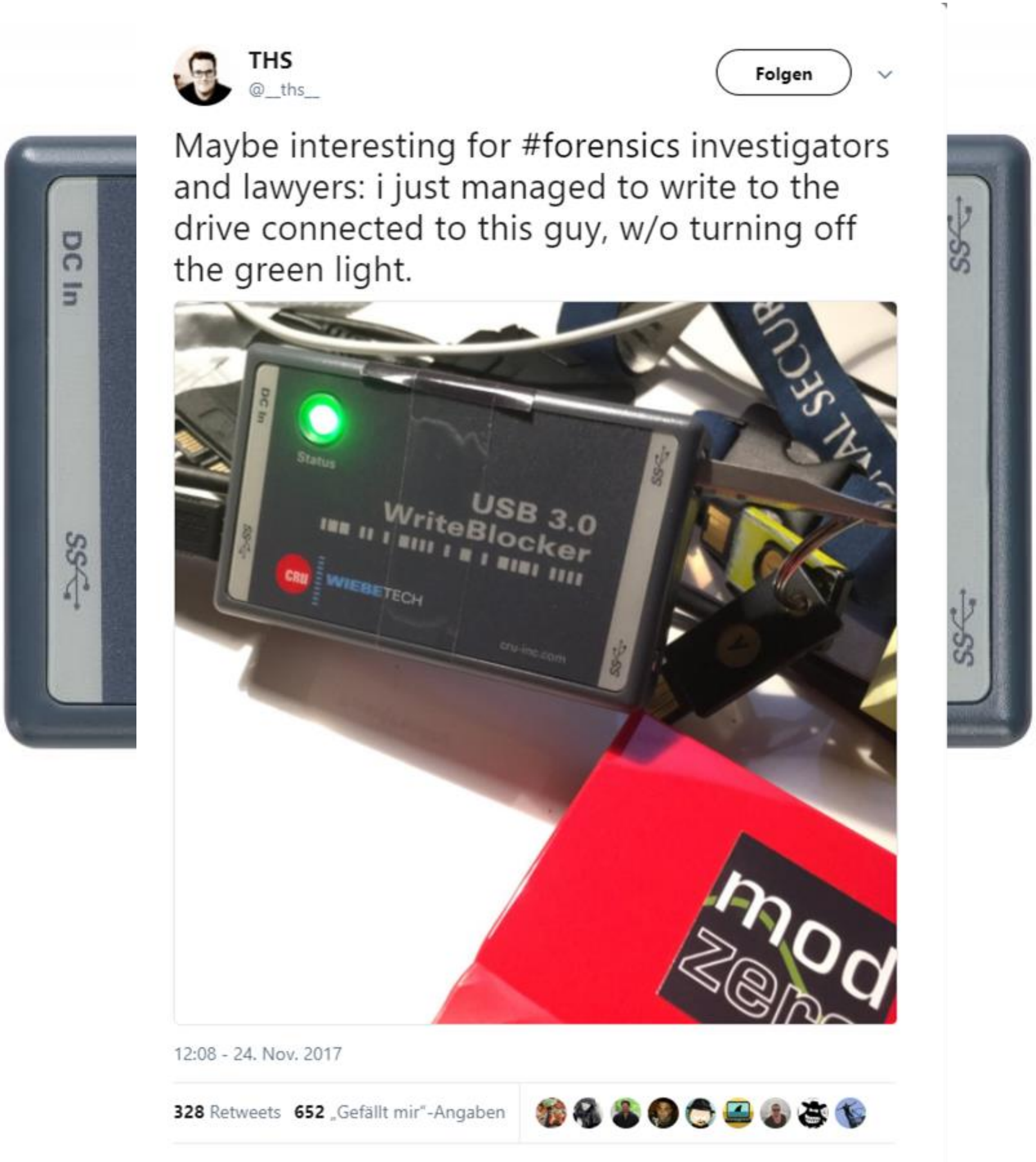
- Prevents writes to a medium being imaged by filtering respective commands
- Often combined with imaging options
- Example: Tableau TX1 (https://www.guidancesoftware.com/tableau/hardware/tx1)



BROAD MEDIA SUPPORT

The TX1 can forensically image a broad range of media, including PCIe and 10Gb Ethernet devices, and supports up to two active forensic jobs at a time (simultaneous imaging). When imaging, TX1 outputs to raw .DD and .dmg formats, .e01 (compressed), or .ex01 (compressed), and features extensive file system support (ExFAT, NTFS, EXT4, FAT32, HFS+).

A word of warning



THS
@_ths_

Maybe interesting for #forensics investigators and lawyers: i just managed to write to the drive connected to this guy, w/o turning off the green light.

12:08 - 24. Nov. 2017

328 Retweets 652 „Gefällt mir"-Angaben

A quick example ...

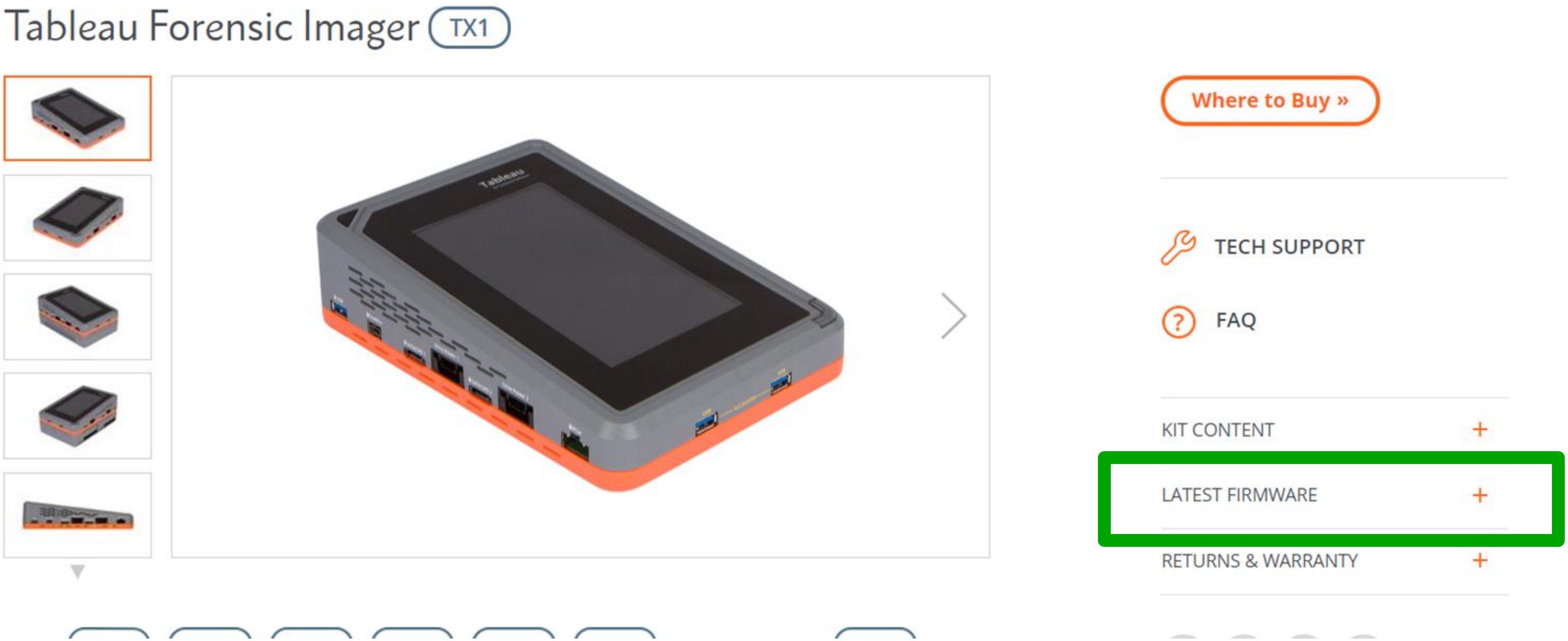


Tableau Forensic Imager TX1

A quick example ...

