Forensics MP Checkpoint 1

Pubali Datta
University of Illinois
CS 461 / ECE 422 - Fall 2019

* Some slides borrowed from CS 461 Spring 2019

Announcements

- MP5 Checkpoint 1 will be released at 6pm today.
  - Due on November 18, 6pm.
- Checkpoint 2 will be released next week due to some technical issues.
  - Due on December 4, 6pm.

Educational Objectives

- Understand the difference between live and dead analysis
- Learn about common artifacts
- Understand disk partitioning
- Introduction to Unix filesystem
- How to ensure evidence integrity
- How to analyze disk images

Live Analysis

- Live analysis – look at evidence on live computer
  - Pro: Volatile information can be retrieved and analyzed.
  - Pro: Evidence can be retrieved in unlocked state if system is password-protected or encrypted.
  - Con: There is a high risk of damaging evidence.
  - Con: System's behaviors cannot be completely controlled.
  - *Fun fact: Cold boot attack* https://leahycenterblog.champlain.edu/2013/08/09/capturing-ram-locked-computer/

Dead Analysis

- Dead analysis – look at evidence on disks with systems switched off
  - Pro: There is small to no risk of damaging evidence.
  - Pro: Analysis is not time-constrained and can be repeated without risking damage.
  - Pro: Hidden or deleted files can be identified and recovered.
  - Con: Evidence cannot be retrieved if system is encrypted.
  - Con: Volatile information cannot be retrieved.

