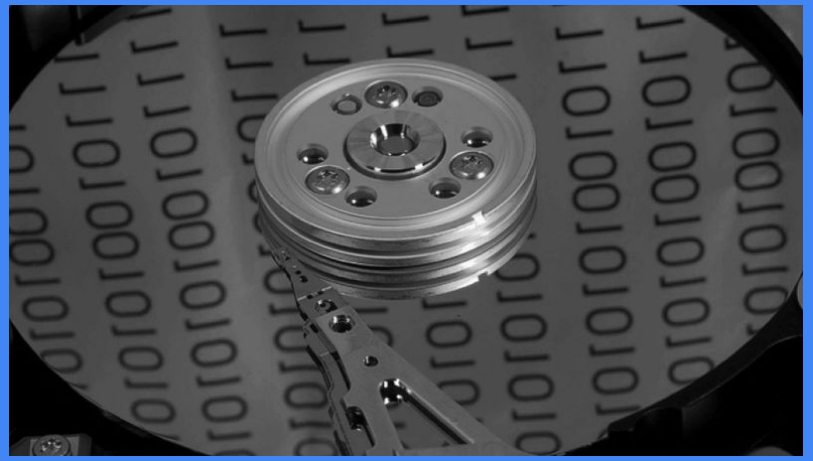
# Digital Forensics



Ravitha Rajalakshmi.N
Asst.Prof (Sr.Gr)
PSG college of Technology

# Digital Forensics

**Cybercrime** - any criminal offence that involves computer / network or ECD where the computer is used to either commit the crime or is the target of the crime.

**Digital Forensics** is the process of identifying, preserving, analyzing and presenting **digital evidence** in a format that is **legally acceptable**.

*Proactive & Reactive*

# Forensic Investigation Process

**IDENTIFICATION**

Identify the system to be investigated

**DATA ACQUISITION/ PRESERVATION**

Taking images of the drives/partition belonged to the identified system


**PRESENTATION**

Reporting of evidences found during analysis phase.

**FORENSIC ANALYSIS**

Analyze digital artifacts inside the data for evidences

**DATA RECOVERY**

Recover deleted data from the image files

# Identification


## Storage Devices


**Source:** https://www.corsair.com/tw/zh/blog/voyager-air-firmware-june2013

# Data Acquisition and Preservation

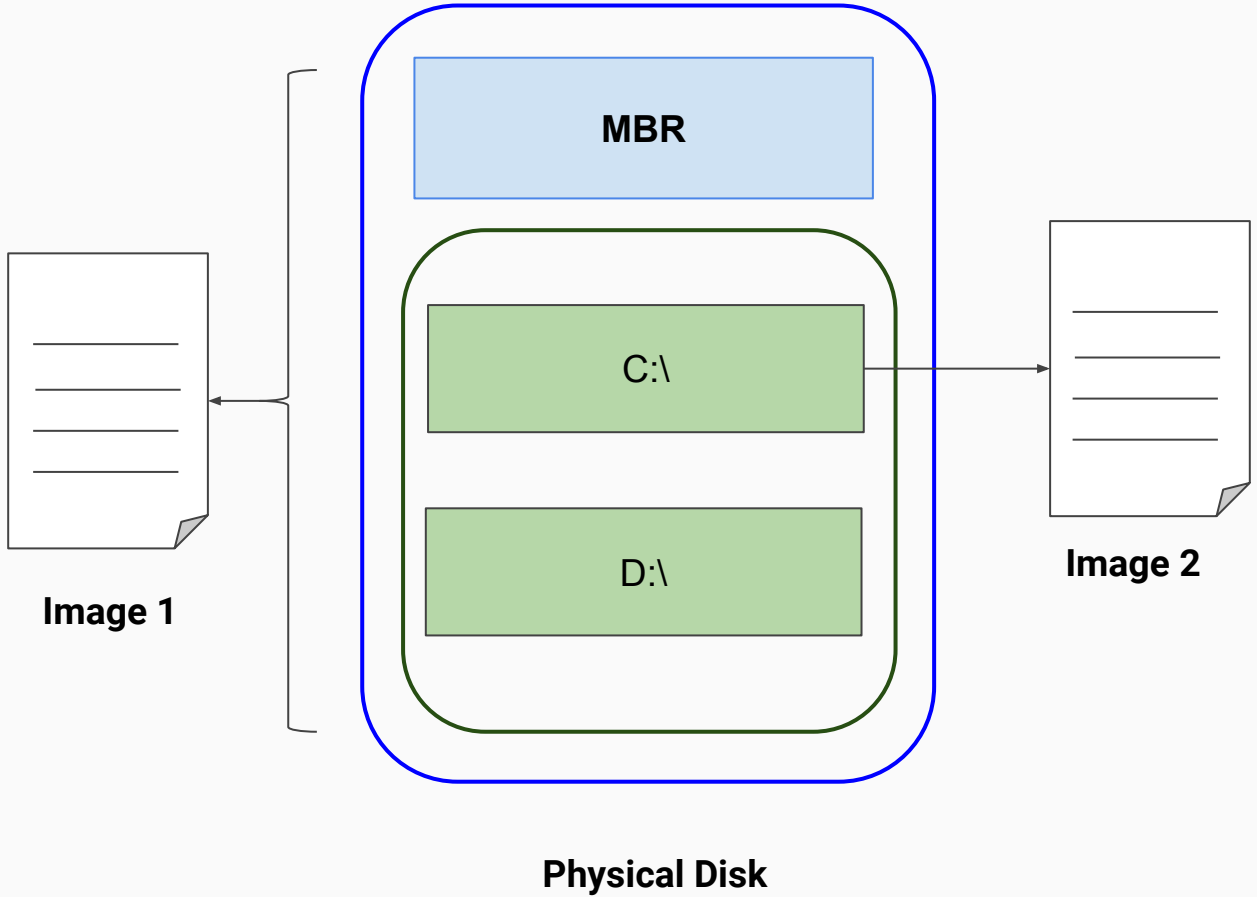
- **Holistic capture and preservation of the evidence is required**
- Transfer the contents to another empty media storage
- Maintain integrity of the copy (Transfer is complete) Copy is an exact replica of the original media
- **Chain of Custody** is maintained

