

Adrián Conesa Guirao

U3 T0 - Verificación de la integridad de  
archivos mediante funciones hash

## Contenido

Práctica en Windows:.....	3
Utilizando la herramienta <b>CertUtil</b> en Windows, calcula el hash de un archivo que tengas disponible en tu ordenador .....	3
Instala <b>HashTab</b> o <b>QuickHash GUI</b> y genera el hash de un archivo usando uno de estos programas. Explica brevemente tu experiencia con estas herramientas .....	3
¿Te parecieron fáciles de usar? .....	4
¿Cuál es la ventaja de tener una interfaz gráfica? .....	4
Práctica en Linux: .....	4
Utilizando una máquina virtual con Linux o una distribución en modo live, calcula el hash de un archivo usando los siguientes comandos:.....	4
<b>md5sum</b> para generar el hash MD5. ....	4
<b>sha256sum</b> para generar el hash SHA-256.....	4
Explica cómo podrías utilizar estos comandos para verificar la integridad de un archivo descargado de internet. ....	5
Comparación de Algoritmos .....	5
Investiga y responde: ¿Por qué los algoritmos <b>MD5</b> y <b>SHA-1</b> ya no son recomendados para aplicaciones críticas? Da un ejemplo de una situación en la que el uso de estos algoritmos podría representar un riesgo. ....	5
Ejemplo: .....	6

## Práctica en Windows:

Utilizando la herramienta **CertUtil** en Windows, calcula el hash de un archivo que tengas disponible en tu ordenador

Resultados Powershell MD5:

```
Administrador: Símbolo del sistema

C:\Users\adrai\Downloads>certutil -hashfile Texto.txt MD5
MD5 hash de Texto.txt:
90646df623dc4e4a07e22fdae4787586
CertUtil: -hashfile comando completado correctamente.
```

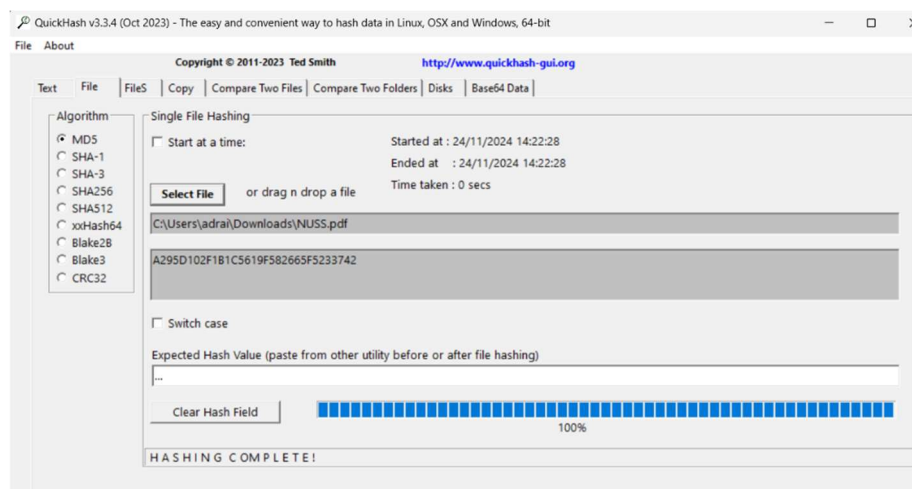
Resultados Powershell SHA256:

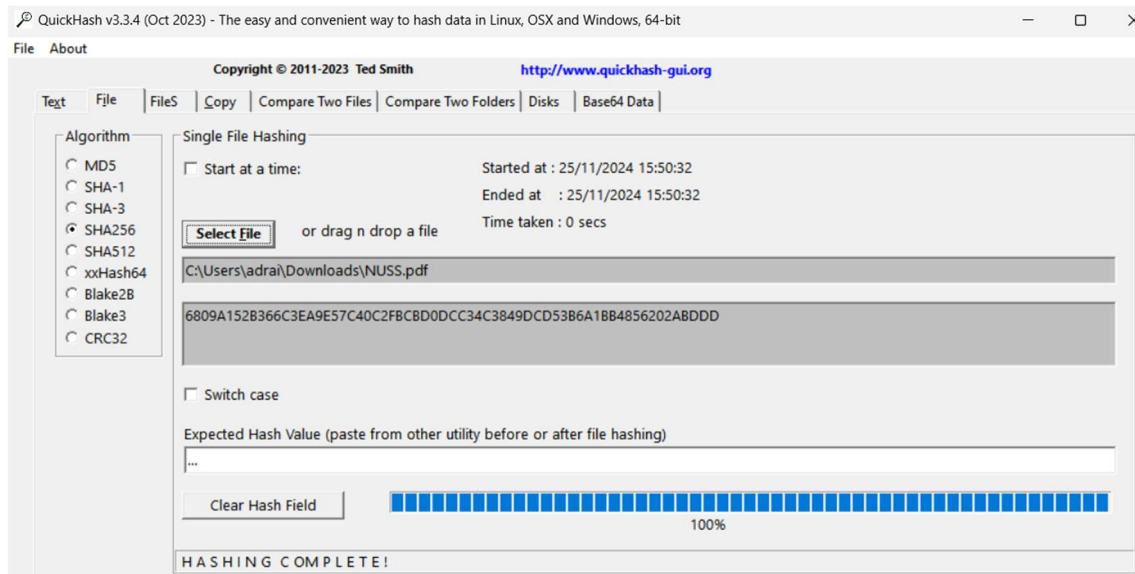
```
C:\Users\adrai\Downloads>certutil -hashfile Texto.txt SHA256
SHA256 hash de Texto.txt:
7893cf69fb50faeb6577a04567f058a5f9a704417ad6e930d0a130a8911ae9b0
CertUtil: -hashfile comando completado correctamente.

C:\Users\adrai\Downloads>_
```