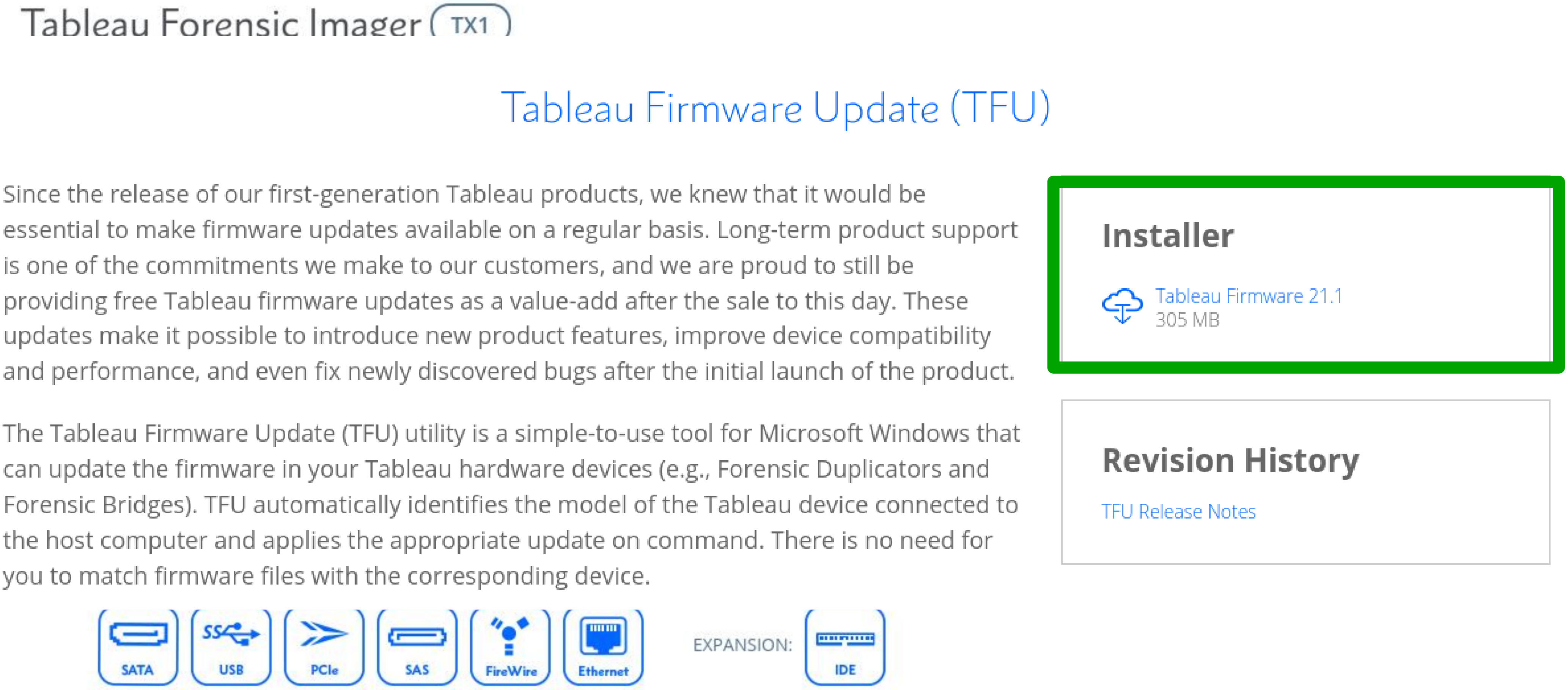


Tableau Forensic Imager (TX1)

Tableau Firmware Update (TFU)

Since the release of our first-generation Tableau products, we knew that it would be essential to make firmware updates available on a regular basis. Long-term product support is one of the commitments we make to our customers, and we are proud to still be providing free Tableau firmware updates as a value-add after the sale to this day. These updates make it possible to introduce new product features, improve device compatibility and performance, and even fix newly discovered bugs after the initial launch of the product.

The Tableau Firmware Update (TFU) utility is a simple-to-use tool for Microsoft Windows that can update the firmware in your Tableau hardware devices (e.g., Forensic Duplicators and Forensic Bridges). TFU automatically identifies the model of the Tableau device connected to the host computer and applies the appropriate update on command. There is no need for you to match firmware files with the corresponding device.