***Chain of Custody*** *- log for the digital evidence*

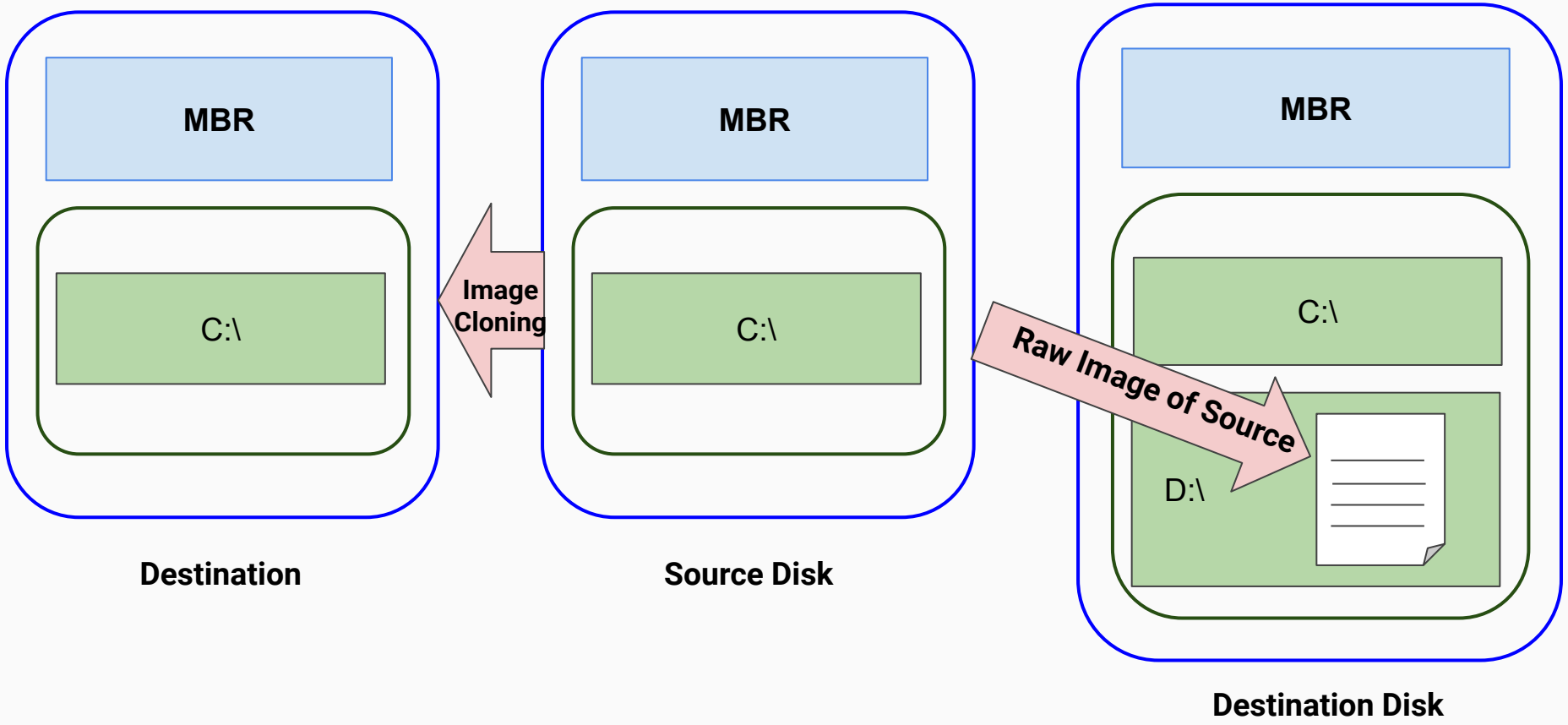
**Hardware Write Blockers**


**Source:** https://en.wikipedia.org/wiki/File:Portable_forensic_tableau.JPG

A **disk image** is a bit by bit copy of a full disk or a single partition from a disk. Because the contents of a disk are constantly changing on a running system, disk images are often created following an intrusion or incident to preserve the state of a disk at a particular point in time.



