Exercise - 4

Executable Hash: Hashing is concept that we hear from time to time in the field of cybersecurity. The long strings of random numbers and alphabets, which are generated and used in various methods. Executable hash is a unique identifier for an executable file. It can be created by applying a hash function to the file. The hash function is a one-way function, meaning that it is very difficult/impossible to reverse the process and obtain the original file from the hash.

They are used for a different purposes, like:

- Verifying the integrity of executable files.
- Detecting malware.
- Signing executable files.

Executable hashes are a very important tool for ensuring the security and integrity.

Background/History:

The idea of hashing appears to have been originated by H. P. Luhn, who wrote an internal IBM memorandum in January 1953 that suggested the use of chaining; in fact, his suggestion was one of the rst applications of linked linear lists.

Konheim has a recent book, "**Hashing in Computer Science**" which focuses on the mathematics and history of non-cryptographic hash functions but considers post 1973 cryptographic hash functions only. Gustavus Simmons worked on authentication codes at Sandia Labs for many years, I looked up for some history in his book but there isn't anything pre 1970 for cryptographic checksums. CRC is a linear conceptually date back to error correcting codes, which are not strong cryptographically. It was proposed by Peterson in 1961.

They became part of computer science for long time. They are initially used in hash tables to locate data from the records quickly. After that the cryptographic hash functions are developed. They have properties making them suitable for security of the applications. Increase in the need of cybersecurity, hashing has become quickly an essential for ensuring data integrity and security.

Creation:

In Windows:

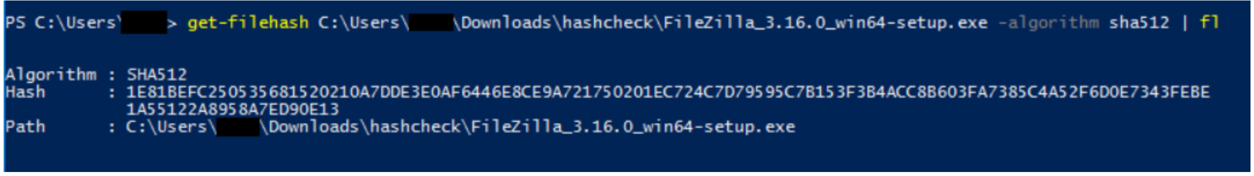
Steps for getting the hash value:

- Open command line/ PowerShell.
- cd to /path/to/your/file/or/directory
- Use the below command to get the hash value of the file

  ```
  Get-Filehash -path file/path/filename.extenision -algorithm Nameofalgorithmtouse
  ```

  or

  ```
  Get-Filehash -path file/path/filename.extenision -algorithm Nameofalgorithmtouse | fl
  ```
- The hash value will be displayed on the shell.
- Sample screenshot:



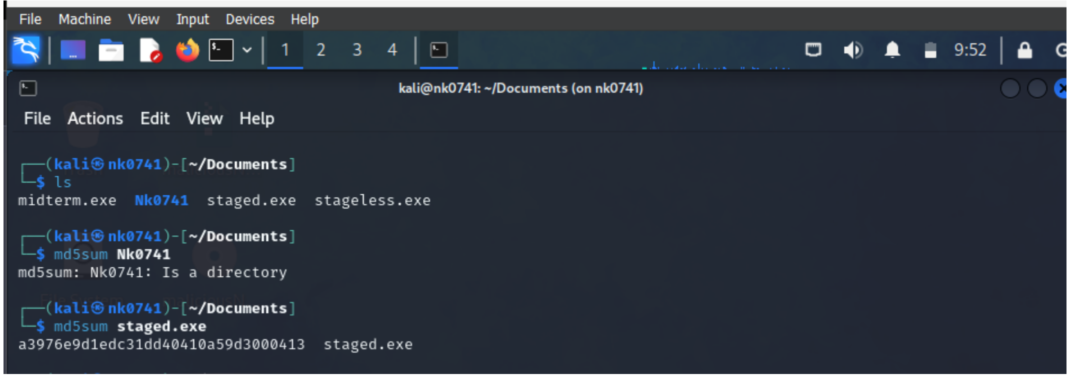
```
PS C:\Users\[redacted]> get-filehash C:\Users\[redacted]\Downloads\hashcheck\FileZilla_3.16.0_win64-setup.exe -algorithm sha512 | fl

Algorithm : SHA512
Hash      : 1E81BEFC2505356815 20210A7DDE3E0AF6446E8CE9A721750201EC724C7D79595C7B153F3B4ACC8B603FA7385C4A52F6D0E7343FEBE
            1A55122A8958A7ED90E13
Path      : C:\Users\[redacted]\Downloads\hashcheck\FileZilla_3.16.0_win64-setup.exe
```

