

# **Verificación de la integridad** **de archivos mediante** **funciones hash**

**Índice:**

<b>Práctica en Windows .....</b>	<b>3</b>
Uso de CertUtil .....	3
QuickHash GUI .....	4
¿Te parecieron fáciles de usar? .....	5
¿Cuál es la ventaja de tener una interfaz gráfica? .....	5
<b>Práctica en Linux .....</b>	<b>5</b>
<b>¿Cómo podrías utilizar estos comandos para verificar la integridad de un archivo descargado de internet? .....</b>	<b>6</b>
<b>¿Por qué los algoritmos MD5 y SHA-1 ya no son recomendados para aplicaciones críticas? ...</b>	<b>6</b>
<b>Indica en qué situaciones podría ser aceptable utilizar MD5 en lugar de algoritmos más seguros como SHA-256 o SHA-512. ....</b>	<b>6</b>

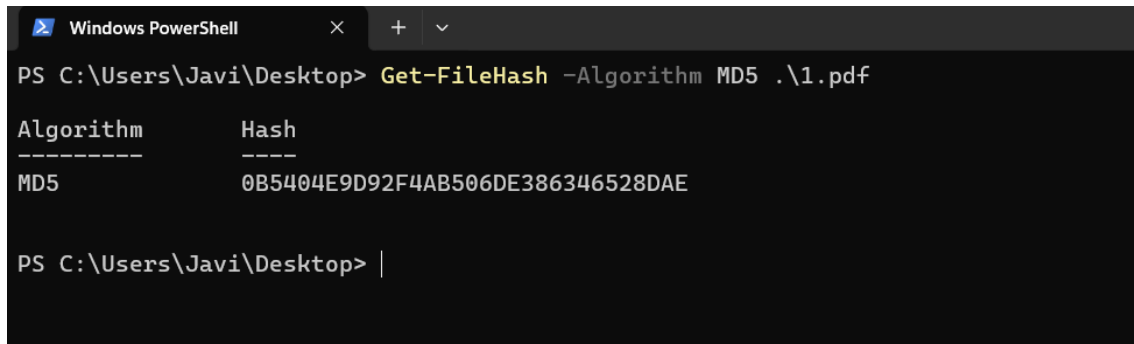
## Práctica en Windows

### Uso de CertUtil

En PowerShell, vamos a usar el comando “Get-FileHash -Algorithm MD5 y el nombre del archivo”

El primero hash que quiero sacar es con el algoritmo MD5

Resultado: 0B5404E9D92F4AB506DE386346528DAE



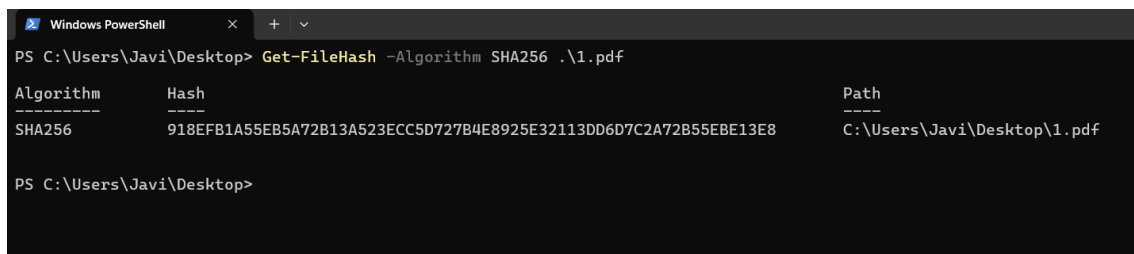
```
Windows PowerShell
PS C:\Users\Javi\Desktop> Get-FileHash -Algorithm MD5 .\1.pdf

Algorithm      Hash
-----
MD5            0B5404E9D92F4AB506DE386346528DAE

PS C:\Users\Javi\Desktop> |
```

Para SHA256 usaremos el comando “Get-FileHash -Algorithm SHA256 y el nombre del archivo”