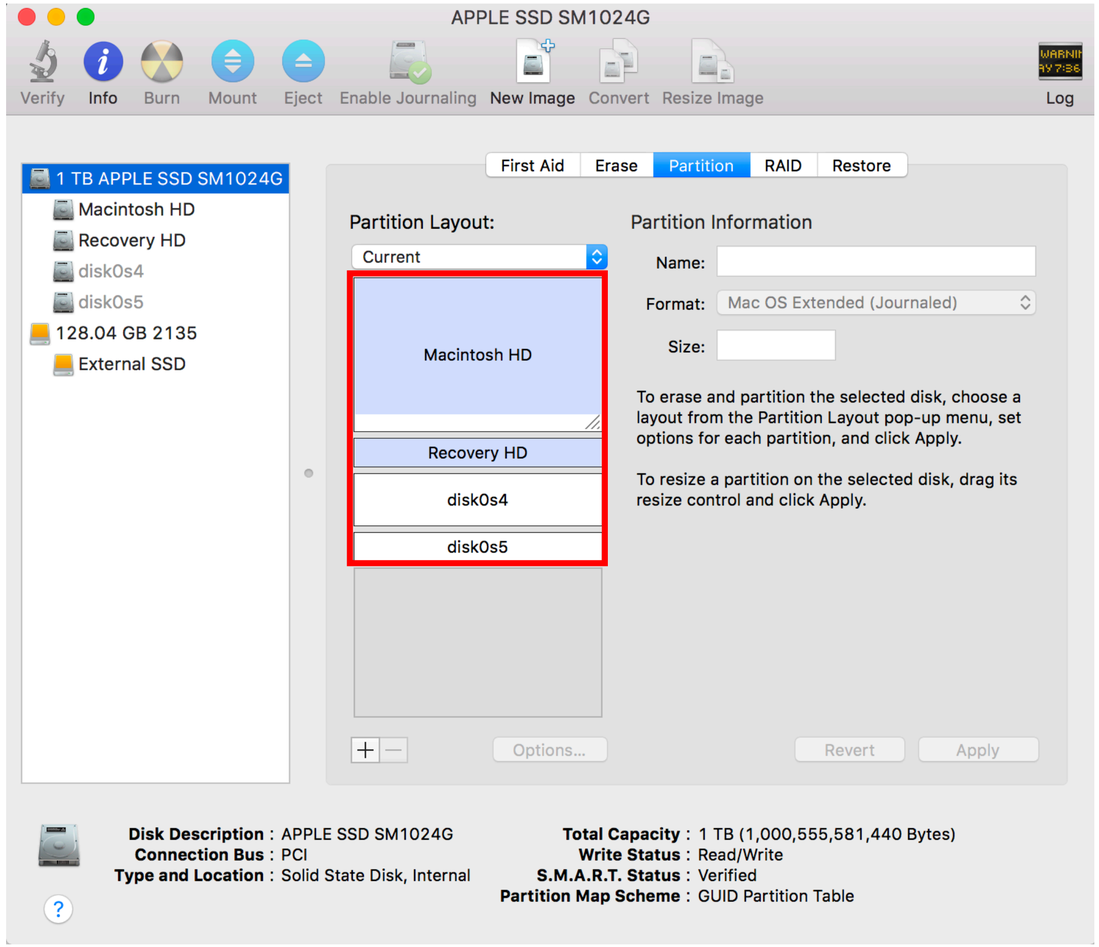
System Artifacts

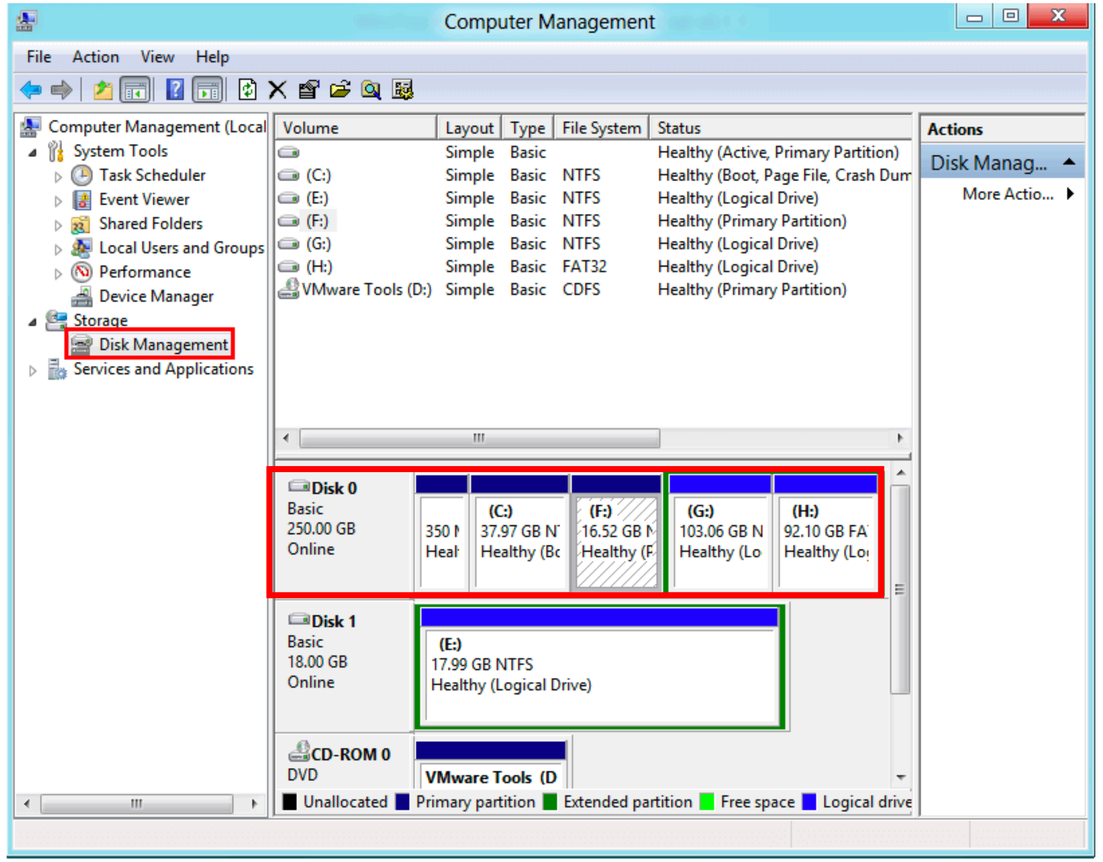
- Disk partitions – hidden partitions, boot order
- Filesystem – file attributes
- OS – system logs, system configuration files, temp files

Disk Partitions

- A *disk partition* is a way of dividing up a hard disk.
- Different partitions may contain different operating systems
- Every disk has a **partition table** that stores metadata about the partitions.