Instala **HashTab** o **QuickHash GUI** y genera el hash de un archivo usando uno de estos programas. Explica brevemente tu experiencia con estas herramientas

Resultados QuickHash GUI:





¿Te parecieron fáciles de usar?

Sí es una herramienta bastante básica en cuanto a funcionamiento y muy fácil de entender sin haberla usado antes.

¿Cuál es la ventaja de tener una interfaz gráfica?

La comodidad a la hora de hacerlo

Práctica en Linux:

Utilizando una máquina virtual con Linux o una distribución en modo live, calcula el hash de un archivo usando los siguientes comandos:

**md5sum** para generar el hash MD5.

**sha256sum** para generar el hash SHA-256.

Resultado md5sum:

```
adrian@adrian-VirtualBox: ~/Descargas
adrian@adrian-VirtualBox:~/Descargas$ md5sum SAD.txt
86cf08c4fd00616eac40ed64469d7333 SAD.txt
adrian@adrian-VirtualBox:~/Descargas$
```

Resultado sha256sum:

```
adrian@adrian-VirtualBox: ~/Descargas
adrian@adrian-VirtualBox:~/Descargas$ md5sum SAD.txt
86cf08c4fd00616eac40ed64469d7333 SAD.txt
adrian@adrian-VirtualBox:~/Descargas$ sha256sum SAD.txt
f715c0bf7a3cb58cb8463bc4c288825d5d36c17112b781a2cfe8c3562827a624 SAD.txt
adrian@adrian-VirtualBox:~/Descargas$
```

Explica cómo podrías utilizar estos comandos para verificar la integridad de un archivo descargado de internet.

Por ejemplo Ubuntu proporciona sus hash.

Para que los compruebes cuando has descargado la iso de instalación, lo mejor sería que descargases el .txt de su página web donde indica el hash y el tipo de algoritmo, con este sencillo script lo podrías comprobar sin necesidad de descargar el archivo:

```
adrian@adrian-VirtualBox: ~
adrian@adrian-VirtualBox:~$ if [ "$(sha256sum "/home/adrian/Descargas/ubuntu-24.04.1-desktop-amd64.iso" | awk '{print $1}')" = "c2e6f4dc37ac944e2ed507f87c6188dd4d3179bf4a3f9e110d3c88d1f3294bdc" ]; then echo "Son iguales"; else echo "No son iguales"; fi
Son iguales
adrian@adrian-VirtualBox:~$
```

## Comparación de Algoritmos

Investiga y responde: ¿Por qué los algoritmos **MD5** y **SHA-1** ya no son recomendados para aplicaciones críticas? Da un ejemplo de una situación en la que el uso de estos algoritmos podría representar un riesgo.

MD5 y SHA-1 ya no son seguros debido a su vulnerabilidad a los ataques de colisión. El uso de estos algoritmos en aplicaciones críticas puede poner en riesgo la integridad de los datos, la autenticidad de las comunicaciones y la seguridad de los sistemas en general. Por lo tanto, es fundamental migrar a funciones hash más modernas y robustas como SHA-2 o SHA-3.

### Ejemplo:

Una empresa utiliza SHA-1 para verificar que los archivos descargados de un proveedor sean los mismos archivos que el proveedor coloca en el servidor. En este caso, un atacante podría generar un archivo malicioso que genere el mismo valor hash que un archivo legítimo, luego puede reemplazar el archivo legítimo en el servidor del proveedor con el archivo que ha creado él.

Cuando la empresa obtenga el archivo del servidor del proveedor, el valor hash coincidirá y, por lo tanto, la empresa creará que el archivo es legítimo y ejecutará el archivo, exponiendo así su sistema al compromiso.