

How to verify hash values of SEH download files

Background

The integrity of the file in question can be checked using the hash values provided. The exact match of hash values ensures that the integrity and security of a file has not been manipulated by third parties and that it is an exact copy of the original file.

There are therefore two main reasons why the hash value of a downloaded file does not match the original sum:

- The file has been modified by another user in an unauthorized manner.
- The file was not downloaded properly and some of its elements may have been corrupted.

How to check a hash value

To check the hash value of the file in question, a check number of the downloaded file must be created and compared with the hash value provided by SEH. This can be done with an external tool or directly with the board tools of the respective operating system.

The hash value of the file in question can be calculated under **macOS/Linux** using the *shasum* tools and under **Microsoft Windows** using the *certutil* or *Get-FileHash* tools. The use of the tools is illustrated below using the software *d-sys-uds-20.1.32.bin* for utnserver Pro as an example.

SAH-256 hash value specified by SEH for the file *d-sys-uds-20.1.32.bin*:

3dcbd6cbbc7cc414889049223fe943101da76829497d6fbb7fb453c69e59b7b8

Windows command prompt

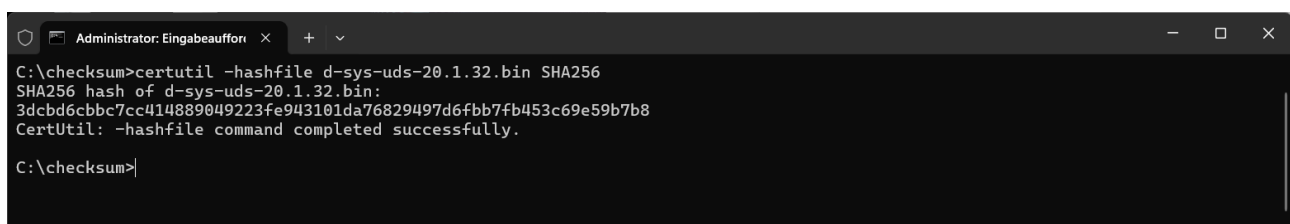
The general function call in the Windows command prompt is:

`CertUtil -hashfile [FILENAME] SHA256`

Example:

Calculating the hash value of the file "*d-sys-uds-20.1.32.bin*":

`C:> certutil -hashfile d-sys-uds-20.1.32.bin SHA256`



```
Administrator: Eingabeauffor... x + v
C:\checksum>certutil -hashfile d-sys-uds-20.1.32.bin SHA256
SHA256 hash of d-sys-uds-20.1.32.bin:
3dcbd6cbbc7cc414889049223fe943101da76829497d6fbb7fb453c69e59b7b8
CertUtil: -hashfile command completed successfully.

C:\checksum>
```

Microsoft Windows PowerShell

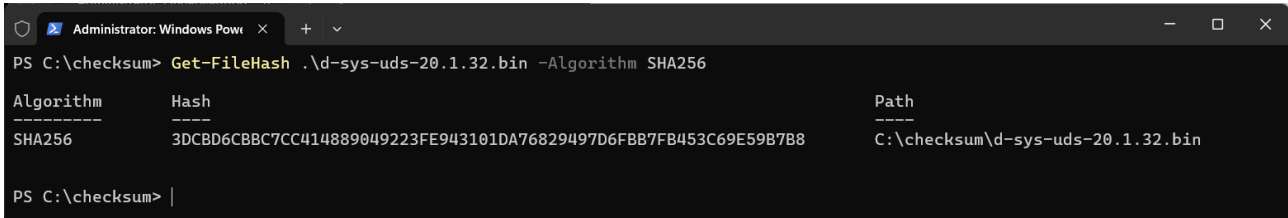
The general function call in Microsoft Windows PowerShell is:

`Get-FileHash [FILENAME] -Algorithm SHA256`

Example:

Calculating the hash value of the file "*d-sys-uds-20.1.32.bin*":

`C:> Get-FileHash .\d-sys-uds-20.1.32.bin -Algorithm SHA256`



```
Administrator: Windows Pow...
PS C:\checksum> Get-FileHash .\d-sys-uds-20.1.32.bin -Algorithm SHA256

Algorithm      Hash                                          Path
-----
SHA256         3DCBD6CBB7CC414889049223FE943101DA76829497D6FBB7FB453C69E59B7B8  C:\checksum\d-sys-uds-20.1.32.bin

PS C:\checksum> |
```

macOS/Linux

The general function call in the terminal is:

`shasum -a 256 [FILENAME]`

Example:

Calculating the hash value of the file "*d-sys-uds-20.1.32.bin*":

`→ shasum -a 256 d-sys-uds-20.1.32.bin`



```
Terminal --bash-- 137x8
|→ shasum -a 256 d-sys-uds-20.1.32.bin
3dcbd6cbbc7cc414889049223fe943101da76829497d6fbb7fb453c69e59b7b8 d-sys-uds-20.1.32.bin
[16:56] [~/checksum]
|→
```

With the `-c` option, the hash value can also be calculated in one step, compared with the hash value provided by SEH and then the result is displayed:

`→ echo '3dcbd6cbbc7cc414889049223fe943101da76829497d6fbb7fb453c69e59b7b8 *d-sys-uds-20.1.32.bin' | shasum -c`



```
Terminal --bash-- 137x8
|→ echo '3dcbd6cbbc7cc414889049223fe943101da76829497d6fbb7fb453c69e59b7b8 *d-sys-uds-20.1.32.bin' | shasum -c
d-sys-uds-20.1.32.bin: OK
[17:22] [~/checksum]
|→
```