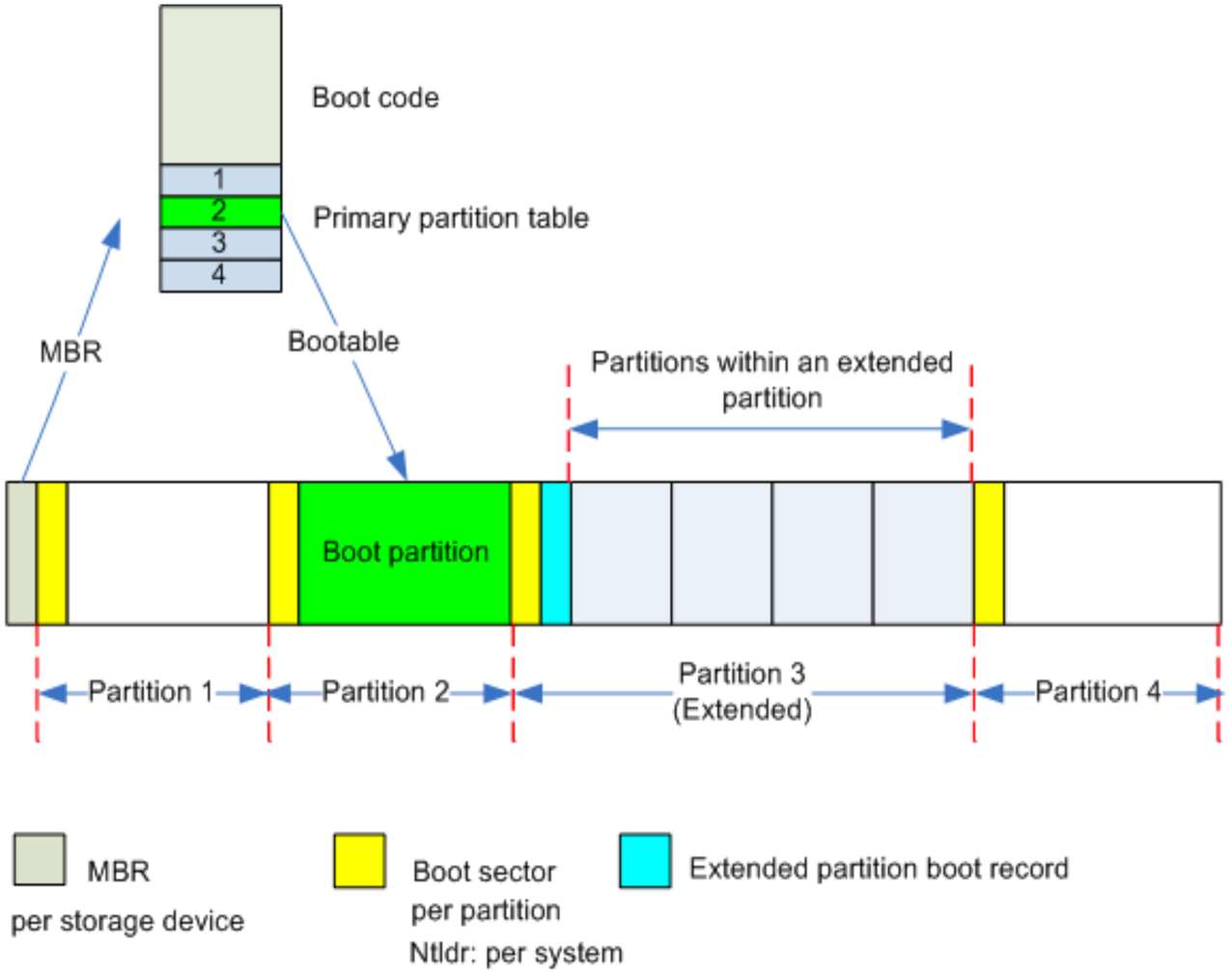
Disk Partition Examples





MBR and Boot Sectors

- Master Boot Record (MBR) in the first sector of disk
- Boot sector in the first sector of partition



