**Activity: Create hash values**

**Overview:** As a security analyst, you’ll need to implement security controls to protect organizations against a range of threats. That’s where hashing comes in. Previously, you learned that a hash function is an algorithm that produces a code that can’t be decrypted. Hash functions are used to uniquely identify the contents of a file so that you can check whether it has been modified. This code provides a unique identifier known as a hash value or digest. In this lab activity, we’ll create hash values for two files and use Linux commands to manually examine the differences.

**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.

**Start your lab:** click on “start lab” to start the lab.

**Task 1. Generate hashes for files**

The lab starts in your home directory, /home/analyst, as the current working directory. This directory contains two files file1.txt and file2.txt, which contain different data.

1. Use the ls command to list the contents of the directory.

**$ ls**

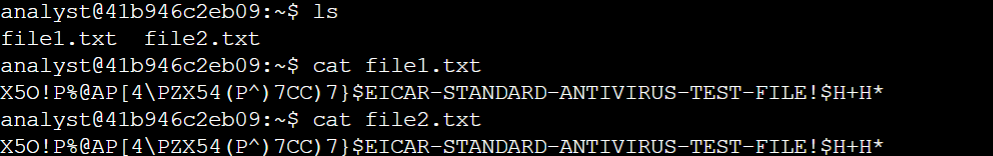
1. Use the cat command to display the contents of the file1.txt file:

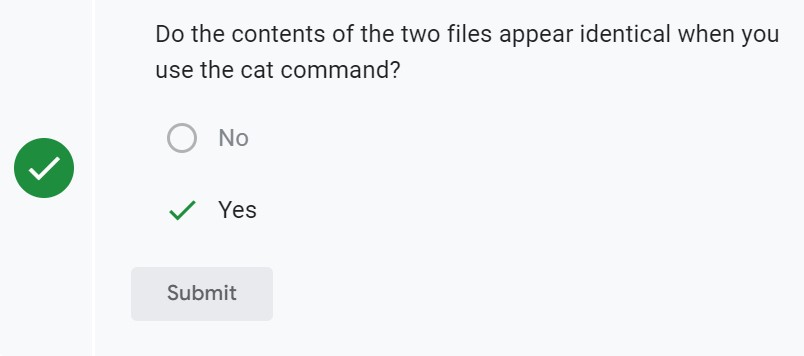
**$ cat file1.txt**

1. Use the cat command to display the contents of the file2.txt file:

**$ cat file2.txt**

1. Review the output of the two file contents:





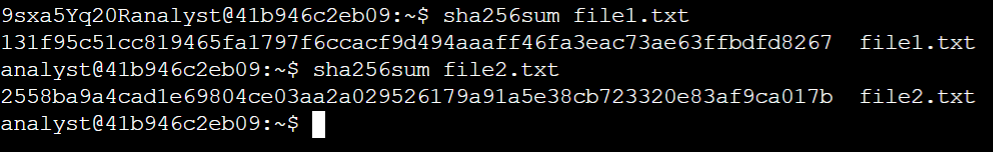
1. Use the sha256sum command to generate the hash of the file1.txt file:

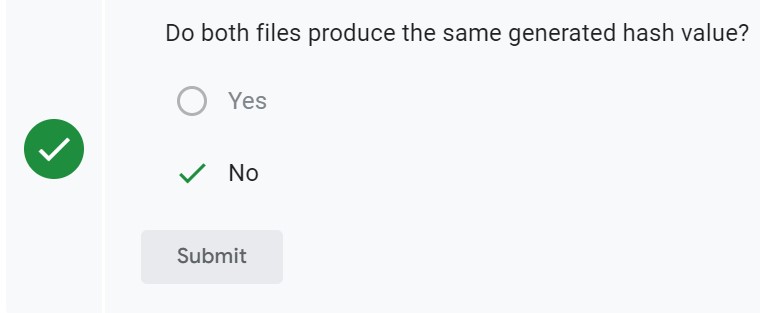
**$ sha256sum file1.txt**

1. Use the sha256sum command to generate the hash of the file2.txt file:

**$ sha256sum file2.txt**

1. Review the generated hashes of the contents of the two files:





**Task 2: compare the hash values**

In this task, you’ll write the hashes to two separate files and then compare them to find the difference.

1. Use the sha256sum command to generate the hash of the file1.txt file, and send the output to a new file called file1hash:

**$ sha256sum file1.txt >> file1hash**

1. Use the sha256sum command to generate the hash of the file2.txt file, and send the output to a new file called file2hash:

**$ sha256sum file2.txt >> file2hash**

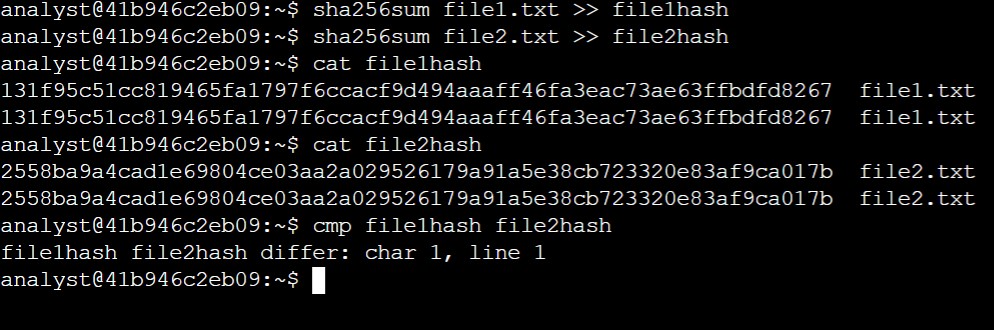
1. Use the cat command to display the hash values in the file1hash and file2hash files.

**$cat file1hash**

**$ cat file2hash**

1. Inspect the output and note the difference in the hash values.
2. Use the cmp command to highlight the differences in the file1hash and file2hash files:

**$ cmp file1hash file2hash**

1. Review the output, which reports the first difference between the two files:

Based on the hash values, is file1.txt different from file2.txt?

No

Yes

**Conclusion:**

You practiced how to

* compute hashes using sha256sum,
* display hashes using the cat command, and
* compare hashes using the cmp command.