(Approx. 450 words)

SHA256 Checksum for Windows 10

By Cal Esneault, Cajun Clickers Past President and leader of many Open-Source Workshops & SIGs

June 2021 issue, CCCC Computer News

www.clickers.org

office (at) clickers.org

A “checksum” is computed from the contents of a file using an algorithm that only considers the actual bytes that are in a file (works for any file type - .exe, .jpg, .docx, etc.). The result of the operation is a checksum, also known as a “hash” code. SHA256 is an industry standard encryption method generating a 256-bit hash, no matter the size of the target file. The hash code is given as 64 hexadecimal characters (each character represents 4-bits, thus 4 x 64 = 256).

No two files will have the same SHA256 hash, even if only one byte is changed in the source file. By comparing the hash of the original file to a subsequent hash of a copy of that file, you can verify that no changes have been made and no potentially nefarious code has been inserted. This is one way you can “sign” a document for authenticity or receive certified application code from a known source.

As a test, I created a short text file using Windows 10 Notepad (see below)



Next, I opened the command prompt by typing “cmd” in the search box next to the program “Start” button.



To get the hash, I used the certutil utility (see below, with response).

C:\Users\Windows 10\Documents>certUtil –

hashfile test1.txt SHA256

SHA256 hash of test1.txt:

287c24eaa9f4ff5fe71e53a72ddbd7e2ba0bf5ad21

b63d2f6215778b7a04f2e4

As a test, I changed the name of the test file and ran the hash again. It gave the same 64-character result.

I ran another test where I changed the capital “S” in the test file to lower case and rename d the file to “test1-bad.text”. The hash of this file was:

C:\Users\Windows 10\Documents>certUtil –

hashfile test1-bad.txt SHA256

SHA256 hash of test1-bad.txt:

947226813b54354fabb021ff8a026b2391365622e

0e78239b2950c07331e31f8

As you can see, any slight change in a file will dramatically change the hash result. Thus, if you have the resultant hash of a file you create, you can guarantee that no changes have been made. This is good for legal issues, and is the underlying approach used for “digitally signed” documents. used the SHA264 since it is still considered secure. MD5 and SHA1 algorithms are now considered vulnerable by some experts.

Other options, SHA384 and SHA512 are also available, but they have more cryptographic bits and are thus slightly more complex. There are many third-party programs to perform hashing, but I chose to demonstrate the default option that is always available in Windows 10. Hashing is also available in other operating systems. I usually encounter it when verifying the authenticity of downloads of Linux distributions