Partition Table Example (`fdisk`)

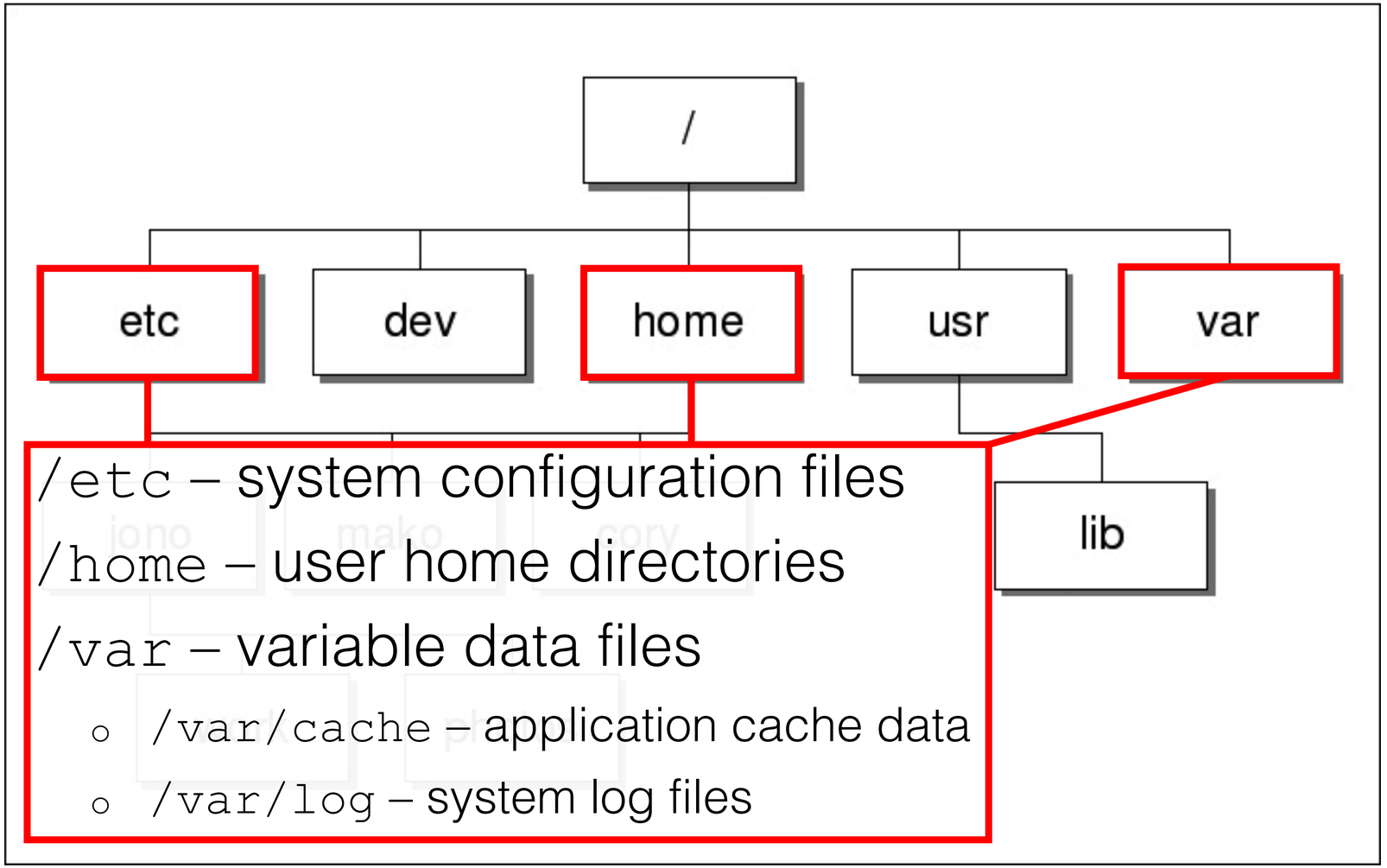
```
Disk /dev/sda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xf6db58cc

Device     Boot     Start       End   Sectors  Size Id Type
/dev/sda1  *         2048  15204351  15202304  7.3G 83 Linux
/dev/sda2        15206398  16775167   1568770  766M  5 Extended
/dev/sda5        15206400  16775167   1568768  766M 82 Linux swap / Solaris
(END)
```