Resultado: 918EFB1A55EB5A72B13A523ECC5D727B4E8925E32113DD6D7C2A72B55EBE13E8



```
Windows PowerShell
PS C:\Users\Javi\Desktop> Get-FileHash -Algorithm SHA256 .\1.pdf

Algorithm      Hash                                          Path
-----
SHA256         918EFB1A55EB5A72B13A523ECC5D727B4E8925E32113DD6D7C2A72B55EBE13E8  C:\Users\Javi\Desktop\1.pdf

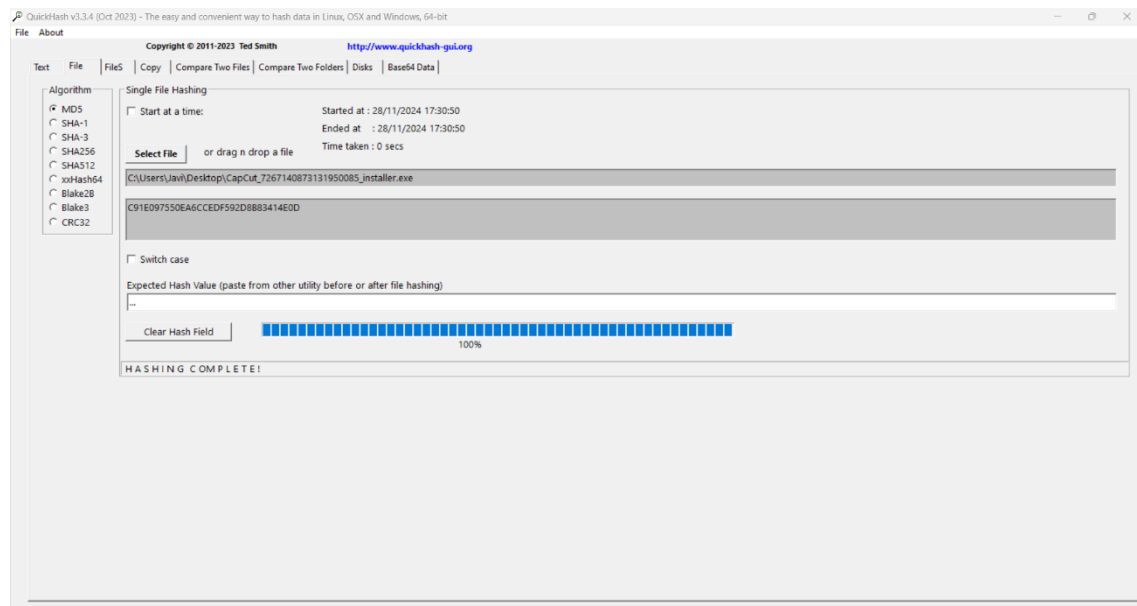
PS C:\Users\Javi\Desktop>
```

## QuickHash GUI

Esta aplicación es mas sencilla de utilizar ya que no hay que poner ningún comando, simplemente seleccionar el algoritmo y el archivo del cual queremos sacar el hash

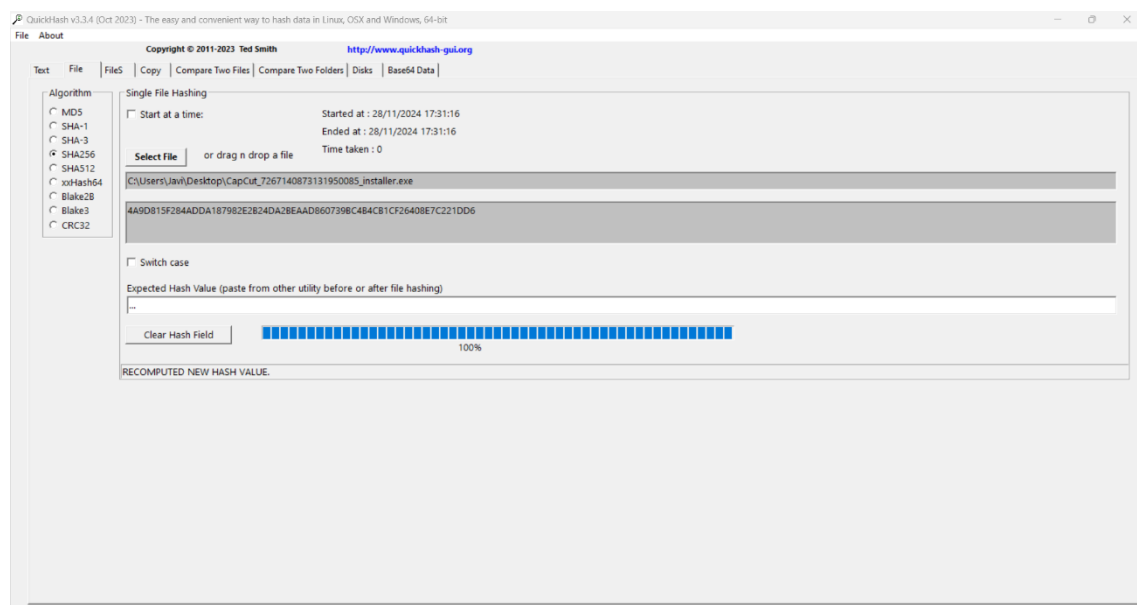
### MD5

Resultado: C916097550EA6CCEDF59208883414E0D



### SHA256

Resultado: 4A90815F284ADDA187982E2BZ4DAZBEAAD6607398C4B4C81CF26408E7C221DD6



¿Te parecieron fáciles de usar?

QuickHash es muy sencilla de usar ya que es todo gráfico y únicamente tienes que seleccionar el algoritmo y el archivo, cualquier persona podría hacerlo fácilmente. Con los comandos en PowerShell, es un poco más complicado porque tienes que tener un mínimo de idea, pero la verdad es que las dos son sencillas de utilizar.