# Methods for creating Forensic Image

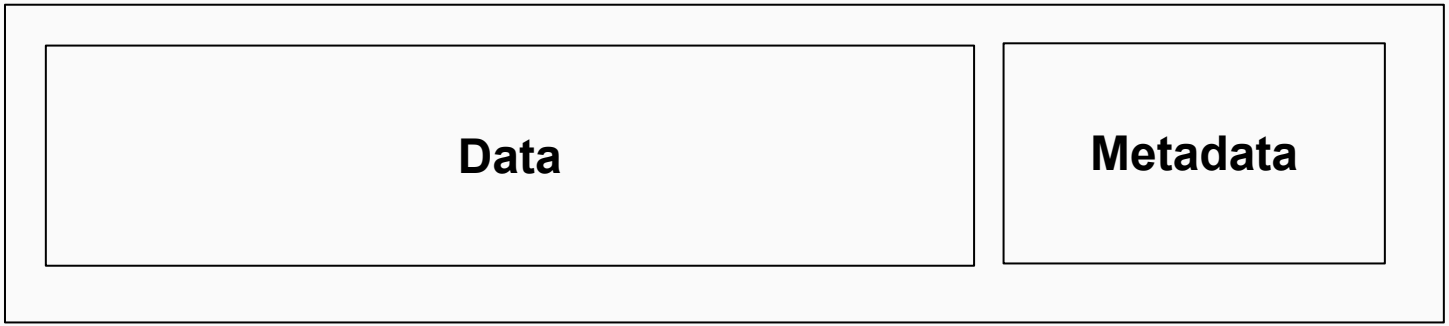


# Disk Image File Formats

- Raw Imaging Format
- AFF (Advanced Forensic Format)

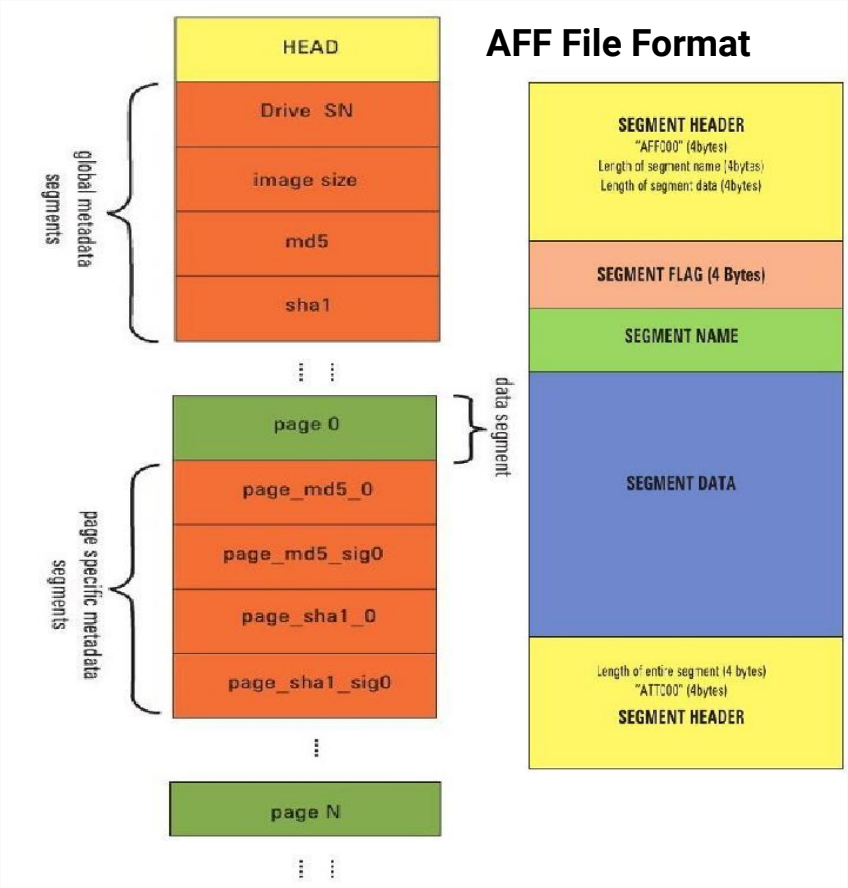
} Independent of any forensic package

- Encase Evidence File Images (Express witness format E01(EWF))

