

## Activity: Create hash values

### Scenario

In this scenario, we need to investigate whether two files are identical or different.

Here's how you'll do this task: **First**, you'll display the contents of two files and create hashes for each file. **Next**, you'll examine the hashes and compare them.

### Task 1. Generate hashes for files

In this task, you need to display the contents of each of these files. You'll then generate a hash value for each of these files and send the values to new files, which you'll use to examine the differences in these values later.

1. Use the **ls** command to list the contents of the directory.
2. Use the **cat** command to display the contents of the **file1.txt** file:
3. Use the **cat** command to display the contents of the **file2.txt** file:
4. Review the output of the two file contents:

```
analyst@745baa6ce501:~$ ls
file1.txt  file2.txt
analyst@745baa6ce501:~$ cat file1.txt
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
analyst@745baa6ce501:~$ cat file2.txt
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

Although the contents of both files appear identical when you use the cat command, you need to generate the hash for each file to determine if the files are actually different.

5. Use the **sha256sum** command to generate the hash of the file1.txt file:
6. Use the **sha256sum** command to generate the hash of the file2.txt file:
7. Review the generated hashes of the contents of the two files:

```
9sxa5Yq20Ranalyst@745baa6ce501:~$ sha256sum file1.txt
131f95c51cc819465fa1797f6ccacf9d494aaaff46fa3eac73ae63ffbfd8267  file1.txt
analyst@745baa6ce501:~$ sha256sum file2.txt
2558ba9a4cad1e69804ce03aa2a029526179a91a5e38cb723320e83af9ca017b  file2.txt
```

## **Task 2. Compare hashes**

1. Use the **sha256sum** command to generate the hash of the file1.txt file, and send the output to a new file called **file1hash**:
2. Use the **sha256sum** command to generate the hash of the file2.txt file, and send the output to a new file called **file2hash**:
3. Use the **cat** command to display the hash values in the file1hash and file2hash files
4. Inspect the output and note the difference in the hash values.
5. Use the **cmp** command to highlight the differences in the file1hash and file2hash files
6. Review the output, which reports the first difference between the two files:

```
analyst@745baa6ce501:~$ sha256sum file1.txt >> file1hash
analyst@745baa6ce501:~$ sha256sum file2.txt >> file2hash
analyst@745baa6ce501:~$ cat file1hash
131f95c51cc819465fa1797f6ccacf9d494aaaff46fa3eac73ae63ffbfd8267  file1.txt
analyst@745baa6ce501:~$ cat file2hash
2558ba9a4cad1e69804ce03aa2a029526179a91a5e38cb723320e83af9ca017b  file2.txt
analyst@745baa6ce501:~$ cmp file1hash file2hash
file1hash file2hash differ: char 1, line 1
```