Installer

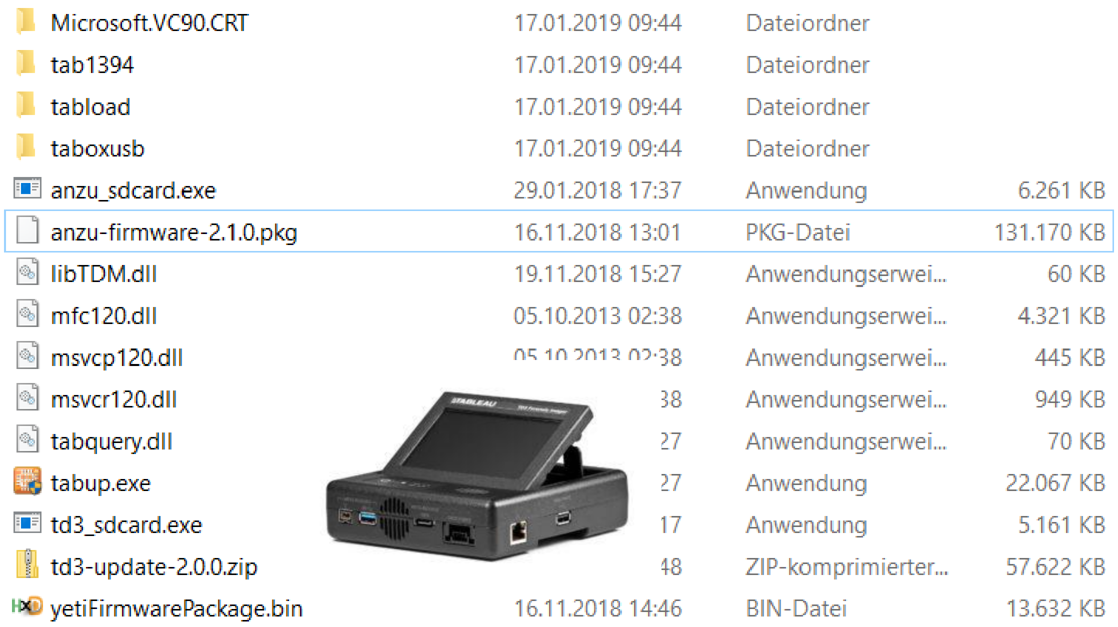
Tableau Firmware 21.1
305 MB

Revision History

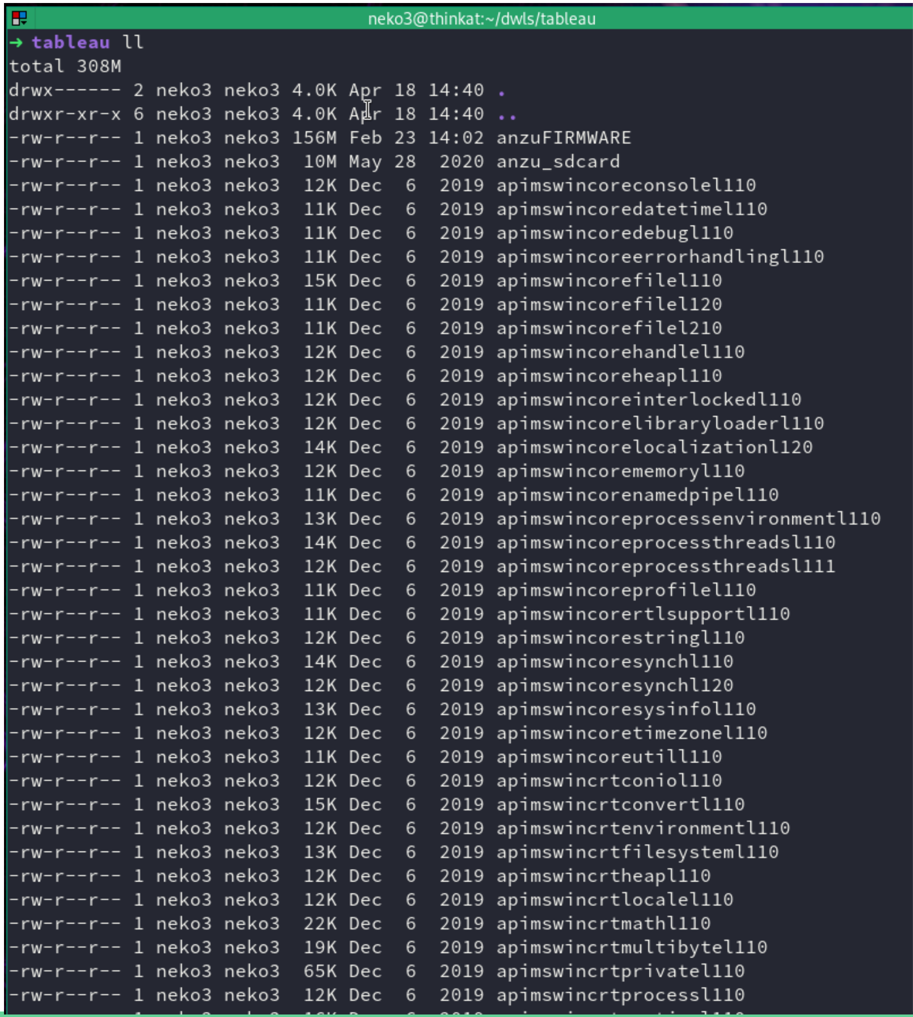
TFU Release Notes

A quick example ...

2019 vs 2021



Microsoft.VC90.CRT	17.01.2019 09:44	Dateiordner	
tab1394	17.01.2019 09:44	Dateiordner	
tabload	17.01.2019 09:44	Dateiordner	
taboxusb	17.01.2019 09:44	Dateiordner	
anzu_sdcard.exe	29.01.2018 17:37	Anwendung	6.261 KB
anzu-firmware-2.1.0.pkg	16.11.2018 13:01	PKG-Datei	131.170 KB
libTDM.dll	19.11.2018 15:27	Anwendungserwei...	60 KB
mfc120.dll	05.10.2013 02:38	Anwendungserwei...	4.321 KB
msvcp120.dll	05.10.2013 02:38	Anwendungserwei...	445 KB
msvcr120.dll	38	Anwendungserwei...	949 KB
tabquery.dll	27	Anwendungserwei...	70 KB
tabup.exe	27	Anwendung	22.067 KB
td3_sdcard.exe	17	Anwendung	5.161 KB
td3-update-2.0.0.zip	48	ZIP-komprimierter...	57.622 KB
yetiFirmwarePackage.bin	16.11.2018 14:46	BIN-Datei	13.632 KB