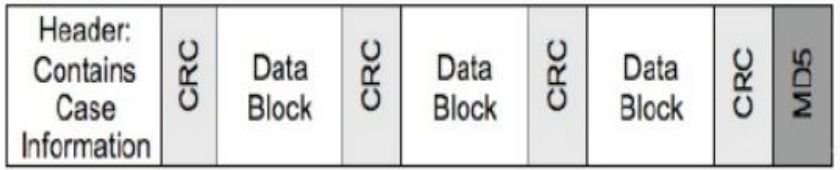



**Raw Image Format** (.raw , .dd , .img)

# Contd ..




## Encase File Format



# Tools for acquisition

- **dd** — Command line utility for acquiring digital images

  **dd if=<media/partition on a media> of=<image_file>**

  **Example:** dd if=/dev/sdc of=image.dd

- **dd_rescue** — Imaging potentially failing media

  **dd_rescue <input file> <out file>**

  **Example:** dd_rescue /dev/sdc image.ddrescue

- **dcfldd** — Provides additional features

  **dcfldd <options> if=<input media> of=<image_file>**
  **Example:**
  dcfldd if=/dev/hdc **hash=md5 hashwindow=10G md5log=md5.txt hashconv=after bs=512 conv=noerror,sync split=10G splitformat=aa** of=image.dd

# Forensic Analysis

- **Data Recovery**
  - Recover and analyze deleted files that have not been overwritten
  - Carving unallocated and slack space
- **String and Keyword Searching**
  - Identify text within a binary file
- **Volatile Evidence Analysis**
  - Provides details about the state of the system
  - Applications used by the suspect
  - Connections, processes and cache tables gathered from RAM
- **Timeline Analysis**
  - Creates timeline of events
- **System file analysis**
  - Any unauthorized changes that are made to system binaries

## Kali linux

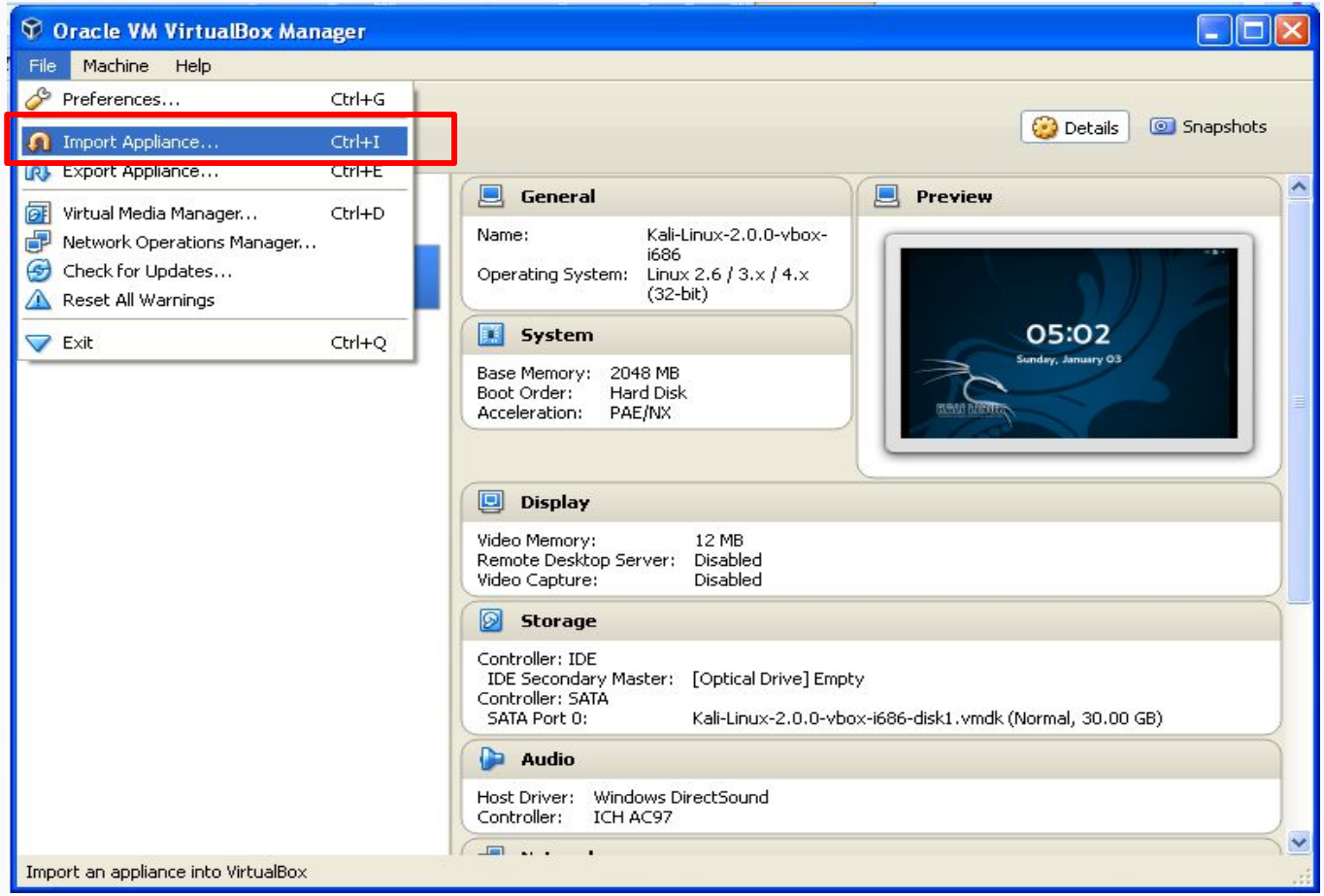
open source linux distribution specially tailored for digital forensics

