NCCIC

National Cybersecurity and
Communications Integration Center

File Hashing

What is Hashing?

These days, pretty much everything has some kind of unique identifier; for example, stock items have SKUs, cars have VINs, employees have IDs, US citizens have SSNs, etc. One could say that **hashing is an attempt to "label" some chunk of data with a fixed-length, unique identifier.** A message digest, or hash, is a signature that identifies some amount of data, usually a file or message. Cryptographic hashing algorithms are one-directional mathematical formulae designed to generate a unique value for every possible input—in this case, the data. Common algorithms include MD5, SHA-1, SHA-256, and SHA-512, although there exist a number of more esoteric or use-case-specific methods.

Obviously, one cannot represent an infinite number of inputs with a finite-sized signature; when two inputs result in the same output, this is called a hash collision. Algorithms that produce a shorter hash are more likely to cause collisions. Both the MD5 and SHA-1 hashing algorithms have methods of producing collisions that threat actors can use to hide malicious content.

Why Should I Care About Hashing?

Hashing has a large number of applications in cybersecurity, centering around the ability to verify that data is legitimate. With a secure hashing algorithm, one could verify that a downloaded file matched what was expected. In fact, many projects provide hashes alongside their downloads. This provides some peace of mind that a file is authentic and has not been altered in transmission.

In addition, some online services provide the ability to look up whether a file is known to be malicious based on file hash. Rather than uploading the entire file for the service to analyze, one can just provide the hash, since it uniquely identifies the data in the file.

Hashes, Vendors, and You

Often, vendors will provide hashes with downloads of firmware or other critical files. This can provide an extra level of security/safety when downloading files and help to prevent watering hole attacks. If your vendor does not provide hashes, it may be possible to contact them and ask before deploying a new package. Additional safety measures are never a bad thing, especially when it comes to downloading files, even from trusted sources.

What and Why VirusTotal?

VirusTotal (VT) and other similar services allow one to submit files and URLs for inspection for malicious content and look up URLs and hashes to see existing reports from past submissions. This can be useful when attempting to determine if a resource is trustworthy, providing a way for someone to get an initial assessment of a file or website to make a more informed decision about whether or not it should be used. However, with great power comes great responsibility and two major caveats:

Caveat #1: If a file is not shown in VT as being malicious, this does not mean that it is safe. Antivirus software does not detect some malware, so a lack of detection does not mean that it is benign!

Caveat #2: While uploading files to VT is helpful to the security community as a whole, one should never upload files with sensitive content. Users with a paid subscription can download files, so any file uploaded to VT should be considered publicly available!

These caveats are not meant to dissuade use of online services like VT; instead, they are meant as a caution to think before uploading anything to an online service. Uploading any information to a third-party site should be considered carefully.

Hashing Options for Windows

CertUtil is a standalone command-line program that is shipped with Windows 7 and newer that can, as one of its functions, hash files. It supports the MD5, SHA-1, SHA-256, and SHA-512 algorithms. It can be run as follows:

```
CertUtil -hashfile <path> <algorithm>
```

Get-FileHash is a PowerShell cmdlet available in PS 4.0 (Windows 8.1) and newer that hashes files. It supports the MD5, SHA-1, SHA-256, and SHA-512 algorithms. It can be run as follows:

```
Get-FileHash -algorithm <algorithm> <path>
```

The Microsoft **File Checksum Integrity Verifier (FCIV)** (https://support.microsoft.com/en-us/help/841290; Microsoft KB841290) is a standalone command-line utility to hash files and verify hashes. Microsoft no longer supports it, but it works on modern Windows through Windows 10. It supports the MD5 and SHA-1 algorithms. It can be run as follows:

```
fciv.exe -both <path>
```

SysInternals **Sigcheck** (https://docs.microsoft.com/en-us/sysinternals/downloads/sigcheck) is a command-line tool that can calculate file hashes and other information for some file types. For example, it can display signing information for Windows executables. It supports the MD5, SHA-1, and SHA-256 algorithms. It can be run as follows:

```
sigcheck.exe -h <path> [<path> ...]
```

HashMyFiles (http://www.nirsoft.net/utils/hash_my_files.html) is a standalone GUI tool from NirSoft that calculates hashes over a set of files. It supports the CRC32, MD5, SHA-1, SHA-256, SHA-384, and SHA-512 algorithms, and also supports quickly pivoting to viewing reports about file hashes in VT. This tool runs under Windows 2000 and newer.

HashCheck (https://github.com/gurnec/HashCheck) is an open-source hash-calculation tool that integrates tightly into the Windows shell. It adds a "Checksums" tab to the Explorer "Properties" dialog that shows hashes for files relevant to the selected item. It supports the CRC32, MD5, SHA-1, SHA-256, SHA-512, SHA3-256, and SHA3-512 algorithms, and also has the ability to generate signature files.

About NCCIC

The National Cybersecurity and Communications Integration Center (NCCIC) is a 24x7 cyber situational awareness, incident response, and management center that is a national nexus of cyber and communications integration for the Federal Government, intelligence community, and law enforcement.

http://www.dhs.gov/national-cybersecurity-communications-integration-center