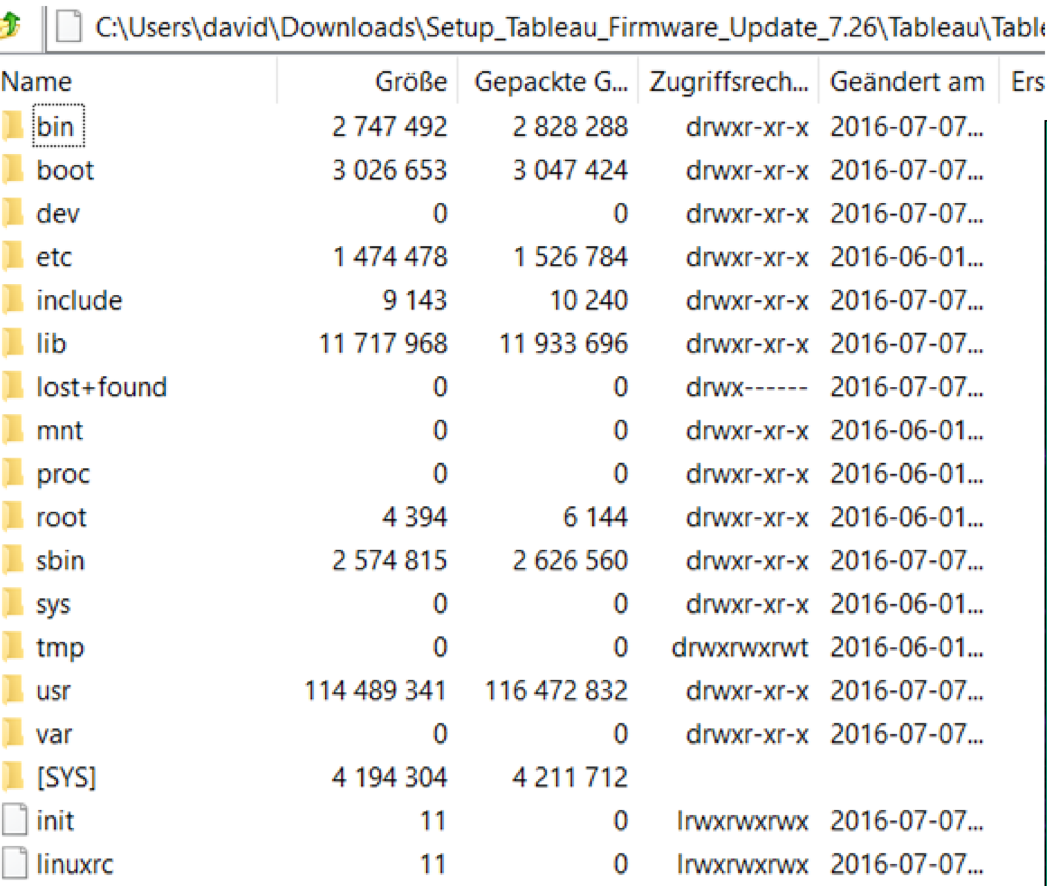


neko3@thinkat:~/dwls/tableau

```
→ tableau ll
total 308M
drwx------ 2 neko3 neko3 4.0K Apr 18 14:40 .
drwxr-xr-x 6 neko3 neko3 4.0K Apr 18 14:40 ..
-rw-r--r-- 1 neko3 neko3 156M Feb 23 14:02 anzuFIRMWARE
-rw-r--r-- 1 neko3 neko3  10M May 28  2020 anzu_sdcard
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoreconsolel110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincoredatetimel110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincoredebugl110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincoreerrorhandlingl110
-rw-r--r-- 1 neko3 neko3  15K Dec  6  2019 apimswincorefilel110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincorefilel120
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincorefilel210
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincorehandlel110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoreheapl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoreinterlockedl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincorelibraryloaderl110
-rw-r--r-- 1 neko3 neko3  14K Dec  6  2019 apimswincorelocalizationl120
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincorememoryl110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincorenamedpipel110
-rw-r--r-- 1 neko3 neko3  13K Dec  6  2019 apimswincoreprocessenvironmentl110
-rw-r--r-- 1 neko3 neko3  14K Dec  6  2019 apimswincoreprocessthreadsl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoreprocessthreadsl111
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincoreprofilel110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincorertlsupportl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincorestringl110
-rw-r--r-- 1 neko3 neko3  14K Dec  6  2019 apimswincoresynchl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoresynchl120
-rw-r--r-- 1 neko3 neko3  13K Dec  6  2019 apimswincoresysinfol110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincoretimezonel110
-rw-r--r-- 1 neko3 neko3  11K Dec  6  2019 apimswincoreutill110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincrtconiol110
-rw-r--r-- 1 neko3 neko3  15K Dec  6  2019 apimswincrtconvertl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincrtenvironmentl110
-rw-r--r-- 1 neko3 neko3  13K Dec  6  2019 apimswincrtfilesysteml110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincrtheapl110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincrtlocalel110
-rw-r--r-- 1 neko3 neko3  22K Dec  6  2019 apimswincrtmathl110
-rw-r--r-- 1 neko3 neko3  19K Dec  6  2019 apimswincrtmultibytel110
-rw-r--r-- 1 neko3 neko3  65K Dec  6  2019 apimswincrtprivatel110
-rw-r--r-- 1 neko3 neko3  12K Dec  6  2019 apimswincrtprocessl110
```

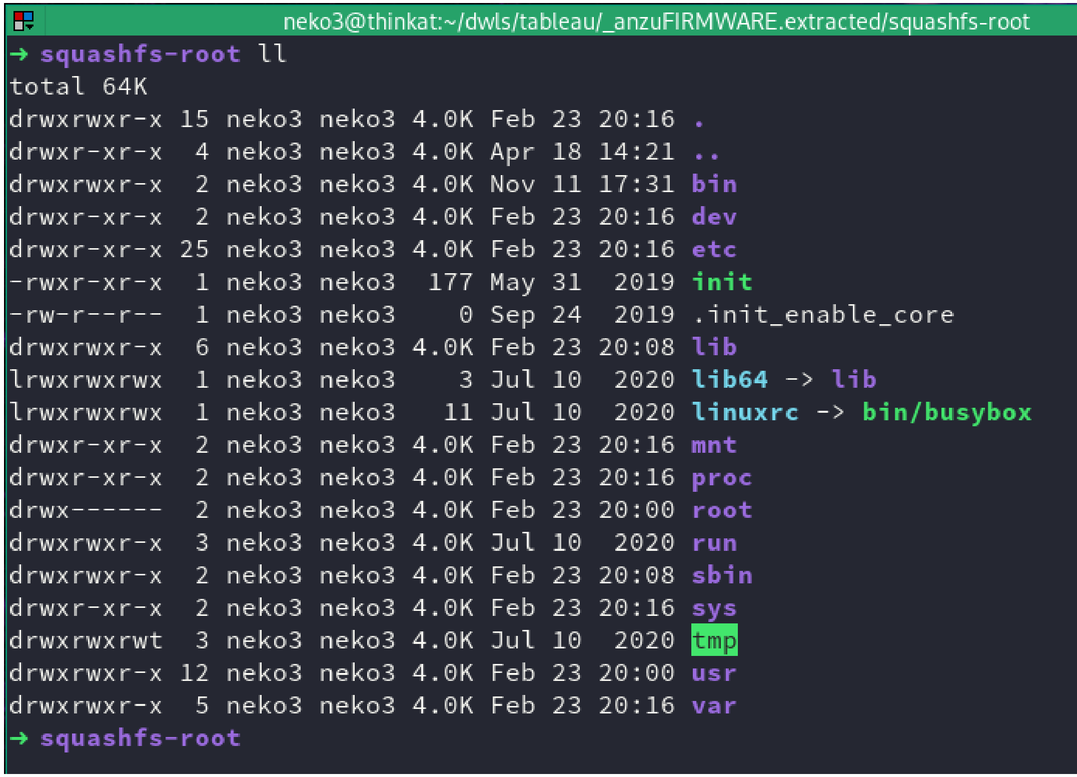
A quick example ...

2019 vs 2021



C:\Users\david\Downloads\Setup_Tableau_Firmware_Update_7.26\Tableau\Table

Name	Größe	Gepackte G...	Zugriffsrech...	Geändert am	Ers
bin	2 747 492	2 828 288	drwxr-xr-x	2016-07-07...	
boot	3 026 653	3 047 424	drwxr-xr-x	2016-07-07...	
dev	0	0	drwxr-xr-x	2016-07-07...	
etc	1 474 478	1 526 784	drwxr-xr-x	2016-06-01...	
include	9 143	10 240	drwxr-xr-x	2016-07-07...	
lib	11 717 968	11 933 696	drwxr-xr-x	2016-07-07...	
lost+found	0	0	drwx------	2016-07-07...	
mnt	0	0	drwxr-xr-x	2016-06-01...	
proc	0	0	drwxr-xr-x	2016-06-01...	
root	4 394	6 144	drwxr-xr-x	2016-06-01...	
sbin	2 574 815	2 626 560	drwxr-xr-x	2016-07-07...	
sys	0	0	drwxr-xr-x	2016-06-01...	
tmp	0	0	drwxrwxrwt	2016-06-01...	
usr	114 489 341	116 472 832	drwxr-xr-x	2016-07-07...	
var	0	0	drwxr-xr-x	2016-07-07...	
[SYS]	4 194 304	4 211 712			
init	11	0	lrwxrwxrwx	2016-07-07...	
linuxrc	11	0	lrwxrwxrwx	2016-07-07...	