- Binwalk
- Autopsy
- Bulk_extractor
- DFF
- md5deep
- Volatility
- Volafox
- chrrootkit

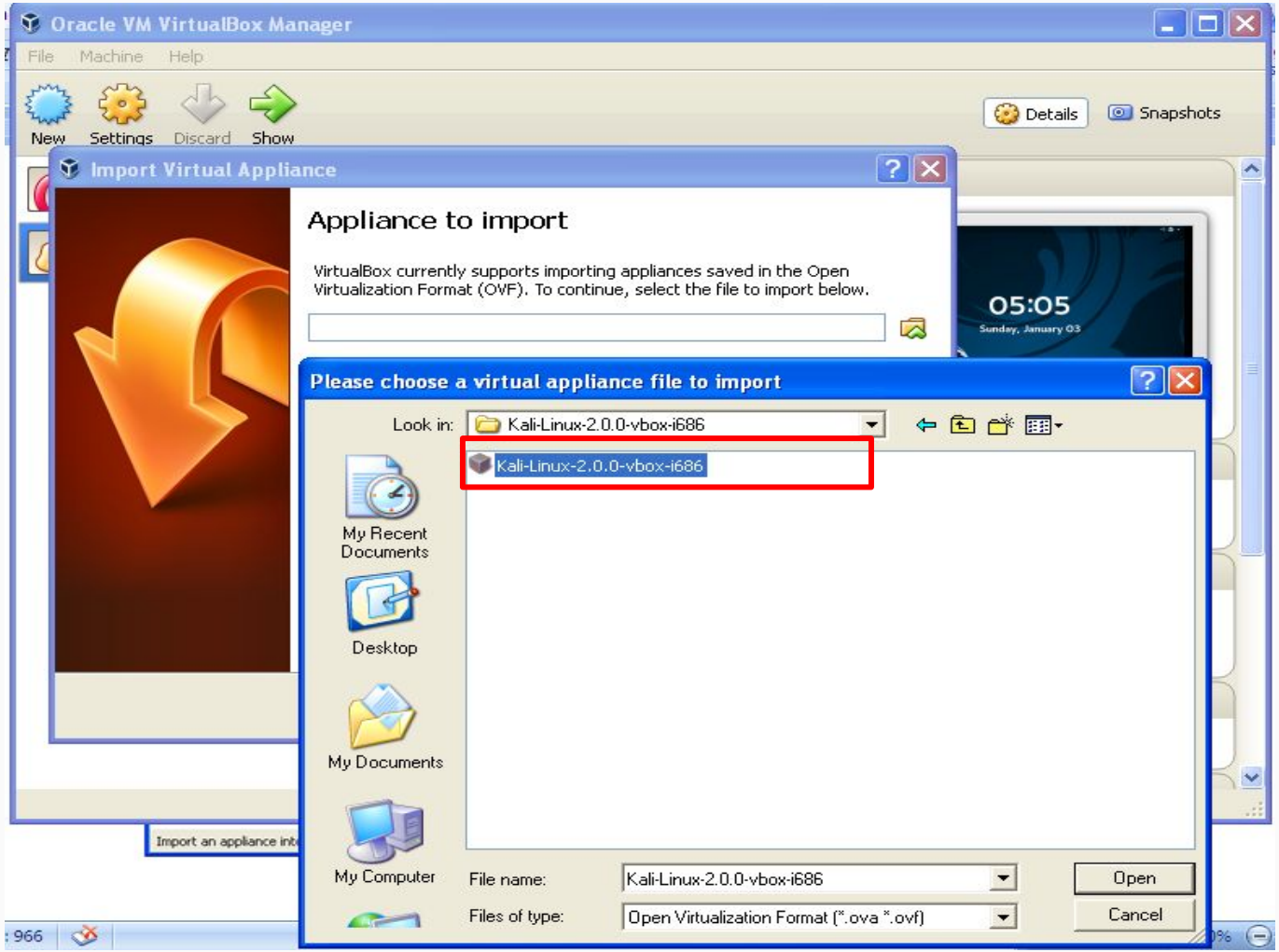
| Tools | Purpose |
|---|---|
| Autopsy | Examine file system in an non-intrusive manner |
| Binwalk | Search given binary image for embedded files and executable code |
| Bulk-Extractor | Extract features such as email-address, credit card numbers, URLs and other type of information from digital evidence files. |
| DFF | Digital Forensic Framework<br>Collect, Preserve and reveal digital evidences without compromising system and data |
| Guymager | Media acquisition |
| Foremost | Recover lost files based on headers, footers and internal data structures |
| md5deep | Verify the integrity of the file |
| Volatility | Extraction of digital artifacts from RAM (Random Access Memory) |
| chkrootkit | Check the system for common rootkits |

Kali linux distribution is booted as a virtual machine inside Virtual Box.