Unix File System

- Conventional directory layout
  - https://en.wikipedia.org/wiki/Unix_filesystem#Conventional_directory_layout
  - https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard

Unix Filesystem Directory Layout

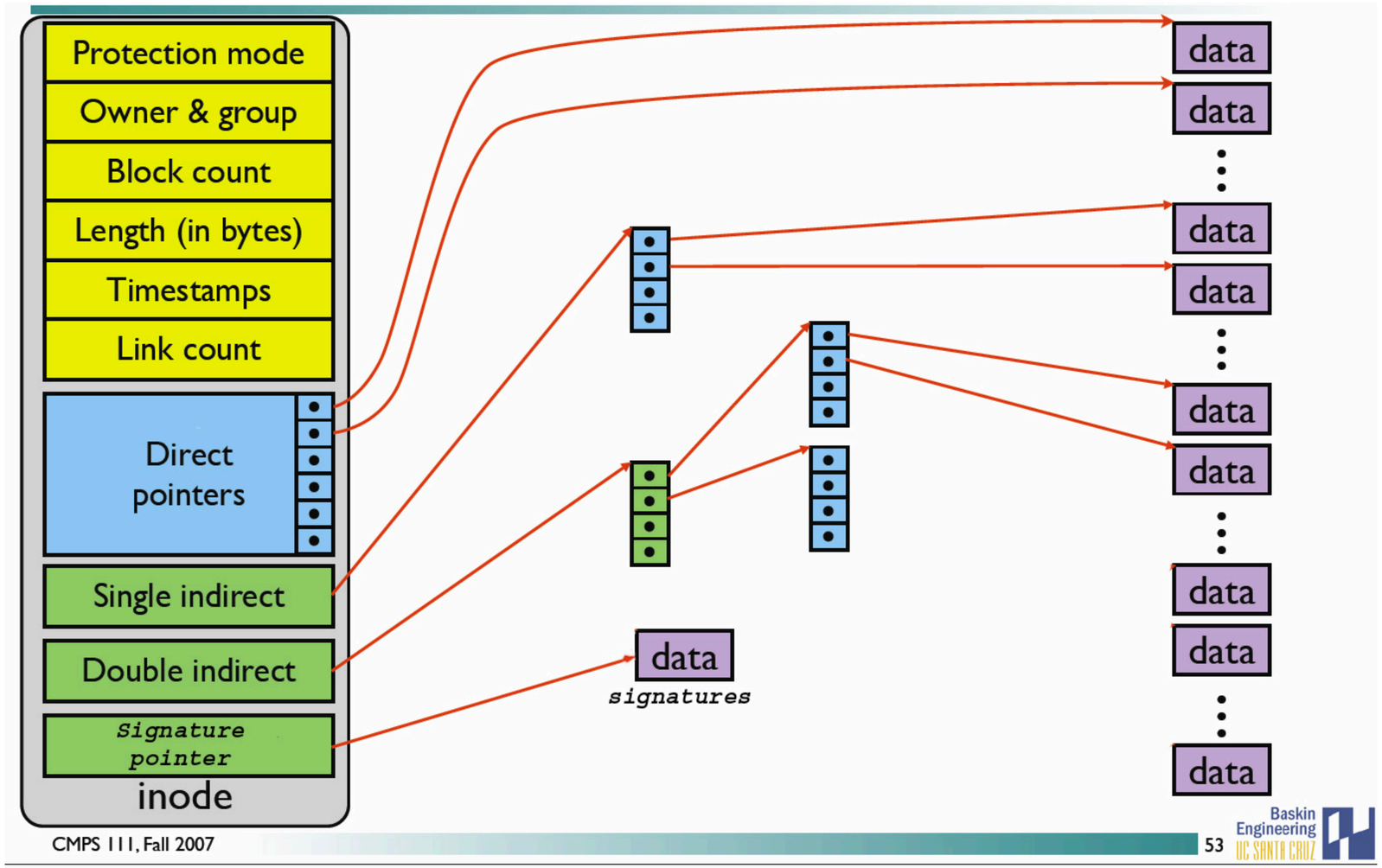


Unix File System

- Conventional directory layout
  - https://en.wikipedia.org/wiki/Unix_filesystem#Conventional_directory_layout
  - https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard

- Inode data structure – stores attributes and block location of data
  - https://en.wikipedia.org/wiki/Inode
  - https://cs241.cs.illinois.edu/coursebook/Filesystems#storing-data-on-disk

Inode Data Structure



Inode Content Example (`stat`)

```
  File: Forensics-cp1.pptx
  Size: 19237497        Blocks: 37576      IO Block: 4096   regular file
Device: 801h/2049d      Inode: 3277781     Links: 1
Access: (0644/-rw-r--r--)  Uid: ( 1001/forensics)   Gid: ( 1001/forensics)
Access: 2019-04-08 23:26:26.377300000 -0500
Modify: 2019-04-08 23:26:12.001939000 -0500
Change: 2019-04-08 23:26:26.485296059 -0500
 Birth: -
(END)
```


Common User Artifacts

- Internet – browsing history
  - Ex) Firefox stores the history in SQLite (see https://davidkoepi.wordpress.com/2010/11/27/firefoxforensics/)
- Multimedia – photos, video, audio
  - Ex) Cached image thumbnails
- Documents – Office, PDFs, RTF, XML
- Other applications – instant messaging

File Metadata

- Metadata – description and context of data
- Two types
  - System metadata – file attributes stored in inode
    - Ex) Permissions, owner, size, timestamps
  - Application metadata – application-specific information stored in data
    - Ex) document title, creator, geolocation data