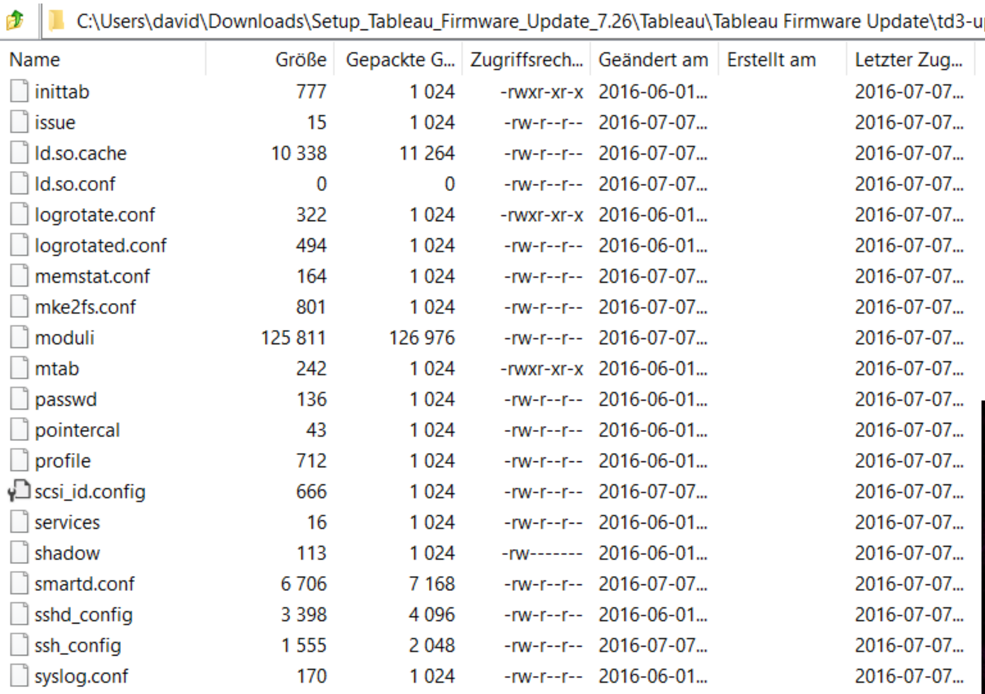


neko3@thinkat:~/dwls/tableau/_anzuFIRMWARE.extracted/squashfs-root

```
→ squashfs-root ll
total 64K
drwxrwxr-x 15 neko3 neko3 4.0K Feb 23 20:16 .
drwxr-xr-x  4 neko3 neko3 4.0K Apr 18 14:21 ..
drwxrwxr-x  2 neko3 neko3 4.0K Nov 11 17:31 bin
drwxr-xr-x  2 neko3 neko3 4.0K Feb 23 20:16 dev
drwxr-xr-x 25 neko3 neko3 4.0K Feb 23 20:16 etc
-rwxr-xr-x  1 neko3 neko3  177 May 31  2019 init
-rw-r--r--  1 neko3 neko3    0 Sep 24  2019 .init_enable_core
drwxrwxr-x  6 neko3 neko3 4.0K Feb 23 20:08 lib
lrwxrwxrwx  1 neko3 neko3    3 Jul 10  2020 lib64 -> lib
lrwxrwxrwx  1 neko3 neko3   11 Jul 10  2020 linuxrc -> bin/busybox
drwxr-xr-x  2 neko3 neko3 4.0K Feb 23 20:16 mnt
drwxr-xr-x  2 neko3 neko3 4.0K Feb 23 20:16 proc
drwx------  2 neko3 neko3 4.0K Feb 23 20:00 root
drwxrwxr-x  3 neko3 neko3 4.0K Jul 10  2020 run
drwxrwxr-x  2 neko3 neko3 4.0K Feb 23 20:08 sbin
drwxr-xr-x  2 neko3 neko3 4.0K Feb 23 20:16 sys
drwxrwxrwt  3 neko3 neko3 4.0K Jul 10  2020 tmp
drwxrwxr-x 12 neko3 neko3 4.0K Feb 23 20:00 usr
drwxrwxr-x  5 neko3 neko3 4.0K Feb 23 20:16 var
→ squashfs-root
```

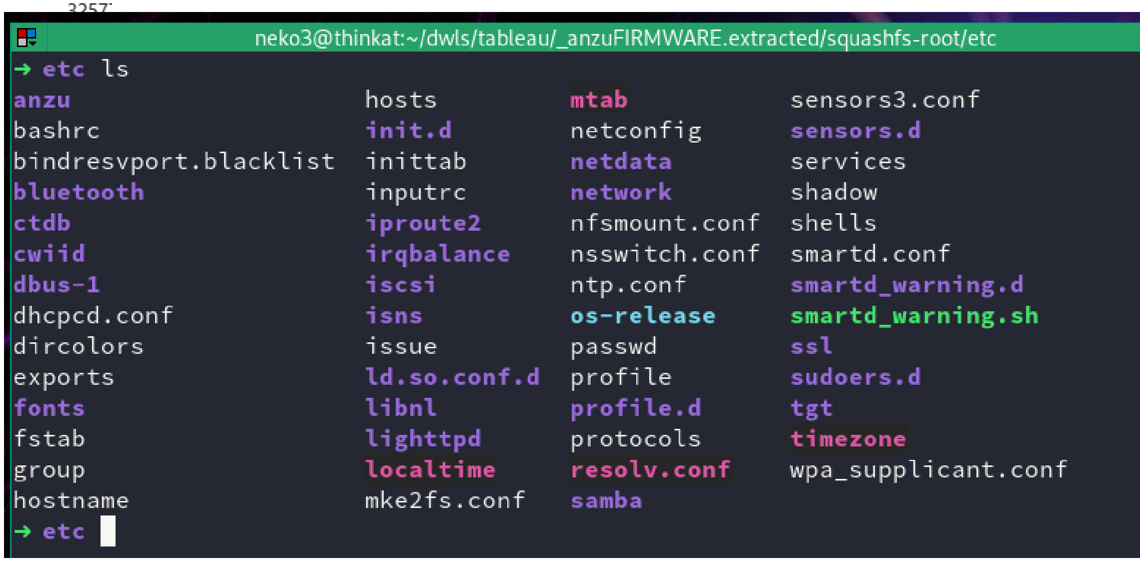
A quick example ...

2019 vs 2021



C:\Users\david\Downloads\Setup_Tableau_Firmware_Update_7.26\Tableau\Tableau Firmware Update\td3-update-2.0.0\