**Virtual Appliance - pre-configured operating system**



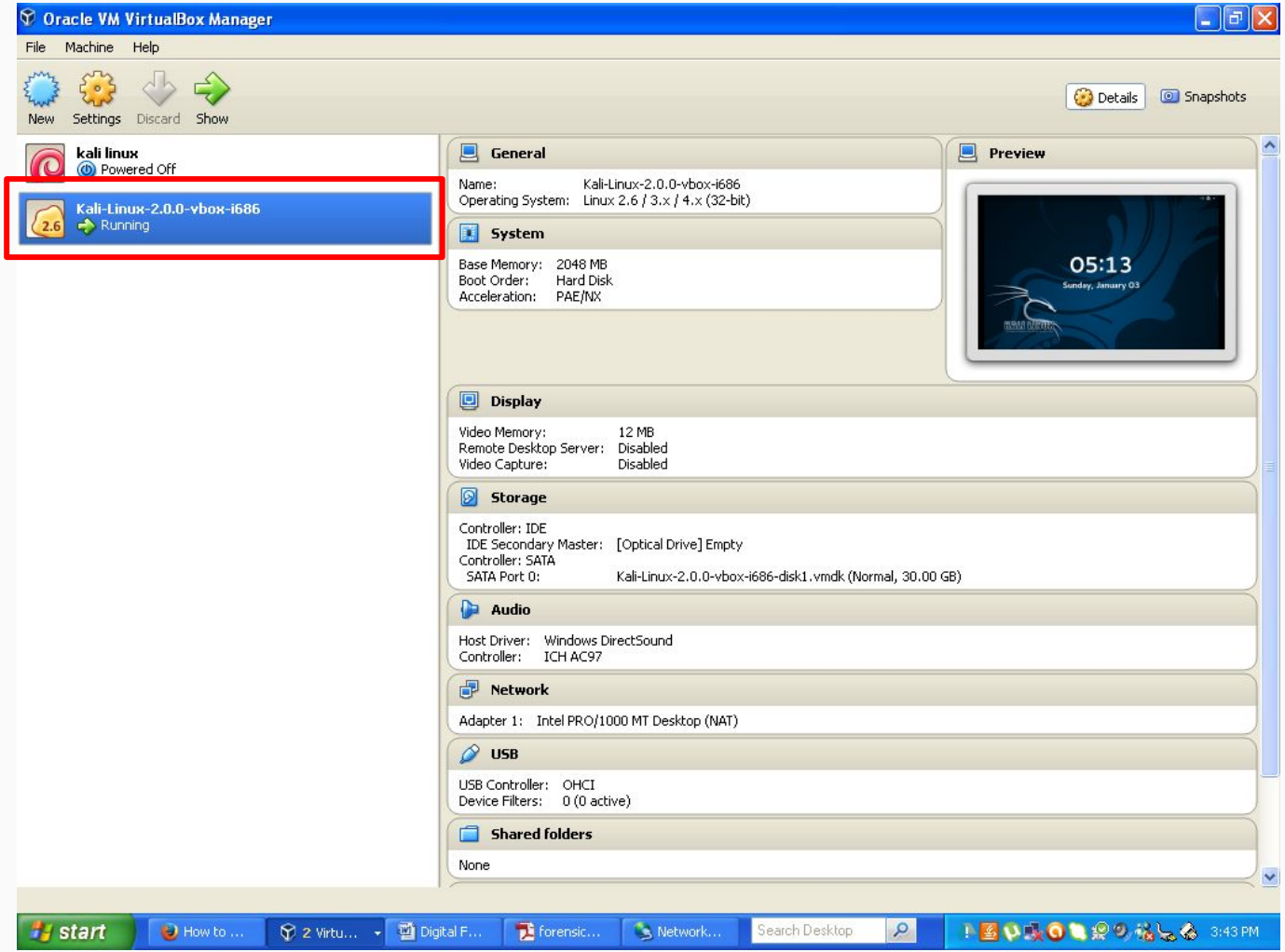
Choose the kali linux ova file located in the local directory and click **Import**

