

## MultiHash\_V8

- MultiHashV8 es una herramienta de línea de comandos escrita en Bash, diseñada para la generación simultánea de múltiples hashes de archivos, incluyendo MD5, SHA1, SHA256 y SHA512 integrando un análisis de metadatos. Esta utilidad está dirigida a profesionales de ciberseguridad, administradores de sistemas, auditores forenses, y cualquier usuario que necesite verificar la integridad de archivos mediante funciones criptográficas hash.

## Descripción General

- MultiHashV8 permite analizar uno o varios tipos de archivos al mismo tiempo, con un porcentaje dinámico de cálculo, indicándole la ruta precisa del file desde la misma herramienta, además de incorporar una función para el análisis de metadatos, utilizando 'exiftool', de esta manera genera un resumen estructurado de sus huellas digitales. Este recurso cuenta con 7 funciones que podemos emplear.

## Modo de uso

# Para ejecutar la herramienta se emplea el comando: **bash MultihashV8** y listo, ya podrás ingresar a la ruta en donde se encuentra el archivo o el directorio al cual quieres calcular el Hash. ( Ejemplo de ruta: [/home/Kali/Documents/Archivo.txt](#) ).

### # Ejemplo de salida

===== MultiHashV8 =====

Archivo: prueba.txt

MD5 : d41d8cd98f00b204e9800998ecf8427e

SHA1 : da39a3ee5e6b4b0d3255bfef95601890afd80709

SHA256 : e3b0c44298fc1c149afb4c8996fb924...

SHA512 : cf83e1357eefb8bdf1542850d66d8007...

## # Opciones a selección

- 1) **Cambiar archivo o directorio:** Aca podrás indicar la ruta en la que se encuentra el archivo a calcular, esta opción abarca files específicos como rutas de directorio. (Ejemplo: [/home/Kali/Documents/Archivo.txt](#) como [/home/Kali/Documents](#)), este último calculara los hashes de todos los archivos que estén dentro del directorio.
  - 2) **Calcular haches:** En esta opción, podrás calcular nuevamente el mismo archivo o directorio en la ruta especificada.
  - 3) **Validar hash manualmente:** Aquí puedes ingresar el algoritmo hash solo escribiéndolo, (**Nota:** Debido a la cantidad enorme de caracteres que integra un hash sha-512 o sha-256 puede resulta tediosos el proceso, por lo que se debe emplear solo es ocasiones que realmente lo requiera).
  - 4) **Validar desde archivo .txt:** Aca deberás ingresar la ruta del archivo, que almacena el código hash que deseas validar, con el archivo ya calculado, una vez ingresada la ruta, te pedirá que elijas las opciones del algoritmo a validar. (**Nota:** Esta función esta en fase de desarrollo aun, pronto ya estará disponible).
  - 5) **Exportar los resultados:** Esta función generara 3 reportes en diferentes formatos .txt - .log - .json, estos documentos se descargarán cuando el proceso de cálculo finalice con el archivo ingresado. Se almacenarán automáticamente en la ruta [/home/Kali/Downloads](#).
  - 6) **Analizar metadatos de archivos:** Con esta opción podrás analizar e identificar metadatos con soporte para múltiples formatos de archivos. Integrando la herramienta exiftool. Solo debes ingresar la ruta en donde se aloja el archivo por examinar.
  - 7) **Salir:** Como indica el nombre, con esta opción podrás salir de la herramienta.
-

- Esta capacidad resulta especialmente útil en tareas como:
  - Verificación de integridad tras transferencias de archivos.
  - Calcular hash para múltiples formatos de archivos
  - Generación de firmas digitales.
  - Comparación de hashes para detección de alteraciones.
  - Extracción y análisis de metadatos

## **Funcionalidades**

- Soporte para múltiples archivos como parámetros.
- Cálculo simultáneo de hashes con formato amigable.
- Validación de errores por archivo inexistente o sin permisos.
- ASCII Art de bienvenida incluido.
- Mensajes de advertencia y validación en tiempo real.
- Reconocimiento de metadatos.

# Esta herramienta fue desarrollada en código bash con objetivos de compatibilidad, portabilidad y simplicidad, previniendo dependencias adicionales fuera de las herramientas estándar presentes en la mayoría de las distribuciones GNU/Linux.

## **Requisitos Recomendados de Software**

# Este script utiliza varios comandos externos, por lo tanto, el sistema operativo (**Kali Linux**) debe tener instaladas las siguientes herramientas:

### 1. Hashing:

- md5sum
- sha1sum
- sha256sum
- sha512sum

### 2. Utilidades de archivos y sistema:

- realpath
- stat
- find
- file

### 3. Metadatos:

- **exiftool** (de *Perl Image::ExifTool*)
- **identify** (parte del paquete **ImageMagick**)

# Puedes instalarlas en distribuciones basadas en Debian/Ubuntu con:

- bash

Comandos
<b>sudo apt update -y</b>
<b>sudo apt install coreutils findutils exiftool imagemagick -y</b>

- en Fedora/RHEL:

Comando
<b>sudo dnf install coreutils findutils perl-Image-ExifTool ImageMagick</b>

---



## Requisitos Recomendados de Hardware

# Aunque el script no es intensivo en recursos, para un rendimiento fluido en grandes volúmenes de archivos:

- **CPU:** 1 núcleo mínimo (2 núcleos recomendado)
  - **RAM:** 1 GB mínimo (2 GB recomendado si analizas archivos grandes o muchos a la vez)
  - **Almacenamiento:** Al menos 100 MB libres para reportes y archivos temporales (más si los archivos analizados son pesados)
  - **Pantalla compatible con colores ANSI** (la mayoría de las terminales modernas ya lo son)
- 



## Instalación

# Opción local (entorno de desarrollo o pruebas)

- `git clone https://github.com/BitThor/MultiHashV8.git`
  - `cd MultiHashV8`
  - `chmod +x MultiHashV8`
  - `./MultiHashV8 archivo.txt`
- 

“ Para su ejecución desde cualquier ruta (recomendada para su uso diario)

- `sudo mv MultiHashV8 /usr/local/bin/`

## **Contribuciones**

# Este proyecto es de código abierto y está disponible para su optimización y mejora.  
Puedes contribuir mediante:

- I. Fork del repositorio.
- II. Creación de una rama de desarrollo.
- III. Pull request con tu propuesta de cambio.

## **Autor**

Desarrollado por [ **Victor Cavada Hernández** ].

Contacto: viitooko.166@gmail.com

Año: 2025