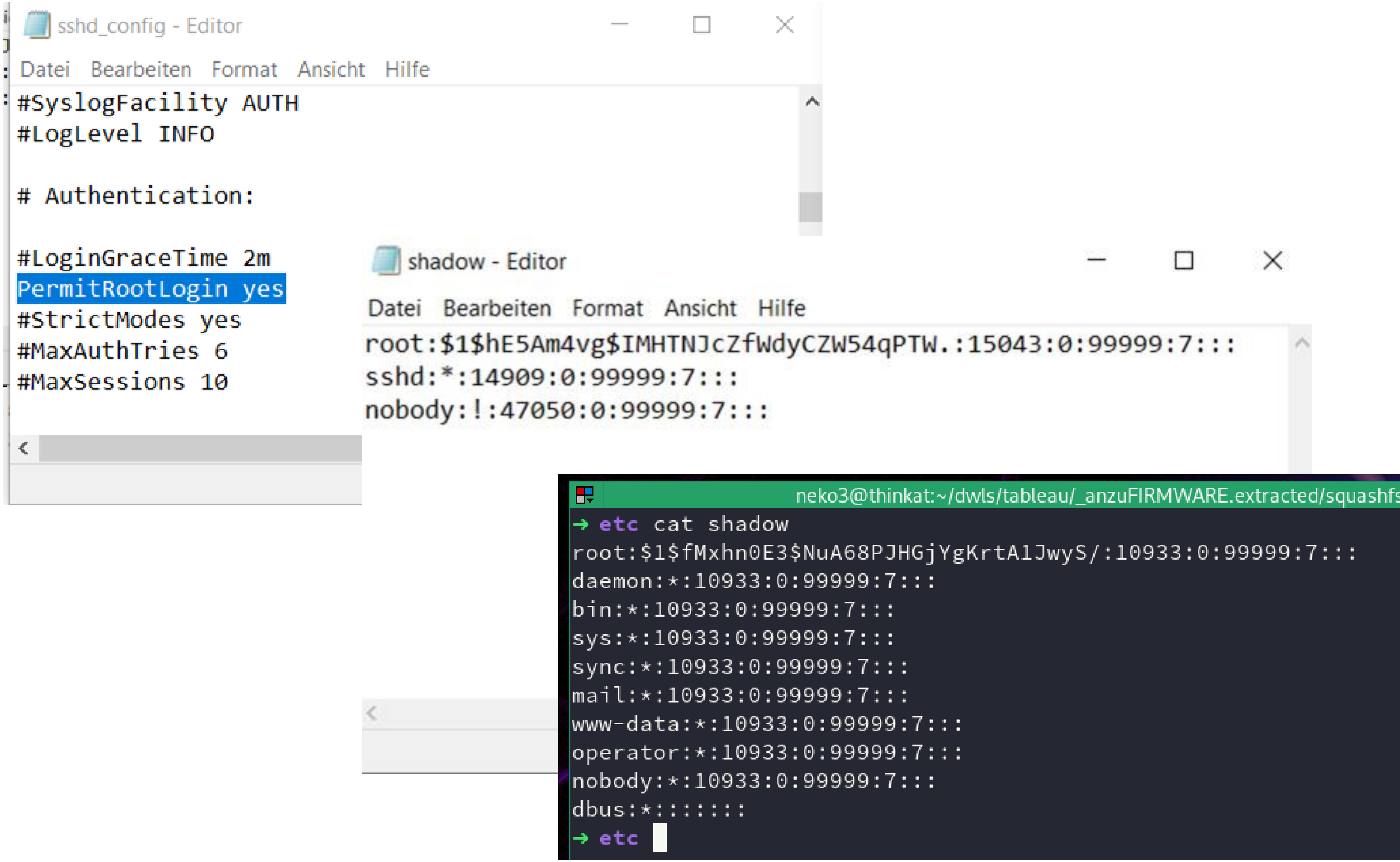
Name	Größe	Gepackte G...	Zugriffsrech...	Geändert am	Erstellt am	Letzter Zug...	iNode
inittab	777	1 024	-rwxr-xr-x	2016-06-01...		2016-07-07...	32536
issue	15	1 024	-rw-r--r--	2016-07-07...		2016-07-07...	32555
ld.so.cache	10 338	11 264	-rw-r--r--	2016-07-07...		2016-07-07...	32583
ld.so.conf	0	0	-rw-r--r--	2016-07-07...		2016-07-07...	32605
logrotate.conf	322	1 024	-rwxr-xr-x	2016-06-01...		2016-07-07...	32529
logrotated.conf	494	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	32539
memstat.conf	164	1 024	-rw-r--r--	2016-07-07...		2016-07-07...	32544
mke2fs.conf	801	1 024	-rw-r--r--	2016-07-07...		2016-07-07...	32540
moduli	125 811	126 976	-rw-r--r--	2016-07-07...		2016-07-07...	32523
mtab	242	1 024	-rwxr-xr-x	2016-06-01...		2016-07-07...	32586
passwd	136	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	3257
pointercal	43	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	
profile	712	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	
scsi_id.config	666	1 024	-rw-r--r--	2016-07-07...		2016-07-07...	
services	16	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	
shadow	113	1 024	-rw-------	2016-06-01...		2016-07-07...	
smartd.conf	6 706	7 168	-rw-r--r--	2016-07-07...		2016-07-07...	
sshd_config	3 398	4 096	-rw-r--r--	2016-06-01...		2016-07-07...	
ssh_config	1 555	2 048	-rw-r--r--	2016-07-07...		2016-07-07...	
syslog.conf	170	1 024	-rw-r--r--	2016-06-01...		2016-07-07...	



neko3@thinkat:~/dwls/tableau/_anzuFIRMWARE.extracted/squashfs-root/etc

```
→ etc ls
anzu                     hosts           mtab             sensors3.conf
bashrc                   init.d          netconfig        sensors.d
bindresvport.blacklist   inittab         netdata          services
bluetooth                inputrc         network          shadow
ctdb                     iproute2        nfsmount.conf    shells
cwiid                    irqbalance      nsswitch.conf    smartd.conf
dbus-1                   iscsi           ntp.conf         smartd_warning.d
dhcpcd.conf              isns            os-release       smartd_warning.sh
dircolors                issue           passwd           ssl
exports                  ld.so.conf.d    profile          sudoers.d
fonts                    libnl           profile.d        tgt
fstab                    lighttpd        protocols        timezone
group                    localtime       resolv.conf      wpa_supplicant.conf
hostname                 mke2fs.conf     samba
→ etc
```

A quick example ...