System Metadata Example (`ls`)

```
total 8.1G
drwxr-xr-x 3 forensics forensics 4.0K Apr  8 14:13 Desktop
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Documents
drwxr-xr-x 2 forensics forensics 4.0K Apr  8 13:15 Downloads
-rw-r--r-- 1 forensics forensics  19M Apr  8 23:26 Forensics-cp1.pptx
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Music
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Pictures
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Public
-rw-r--r-- 1 forensics forensics  483 Apr  8 15:44 SHA256SUMS-disks.txt
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Templates
-rw-r--r-- 1 forensics forensics 8.0G Apr  7 18:30 victim.raw
drwxr-xr-x 2 forensics forensics 4.0K Apr  7 21:15 Videos
(END)
```


App. Metadata Example (`exiftool`)

```
MIME Type                       : application/vnd.openxmlformats
-officedocument.presentationml.presentation
Zip Required Version            : 20
Zip Bit Flag                    : 0x0006
Zip Compression                 : Deflated
Zip Modify Date                 : 1980:01:01 00:00:00
Zip CRC                         : 0x587b3fe9
Zip Compressed Size             : 613
Zip Uncompressed Size           : 7893
Zip File Name                   : [Content_Types].xml
Title                           : Forensics MP Checkpoint 1
Last Modified By                : Kim, Simon
Revision Number                 : 55
Modify Date                     : 2019:04:09 04:26:11Z
Total Edit Time                 : 10.2 hours
Words                           : 810
Application                     : Microsoft Macintosh PowerPoint
Presentation Format             : Custom
Paragraphs                      : 147
Slides                          : 26
```


