¿Qué es el Hash?

1. ¿Qué es el Hash (Función Hash)?

El **Hash** (o **Función Hash Criptográfica**) es una operación matemática que toma una entrada de cualquier tamaño (un archivo, una contraseña, un documento, etc.) y produce una **cadena de caracteres alfanuméricos de longitud fija y única**, conocida como **valor hash**, **código hash** o **resumen digital (*digest*)**.

Propiedades Clave del Hash Criptográfico:

- **Determinismo:** La misma entrada siempre produce la misma salida hash.
- **Irreversibilidad (Unidireccional):** Es prácticamente imposible recrear la entrada original a partir del valor hash. Es una función de "un solo sentido".
- **Resistencia a Colisiones (Collision Resistance):** Es extremadamente difícil encontrar dos entradas diferentes que produzcan el mismo valor hash.
- **Efecto Avalancha (*Avalanche Effect*):** Un cambio mínimo en la entrada (incluso un solo bit) produce un cambio masivo y completamente diferente en el valor hash de salida.

2. Tipos y Familias de Algoritmos Hash

Los algoritmos hash se agrupan en familias, que se distinguen por la longitud del *digest* que producen y su nivel de seguridad.

A. Familia MD (Message Digest)

Algoritmo	Longitud de Salida	Seguridad	Estado Actual
MD5	128 bits	Baja	**ROTO/Obsoleto.** Se han encontrado colisiones de forma práctica. No usar para seguridad crítica (ej. firmas digitales).

B. Familia SHA (Secure Hash Algorithm)

Esta es la familia más utilizada y se considera el estándar de la industria.

Algoritmo	Longitud de Salida	Seguridad	Uso Típico
SHA-1	160 bits	Media	**Obsoleto.** Se considera débil y ha sido reemplazado.
SHA-256	256 bits	Alta	**Estándar actual.** Utilizado en *blockchain* (Bitcoin) y certificados SSL/TLS.
SHA-512	512 bits	Muy Alta	Usado donde se requiere una seguridad extrema o donde la longitud no es una restricción.
SHA-3	Variable	Muy Alta	Alternativa moderna al SHA-2. Diseñado para ser más resistente a futuros ataques.

3. Diferencias Clave: Hash vs. Cifrado

Es fundamental distinguir el *hashing* del cifrado, ya que ambos manipulan datos, pero con objetivos distintos:

Característica	HASHING (Resumen)	CIFRADO (Encryption)
Propósito Principal	**Verificar la integridad** de los datos.	**Garantizar la confidencialidad** de los datos.
Reversibilidad	**Unidireccional (Irreversible).** No se puede descifrar.	**Bidireccional (Reversible).** Se puede descifrar con una clave.
Salida (Longitud)	**Fija.** 128, 256, 512 bits, etc. (Ej: SHA256 siempre da 64 caracteres).	**Variable.** Generalmente del mismo tamaño o ligeramente mayor que la entrada.
Uso Común	Contraseñas, archivos de integridad (como AIDE), firmas digitales.	Comunicaciones seguras (HTTPS), bases de datos protegidas.

Propuesta de Práctica: Entender la Integridad con AIDE

Para demostrar el funcionamiento del hash y el **efecto avalancha**, puedes usar el entorno de AIDE que acabas de configurar.

Objetivo:

Demostrar cómo AIDE (que usa funciones hash) detecta incluso el cambio más pequeño en un archivo crucial del sistema.

Materiales:

- Tu máquina virtual Linux con AIDE instalado y la base de datos **aide.db** ya inicializada.
- Un editor de texto (como nano o vi).

Pasos de la Práctica:

Paso 1: Establecer la Base de Oro (Ya Realizado)

Asegúrate de que la base de datos fue movida correctamente.

ls -l /var/lib/aide/aide.db

Paso 2: Ejecutar la Verificación de Referencia

Ejecuta una comprobación rápida para establecer el punto de partida (debería mostrar solo los cambios esperados en los *logs*):

sudo aide --config /etc/aide/aide.conf --check

Paso 3: Introducir un Cambio Mínimo (El Ataque)

Edita un archivo de configuración crítico y haz un cambio **mínimo**, como añadir un espacio o una línea vacía al final del archivo. Usaremos el archivo de configuración del servidor SSH que revisaste anteriormente.

sudo nano /etc/ssh/sshd_config

Añade una línea en blanco al final del archivo y guarda (Ctrl+O, luego Enter, luego Ctrl+X).

Paso 4: Ejecutar la Verificación Post-Cambio

Vuelve a ejecutar el chequeo de AIDE:

```
sudo aide --config /etc/aide/aide.conf --check
```

Resultado Esperado y Explicación

El resultado de la verificación mostrará un informe que indica que el archivo /etc/ssh/sshd_config ha sido **MODIFICADO**.

La explicación es la siguiente:

1. AIDE leyó el archivo /etc/ssh/sshd_config.
2. Calculó un nuevo hash (por ejemplo, SHA-256) para el archivo con el espacio añadido.
3. Debido al **Efecto Avalancha**, el nuevo hash es **completamente diferente** del hash guardado en aide.db.
4. AIDE concluye inmediatamente que el archivo ha sido **manipulado**, aunque solo se haya cambiado un solo carácter.

Esto demuestra de forma práctica que el *hashing* es el mecanismo que garantiza la **integridad de los datos**: si el hash coincide, el archivo es idéntico; si el hash es diferente, el archivo ha sido modificado.

https://araintel.com/calculadora-hash/