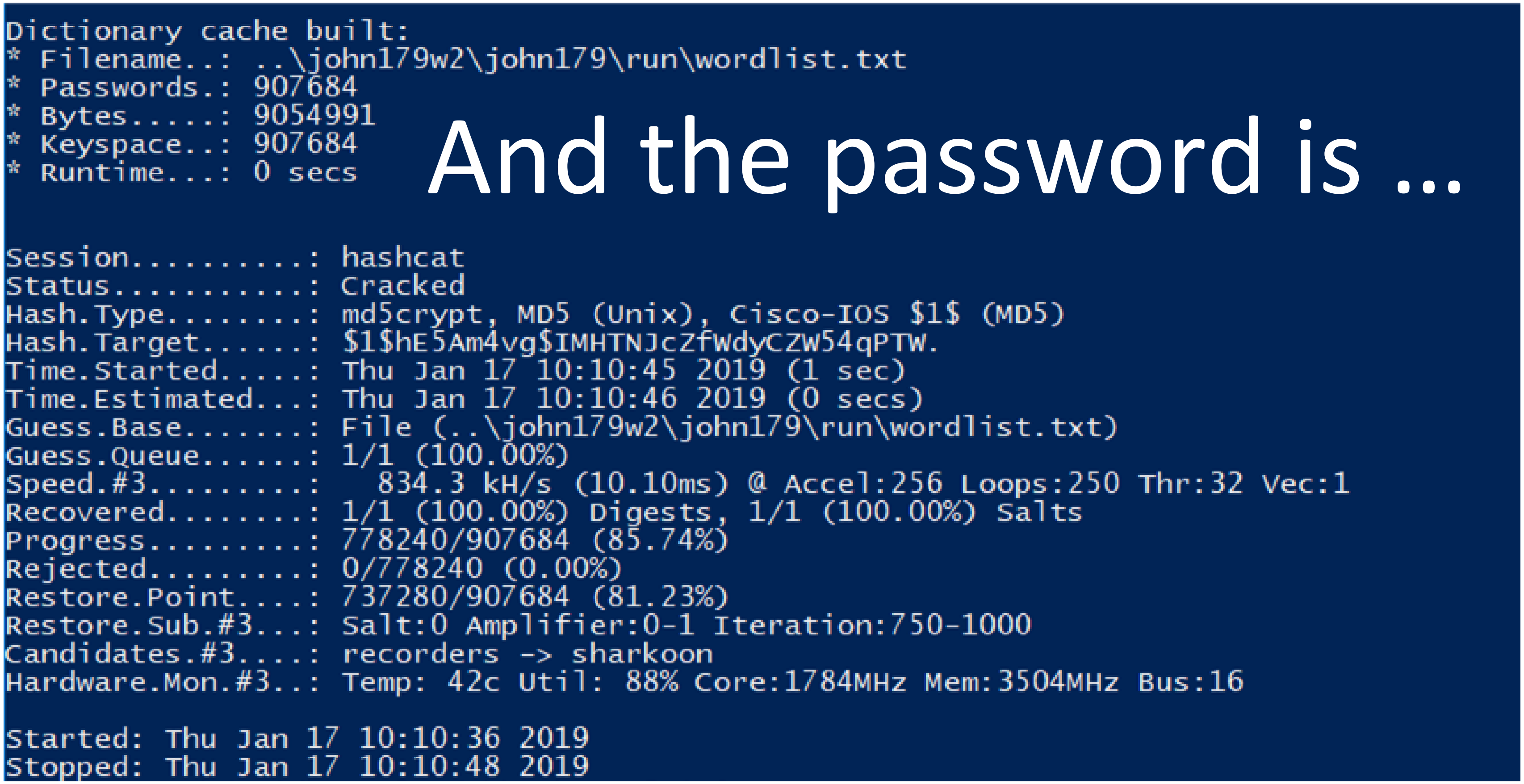

sshd_config - Editor
Datei Bearbeiten Format Ansicht Hilfe
shadow - Editor
Datei Bearbeiten Format Ansicht Hilfe
neko3@thinkat:~/dwls/tableau/_anzuFIRMWARE.extracted/squashfs

```
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

```
root:$1$hE5Am4vg$IMHTNJcZfWdyCZW54qPTW.:15043:0:99999:7:::
sshd:*:14909:0:99999:7:::
nobody:!:47050:0:99999:7:::
```

```
→ etc cat shadow
root:$1$fMxhn0E3$NuA68PJHGjYgKrtA1JwyS/:10933:0:99999:7:::
daemon:*:10933:0:99999:7:::
bin:*:10933:0:99999:7:::
sys:*:10933:0:99999:7:::
sync:*:10933:0:99999:7:::
mail:*:10933:0:99999:7:::
www-data:*:10933:0:99999:7:::
operator:*:10933:0:99999:7:::
nobody:*:10933:0:99999:7:::
dbus:*:::::::
→ etc
```

A quick example …



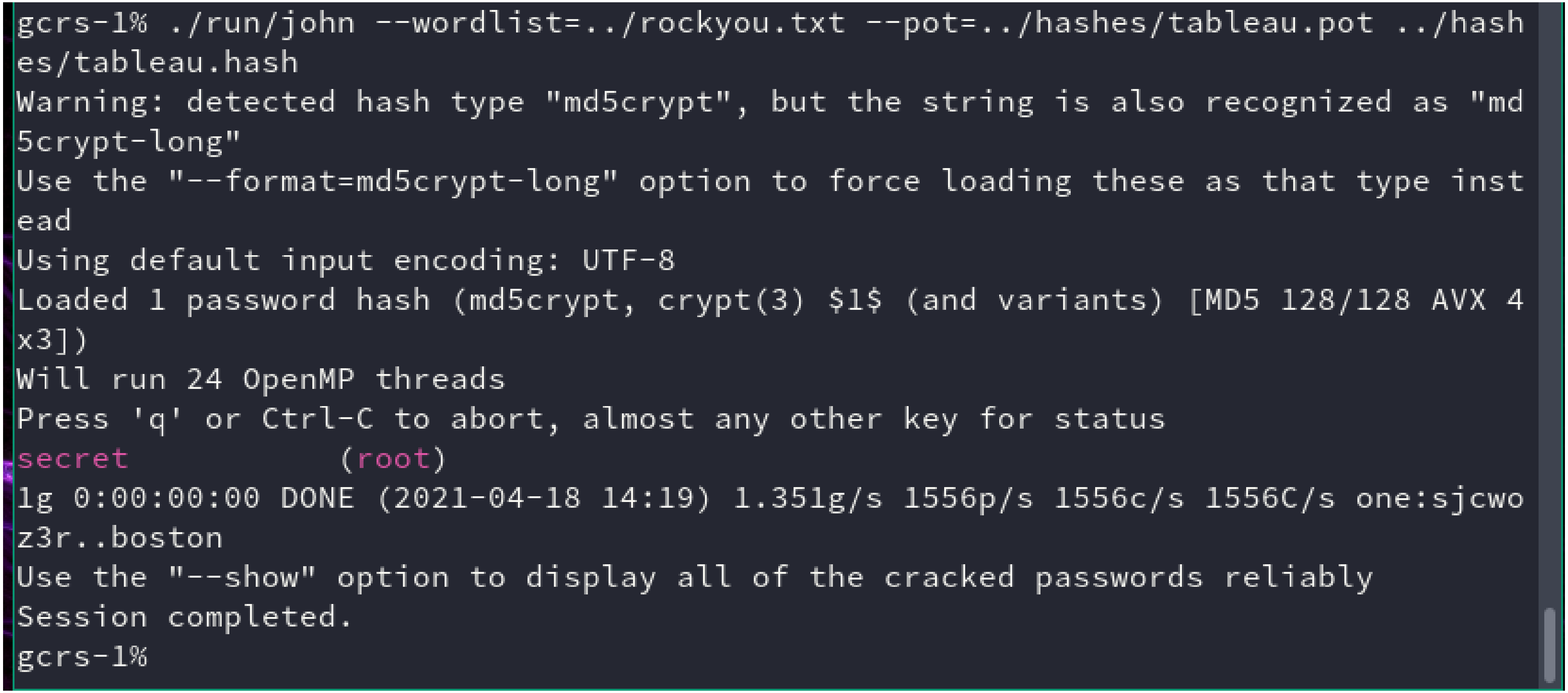
```
Dictionary cache built:
* Filename..: ..\john179w2\john179\run\wordlist.txt
* Passwords.: 907684
* Bytes.....: 9054991
* Keyspace..: 907684
* Runtime...: 0 secs


Session..........: hashcat
Status...........: Cracked
Hash.Type........: md5crypt, MD5 (Unix), Cisco-IOS $1$ (MD5)
Hash.Target......: $1$hE5Am4vg$IMHTNJcZfWdyCZW54qPTW.
Time.Started.....: Thu Jan 17 10:10:45 2019 (1 sec)
Time.Estimated...: Thu Jan 17 10:10:46 2019 (0 secs)
Guess.Base.......: File (..\john179w2\john179\run\wordlist.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#3.........:   834.3 kH/s (10.10ms) @ Accel:256 Loops:250 Thr:32 Vec:1
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 778240/907684 (85.74%)
Rejected.........: 0/778240 (0.00%)
Restore.Point....: 737280/907684 (81.23%)
Restore.Sub.#3...: Salt:0 Amplifier:0-1 Iteration:750-1000
Candidates.#3....: recorders -> sharkoon
Hardware.Mon.#3..: Temp: 42c Util: 88% Core:1784MHz Mem:3504MHz Bus:16

Started: Thu Jan 17 10:10:36 2019
Stopped: Thu Jan 17 10:10:48 2019
```

And the password is …

… secret

A quick example ...

And in 2021...



