# Cyber Security Lab

## Cryptographic Hashing and Data Integrity

Create a Sample File: Open your Linux terminal and create a simple text file named `original_file. txt`.

```
echo "This is the original and authentic content."> original_file.txt
```

`Calculate SHA-256 Hash:` Use the `sha256sum` command to generate the file's hash.

```
sha256sum original_file.txt
```

Record the output hash (The "Original Hash"):

## Demonstrate the Avalanche Effect (Integrity Check)

`Modify the File:` Use a text editor like `nano` to make a tiny change to the file. For example, change one character from a lowercase 'c' to an uppercase 'C' in the sentence.

```
nano original_file.txt
```

# Change: "This is the original and authentic content."
# To: "This is the original and Authentic content."

Recalculate the Hash:
Re-run the `sha256sum` command on the modified file.

```
sha256sum original_file.txt
```

ReCalculate MD5 Hash (for comparison): Use the `md5sum` command.

```
md5sum original_file.txt
```

`Compare:` Compare the `Original Hash` from the new hash.

## Simulating a Downloaded File Integrity Check

**Simulate Download:** Pretend you downloaded a software installer (installer.exe) and the website provided a known-good SHA-256 hash.
Create the simulated file:

```
touch installer.exe
```

Simulate the website's published hash:

```
KNOWN_HASH="e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b85
5" # SHA-256 of an empty file
```

Calculate the hash of your downloaded file:

```
DOWNLOADED_HASH=$(sha256sum installer.exe | cut -d ' ' -f 1)
```

Perform the Integrity Check:

```
echo $DOWNLOADED_HASH
echo $KNOWN_HASH
```