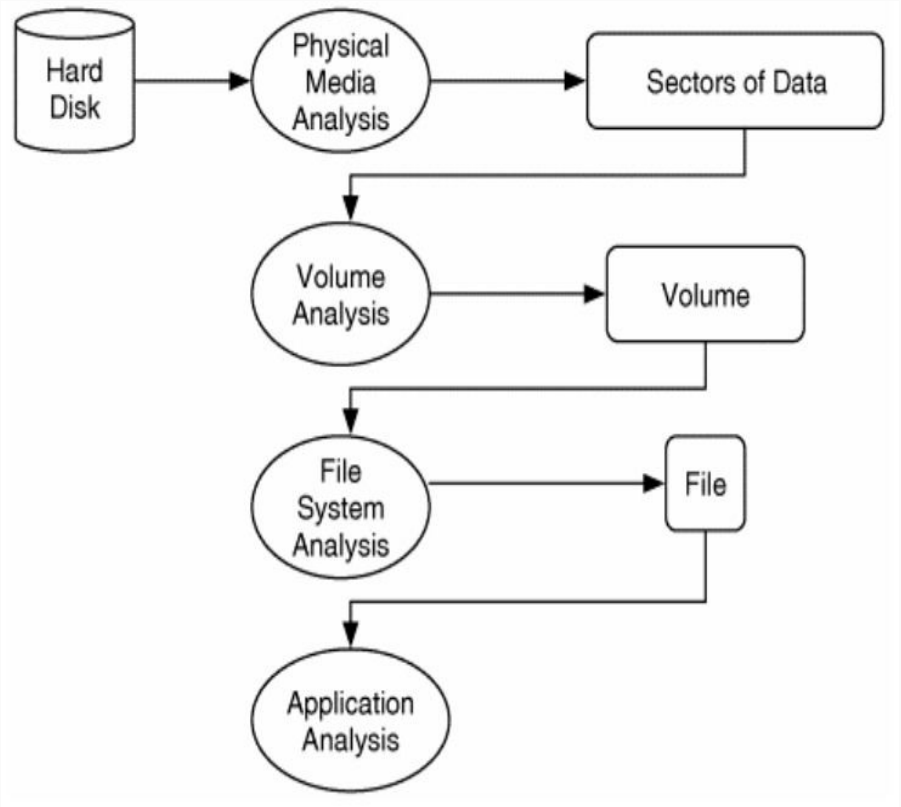
Start the virtual machine

Default username : **root**
Password : **toor**

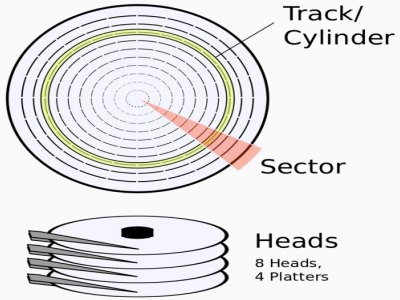
All the necessary Images are available in **/root/Documents/Images**



# Forensic Analysis- Autopsy Sleuth Kit

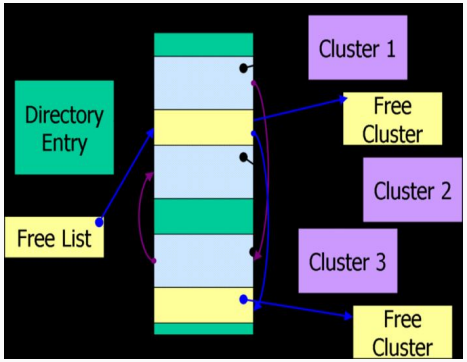


# Layers of Data Organization

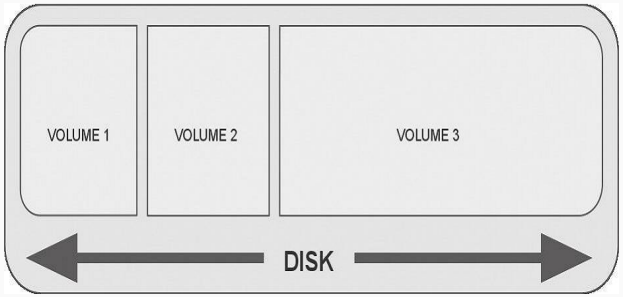



**Disk Layer**
Physical storage device
Analyzing information at this level is bit complex

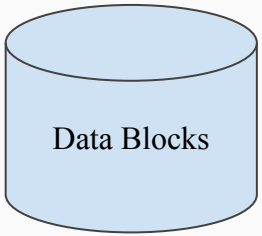
Track - concentric circle that stores information
Sector - section of track with specified size
Cylinder - column of track across platter
Sector - physical address / firmware




**File system Layer**
Metadata specific to file system
Describes the layout of files
Volume is organized effectively by file systems




**Partition Layer**
Disk is divided into partitions
Each partition/volume can use different file system




**Block Layer**
Actual Data resides in a block

# Sleuth kit tools

| Layer | Tools | Purpose |
|---|---|---|
| Image File | img_stat | Display information about a disk image |
| | img_cat | Dump the entire bitstream of disk image<br>(Removes wrapper if using EO1 , AFF ) |
| Volume System | mmls | Display partition layout of a volume system |
| | mmstat | Display information about volume system |
| | mmcat | Dump the entire bitstream of a partition |
| File System | fsstat | Display file system layouts<br>Layout , sizes , labels |
| File Name | fls | List allocated and unallocated file entries |
| | ffind | File entries of a given metadata |

