# MODULE 04 Data Acquisition and Duplication

Lara Alofi

2110886

Y

Computer Hacking Forensic Investigator (CHFI)

Lab Session Identifiers

1. <https://labclient.labondemand.com/LabClient/aaca03af-396b-4571-8d55-07504b73042a>

Username on EC-Council System

1. 2110886@uj.edu.sa

A screenshot of a computer

Description automatically generated

**Lab 01: Create a dd Image of a System Drive and compute MD5 hash for future validation**

In this lab, I learned how to create a forensic image of a system drive using the dd tool on a Windows machine. The task involved making a bit-by-bit copy of the primary disk and saving it as a .dd file. I also learned how to compute an MD5 hash of the image to ensure its integrity for future validation. The image was stored in a shared folder, simulating external storage, and will serve as evidence for investigation. The lab demonstrated essential steps in digital forensics for safely acquiring and verifying disk images.

**Lab 02: Convert Image File from E01 Format to dd Format**  
In this lab, I learned how to convert an E01 forensic image file to a dd format using the `xmount` tool on a Linux workstation. The E01 format is commonly used in Windows-based forensic investigations, but for Linux systems, it’s necessary to convert it to dd for analysis. I installed the `xmount` tool, copied the E01 file from a shared directory, and converted it to dd format by mounting it in the Documents folder. This process allows further forensic investigation on Linux workstations.

**Lab 03: Mount Images on a Linux Forensic Workstation**  
In this lab, I learned how to mount a forensic image file on a Linux forensic workstation. After converting the E01 image to a dd format in the previous lab, I mounted the dd image using the `mount` command to access its contents. I also used the `losetup` command to attach a Mac file system image to a loop device. This process allowed me to analyze the files within the mounted images, which is essential in forensic investigations for gathering evidence and insights from digital storage devices.

**Lab 04: Acquire RAM from Windows and Linux Workstations**  
In this lab, I learned how to acquire a RAM dump from both Windows and Linux systems. Using tools like Belkasoft RAM Capturer for Windows and fmem and LiME for Linux, I was able to capture volatile memory, which contains critical information that can be lost when the machine is powered off. Acquiring and analyzing RAM helps forensic investigators gain insights from live systems, such as running processes, open files, and network connections, which are vital during investigations.

**Lab 05: Create Customized Images from an Image Containing NTFS File System**  
In this lab, I learned how to use DiskExplorer for NTFS to investigate the NTFS file system and analyze forensic images. By importing a forensic image into the tool, I was able to navigate through specific sectors, view hex values, and copy the hex data for further analysis. This process is crucial for forensic investigators, as it allows them to extract and analyze hidden or encrypted data stored in an NTFS file system. Additionally, I practiced creating a customized image from a forensic image and understanding the structure of NTFS at a deeper level.

**Lab 06: View Contents of Forensic Image File**  
In this lab, I learned how to use AccessData FTK Imager to examine forensic images without altering the original evidence. The key steps involved installing and launching the tool, adding a forensic image (Windows\_Evidence\_001.dd), and exploring its contents via the Evidence Tree. I also viewed individual file properties and hex values, which provide insights into the raw data of files, even those that may have been deleted or overwritten. This lab highlighted the importance of reviewing forensic images for potential evidence and ensuring no modifications are made to the original files during the investigation process.

**Lab 07: Access a Disk Image Using PyTSK Tool**  
In this lab, I learned how to use the PyTSK tool to access and examine files and directories within a disk image, specifically using a Python-based script. The key steps involved installing PyTSK in an Ubuntu Forensics virtual machine, copying the forensic image file (Linux\_Evidence\_001.img) to the home directory, and writing a Python script to analyze the disk image. After running the script, I could view the root directories and associated files within the disk image. This lab demonstrated the effectiveness of PyTSK in quickly accessing forensic data in a disk image.