- Here we're checking the hash value of filezilla.exe file.

In Linux:

Steps for getting the hash value:

- Open Terminal.
- cd to /path/to/your/file/or/directory.
- To get MD5 hash value:
  - `Md5sum filename`
- To get Sha hash value:
  - `Sha256sum filename`
- Sample Screenshot of running md5sum command on a staged.exe file.



File Machine View Input Devices Help
1 2 3 4
9:52
kali@nk0741: ~/Documents (on nk0741)
File Actions Edit View Help

```
┌──(kali㉿nk0741)-[~/Documents]
└─$ ls
midterm.exe  Nk0741  staged.exe  stageless.exe

┌──(kali㉿nk0741)-[~/Documents]
└─$ md5sum Nk0741
md5sum: Nk0741: Is a directory

┌──(kali㉿nk0741)-[~/Documents]
└─$ md5sum staged.exe
a3976e9d1edc31dd40410a59d3000413  staged.exe
```

Execution:

Hash values are not executable. They play a crucial role in the process of execution, especially where security and integrity are important. Before an .exe or any other file accessed or used or run, its hash value is checked to ensure that it has not been tampered with or altered. This process is known as hash verification. Thus making sure we have the original file.

Applications:

Examples of how executable hashes are used:

- When we download a software package from the internet, the website may provide you with the executable hash of the package. You can then use this hash to verify the integrity of the package before we install it.
- Security software products will monitor your system for changes to executable files. If the hash of an executable file changes, the security software will alert you, as this may be a sign that the file has been infected with malware.
- Software developers use executable hashes to sign their software packages. This allows users to verify that the software they are installing is authentic and has not been tampered with.
- Security
- Malware detection
- Network administration

Pros and Cons:

Pros:

- Easy to implement
- Lightweight technique
- Fast and efficient
- Provides security
- Can be used on anything/ any type of data.

Cons:

- Bypassed by sophisticated attackers.
- Using older hashing techniques make them vulnerable
- If both file and hash are modified, user cannot know that file is tampered.
- Difficult for regular users to verifying the hashes.
- Take more time as the file size increases.
- Some older hashing techniques produce same hash value for different inputs.

Reference:

- https://stackoverflow.com/questions/1424248/hash-of-an-exe-file
- https://community.spiceworks.com/how_to/127204-how-to-find-the-sha-hash-of-a-given-file
- https://security.stackexchange.com/questions/218591/is-it-possible-to-create-an-executable-file-by-hashing-a-different-file
- Sentinelone. (2023). What is hashing. https://www.sentinelone.com/cybersecurity-101/hashing/
- Crashing dialy. (2008). Hashing the executables – a look at hash and type. https://crashingdaily.wordpress.com/2008/04/21/hashing-the-executables-a-look-at-hash-and-type/
- https://spectrum.ieee.org/hans-peter-luhn-and-the-birth-of-the-hashing-algorithm
- https://spectrum.ieee.org/new-king-of-security-algorithms-crowned
- https://crypto.stackexchange.com/questions/56404/what-was-the-first-hash-and-what-problem-was-it-supposed-to-solve#:~:text=Knuth%2C%20TAOCP%2C%20Vol.,applications%20of%20linked%20linear%20lists.%22