Forensics MP Intro

- You take the role of a forensics investigator examining computers involved in a murder case.
  - Checkpoint 1 – examine victim's computer
  - Checkpoint 2 – examine suspects' computer

Recommended Tools

- The Sleuth Kit (TSK)
  - `fls`: lists allocated and deleted file names in a directory
  - `icat`: extracts content of a file, which is specified by its inode number
  - `istat`: displays details about a given inode structure
  - `mactime`: takes input from the `fls` to create a timeline of file activity
  - For details, see https://wiki.sleuthkit.org/index.php?title=Help_Documents .
- Metadata manipulator – `exiftool`
- File-carving/recovery tools
- Password cracking tools
- Full list provided in the assignment handout

Checkpoint 1 Objectives

- Understand high-level directory structure of Unix systems.
- Identify system configurations.
- Identify application artifacts.
- Understand file attributes.
- Identify suspicious activities and network events from system logs.

Setting Up Environment

- Live analysis
  - Import VMs using the provided OVA files.
    - Link: https://uofi.box.com/v/cs461-forensics-live-analysis
- Dead analysis
  - You may choose and install any tool you find necessary in your environment.
  - Or use the provided Ubuntu VM which has the recommended tools installed.
    - Link: https://uofi.box.com/v/cs461-forensics-vm
  - Download and decompress disk image files you will examine.
    - Link: https://uofi.box.com/v/cs461-forensics-disks

Calculating file checksums

[demo]

- Calculating SHA256 hash of a file:

```
$ shasum --algorithm 256 <filename>
```

- Verifying a file's checksum with a known value:

```
$ # checksum.txt contains following
$ # <sha256-hash>  <filename>
$ shasum --algorithm 256 --check checksum.txt
```


Searching disk images

[demo]

- Search by string using Unix utilities (e.g. `strings` and `grep`)

```
$ # find all strings that look like IPv4 addresses
$ strings -t d -n 7 victim.raw | egrep
"\b([0-9]{1,3}\.){3}[0-9]{1,3}\b” > ips.txt
```

  - Difficult to identify which files contain matching strings

Searching filesystems with TSK

[demo]

First, use `fdisk` to figure out partition offsets.

```
$ fdisk -l victim.raw
```

- Use `fls` to find inode of ‘etc/network/interfaces’ file on the filesystem.

  ```
  $ fls -o <sector-offset> -rp victim.raw |
  grep 'etc/network/interfaces$’
  ```

- Use `istat` to see file attributes (or `icat` to dump file content).

  ```
  $ istat -o <sector-offset> victim.raw <inode>
  ```


Creating a filesystem timeline

[demo]

1. Create a list of all files and their attributes in a format understood by `mactime`.
   ```
   $ fls -o <sector-offset> -r -m / victim.raw > bodyfile.txt
   ```
2. Create a timeline.
   ```
   $ mactime -b bodyfile.txt -d > timeline.csv
   ```
3. Mactime output meaning- https://wiki.sleuthkit.org/index.php?title=Mactime_output

Mounting disk images

[demo]

1. Create a directory where you will mount the image.
   ```
   $ mkdir ~/victim_fs
   ```
2. Use `fdisk` to figure out partition offsets.
   ```
   $ fdisk -l victim.raw
   ```
3. Mount the partition you want.
   ```
   $ sudo mount --options loop,ro,noexec,offset=<byte-offset>
   victim.raw ~/victim_fs
   ```
   - `loop` – set up a loop device (`/dev/loopX`) corresponding to the mounted partition
   - `ro` – mount the filesystem read-only
   - `noexec` – do not allow execution of any binary on the mounted filesystem
   - `offset` – set byte offset to point to the partition's starting point