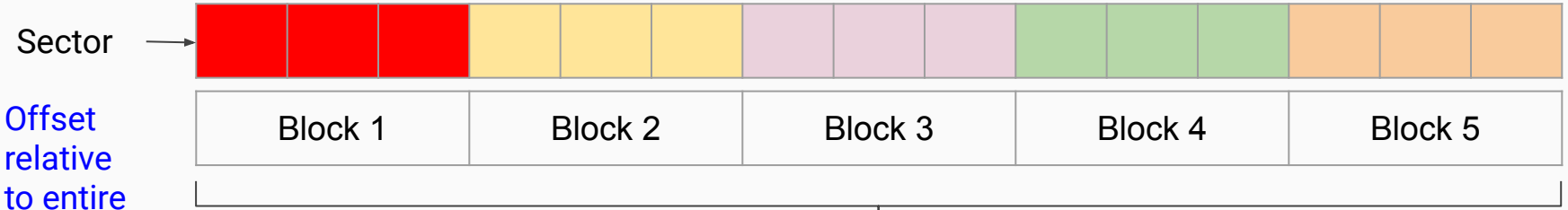
# Contd ..

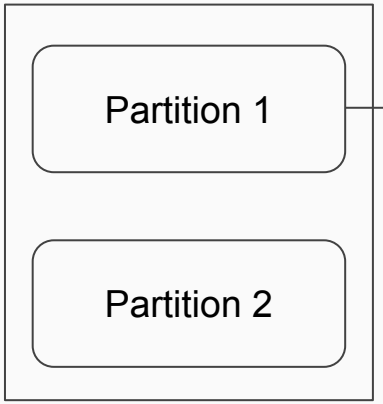
| Layer | Tools | Purpose |
|---|---|---|
| Metadata | ils | List metadata structure and their contents |
| | ifind | Find metadata structure referred by a file name entry |
| | istat | Display information about a specific metadata structure |
| | icat | Extract data units of a file specified by metadata structure |
| Data Unit | blkls | List details about data units (unallocated) |
| | blkstat | Display information about specific data unit |
| | blkcat | Extract contents of a data unit |
| | blkcalc | Calculate location of where data in unallocated space exist |

# Contd ..

| Layer | Tools | Purpose |
|---|---|---|
| Additional tools | tsk_loaddata | Extract metadata into a SQLite database |
| | tsk_recover | Extract allocated or unallocated files from a disk image |
| | mactime | Create timeline of activity |
| | sorter | Sorts files based on type |

# Organization of data




| Inode entry | File | Metadata/ block information | Allocation status |
|---|---|---|---|
| 1023 | /usr/login/x.c | 1-4 | Allocated |
| 34 | /root/bin | 6 | unallocated |

# Real - time Analysis

**Case study 1: Deleted File Identification and Recovery**

**To recover deleted file from raw image able2.dd**

1. Analyze the Image
2. Display the partition table
3. Identify the sector offset for second partition
4. Display the file system
5. List files
6. Identify delete files
7. Recover the image corresponding to inode entry 11108
8. Recover the deleted file corresponding to inode entry 2139
9. Identify the type of the deleted file

# Real - time Analysis

**Case study 2: Physical String Search and Allocation Status**

1. Identify the byte locations which contains string “Cybernetik”
2. Find the sector to which byte corresponds to
3. Identify its partition
4. Calculate byte offset within partition
5. Find the block number
6. Get the status of the block
7. Identify inode entry
8. Using the inode number recover the contents

# Real time analysis

**Case study 3: Speed up the String search by extracting the unallocated block**

1. Extract unallocated blocks of the partition
2. Search keyword inside the block
3. Identify the block in extracted unallocated blocks
4. Find the actual block offset in partition
5. Get the status of the block
6. Identify inode entry
7. Using the inode number recover the contents

# Task 1

Identify deleted files in dfr-01-fat.dd

# Challenge 1

**Unzip the folder to get the flag**
**flag.zip**

# Challenge 2

**Hint is in the figure hex_editor.png**

# Challenge 3

**What is hidden in the image?? flag.png**

# Challenge 4

**Find the flag in raw image animals.dd**

# Test Images

1. Computer Forensic Reference Datasets https://www.cfreds.nist.gov/
2. Digital Corpora https://digitalcorpora.org/corpora/disk-images
3. Digital Forensic Tool Testing Images http://dftt.sourceforge.net/
4. LinuxLeo https://www.linuxleo.com

**Competition links:**

1. https://picoctf.com/
2. https://dfrws.org/dfrws-forensic-challenge

# Tutorial

## dcfldd options

- split=<bytes> Specifies the size of each of the output image files.
- vf=<file> Specifies the image file that needs to be verified against the input.
- splitformat=<text> Specifies the format for multiple image files when using splitting.
- hash=<names> Specifies one or more (a comma-separated list) of hash algorithms, such as md5, sha1, etc.
- hashwindow=<bytes> Specifies the number of bytes of input media for calculation of hash.
- <hash algorithm>log=<file> Specifies output files for hash calculations of a particular hash algorithm.
- conv=<keywords> Specifies conversions of input media according to a comma-separated list of keywords, such as noerror (continue after read errors), ucase (change lower to upper case), etc.
- hashconv=before|after Specifies the hash calculation before or after the specified conversion options.
- bs=<block size> Specifies the input and output block size.