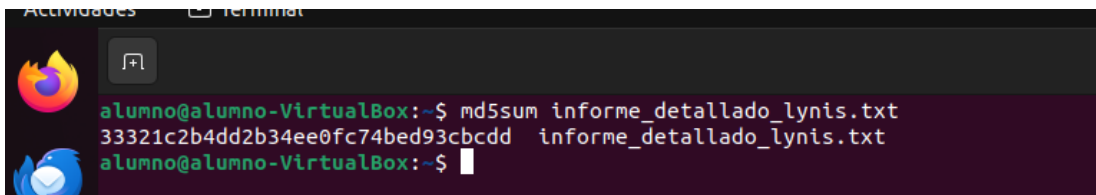
¿Cuál es la ventaja de tener una interfaz gráfica?

Tener una interfaz gráfica hace que todo sea más sencillo, ya que viene todo indicado y no tienes que saber un mínimo para poder utilizarla.

## Práctica en Linux

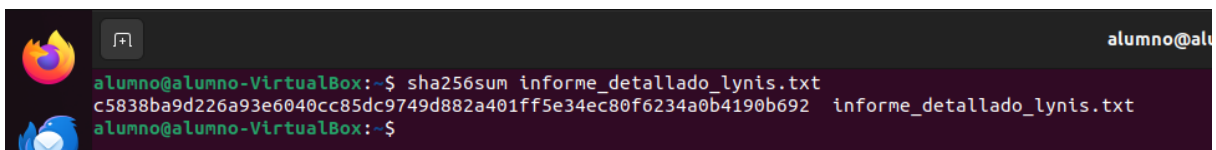
Linux es mucho más sencillo que Windows, simplemente con el comando “md5sum” para el algoritmo MD5 y “sha256sum” para el algoritmo SHA256 y el nombre del archivo ya nos mostraría el hash. El comando quedaría así “md5sum y el nombre del archivo”

Resultado: 33371c7b4dd7b34ee6fc74bed93cbcdd

A screenshot of a Linux terminal window. The window has a dark background with a terminal icon in the top left corner. The prompt is 'alumno@alumno-VirtualBox:~\$'. The command 'md5sum informe\_detallado\_lynis.txt' has been entered, and the output '33371c7b4dd7b34ee6fc74bed93cbcdd informe\_detallado\_lynis.txt' is displayed on the next line. The prompt is now 'alumno@alumno-VirtualBox:~\$' with a cursor.

SHA256

Resultado: c5838ba9d226a93e6048cc85dc9749d887a401ff5e34ec8bf6234a8b4198b697

A screenshot of a Linux terminal window. The window has a dark background with a terminal icon in the top left corner. The prompt is 'alumno@alumno-VirtualBox:~\$'. The command 'sha256sum informe\_detallado\_lynis.txt' has been entered, and the output 'c5838ba9d226a93e6048cc85dc9749d887a401ff5e34ec8bf6234a8b4198b697 informe\_detallado\_lynis.txt' is displayed on the next line. The prompt is now 'alumno@alumno-VirtualBox:~\$' with a cursor.

## ¿Cómo podrías utilizar estos comandos para verificar la integridad de un archivo descargado de internet?

Muchas compañías ofrecen el hash de sus archivos por lo que podríamos comparar el hash que nos dan en la web del archivo con el hash que nos da a nosotros utilizando los comandos o aplicaciones.

Por ejemplo Ubuntu nos ofrece el hash de sus isos.

<https://releases.ubuntu.com/14.04.6/SHA1SUMS>

```
2d3675f14a6884bb42917838a5c4246916fe73b5 *ubuntu-14.04.6-desktop-amd64.iso
91a72f4b623b3bc38a3698eed30e6241b42c8cfd *ubuntu-14.04.6-desktop-i386.iso
13bfe163ca8ad8a6e5676b0460ca60d03387ec24 *ubuntu-14.04.6-server-amd64.iso
17207306647b63f53938266e4726b5604250aba3 *ubuntu-14.04.6-server-i386.iso
bfc1ff3446b8cb49b6481372a2edf1c3861ba727 *wubi.exe
```

Con esto podríamos comprobar perfectamente la integridad de las isos.

## ¿Por qué los algoritmos MD5 y SHA-1 ya no son recomendados para aplicaciones críticas?

MD5 y SHA-1 ya no se recomiendan porque tienen vulnerabilidades que permiten crear colisiones, es decir, generar dos entradas diferentes que producen el mismo hash. Esto compromete la seguridad ya que alguien podría falsificar datos o archivos y hacer que parezcan legítimos.

Por ejemplo, si usas SHA-1 para firmar un documento, un atacante podría crear otro documento con el mismo hash y engañar al sistema para que lo acepte como auténtico. Esto sería grave en contratos digitales o actualizaciones de software.

## Indica en qué situaciones podría ser aceptable utilizar MD5 en lugar de algoritmos más seguros como SHA-256 o SHA-512.

MD5 podría ser aceptable en situaciones donde la seguridad no sea grave, como verificar la integridad de archivos descargados, pero no se debería usar en aplicaciones donde la autenticidad o la protección contra ataques sea importante, ya que es vulnerable a manipulaciones.