```
gcrs-1% ./run/john --wordlist=../rockyou.txt --pot=../hashes/tableau.pot ../hash
es/tableau.hash
Warning: detected hash type "md5crypt", but the string is also recognized as "md
5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type inst
ead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 AVX 4
x3])
Will run 24 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
secret           (root)
1g 0:00:00:00 DONE (2021-04-18 14:19) 1.351g/s 1556p/s 1556c/s 1556C/s one:sjcwo
z3r..boston
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
gcrs-1%
```

Analysing disk images

Analysing disk images

1. Reorder physical drives to logical groups (RAID, LVM) if necessary
2. Recover partition tables
3. Identify file systems
4. Determine content of file systems
5. Use logfiles to create a timeline of events
6. Check for possibly deleted files
7. Check for abnormalities

File forensics (overview)

- Similar approach to file system forensics
- File type can be determined by extension/metadata/content
- Find suitable viewers for files (native applications do not show everything)
- (Maybe) look for old versions of the file
- Tools often support searching for known hashes and for file contents

Possible anomalies

- Partition tables with alignment not used by the OS installer:
  Maybe the partition was fabricated with another tool
- Unusual ordering of data on the drive:
  Data may have been copied there in one go
- No fragments of old data in the free space:
  Drive maybe cleaned and cloned from other system
- Access pattern of file groups not in sync:
  Files put in browser cache, but not the cache index
- Unusual speed/times:
  Files in the download folder downloaded with the speed of a USB3 HDD while user had slow DSL connection
- Least significant bits of timer values are biased

Tools ...

"2009-07 wpy 16" by Ziko - Own work. Licensed under CC BY-SA 3.0

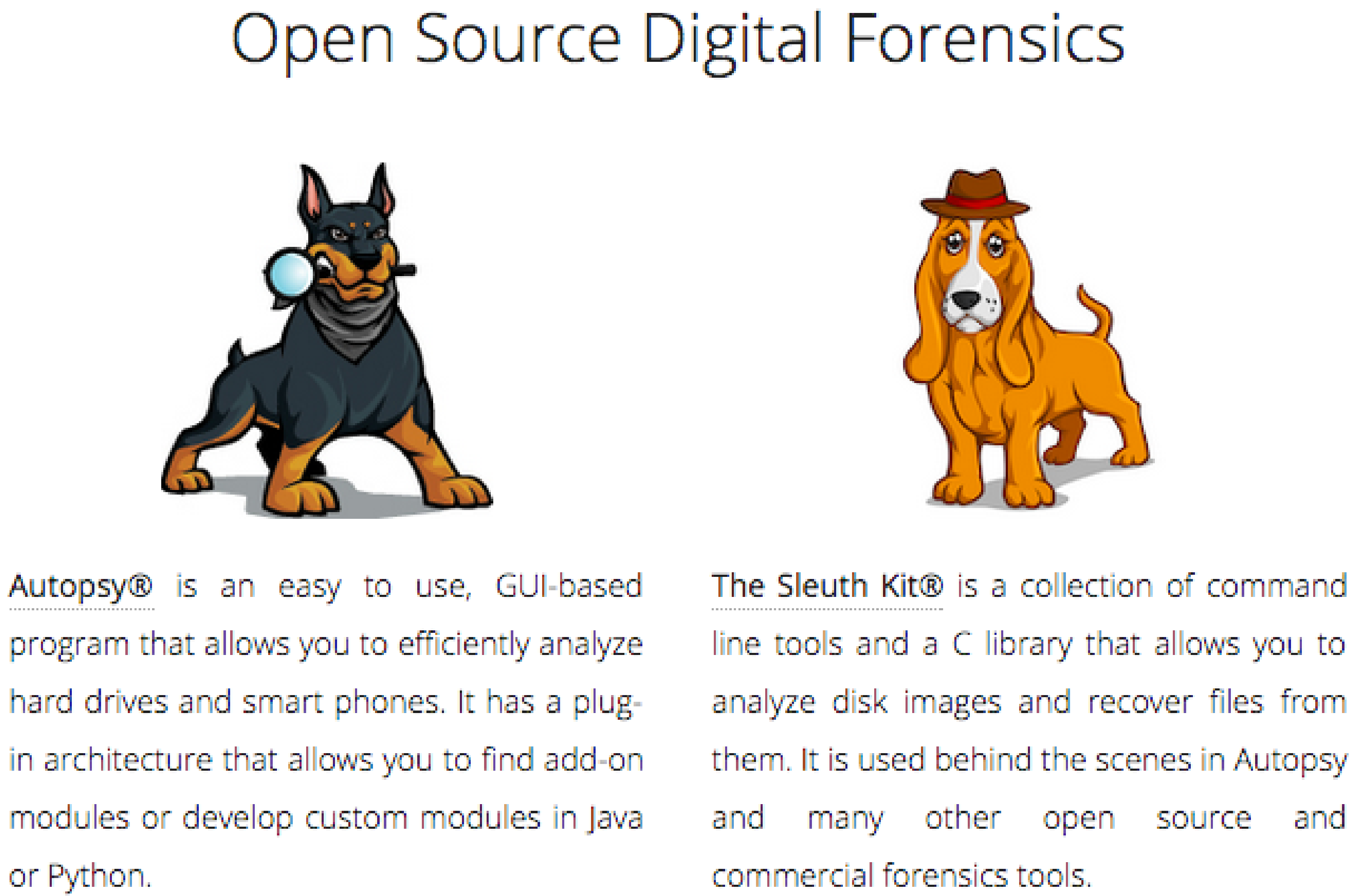
Command line tools

- `dd` **for imaging**
- `sha1sum, sha256sum` **for hashing**
- `fdisk` **for viewing partitions**

  ```
  fdisk -lu disk.img
  ```
- `mount` **for mounting (loopback)**

  ```
  mount -o loop,ro,noexec disk.img /mnt/img
  ```
- `ls` / `find` / `grep` / … **for finding files**
- **Additional tools like** `photorec`, `testdisk`, …

Analysis: Sleuth Kit / Autopsy



Open Source Digital Forensics

Autopsy® is an easy to use, GUI-based program that allows you to efficiently analyze hard drives and smart phones. It has a plug-in architecture that allows you to find add-on modules or develop custom modules in Java or Python.

The Sleuth Kit® is a collection of command line tools and a C library that allows you to analyze disk images and recover files from them. It is used behind the scenes in Autopsy and many other open source and commercial forensics tools.

Autopsy / TSK: Features

- The Sleuth Kit (TSK) http://wiki.sleuthkit.org/
  - Collection of command line tools http://wiki.sleuthkit.org/index.php?title=TSK_Tool_Overview
  - Supports wide range of filesystems: ext2/3/4, (ex)FAT, HFS, ISO 9660 (CD), NTFS, UFS 1/2, YAFFS2
  - Open-source: https://github.com/sleuthkit/sleuthkit
- Autopsy: Graphical front-end to TSK
- Both installed in Forensics VM
- Alternative: https://www.sans.org/tools/sift-workstation/

Summary

- Disk forensics is well supported by tools nowadays (command line and GUI)
- We can only cover a subset of all available tools and focus on open source software
- Many commercial tools (e.g. EnCase, Belkasoft Evidence Centre) used in practice
- **A good forensics tool never modifies the image under any circumstances**

Next part: Log File Forensics