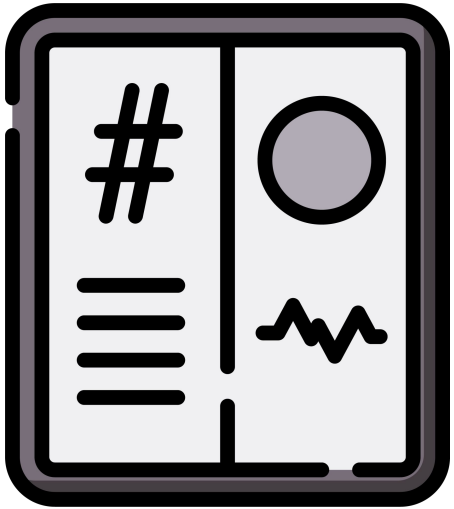
Usar la criptografía para validar integridad

DigitalHouse>

Certified Tech Developer
The Ultimate Degree

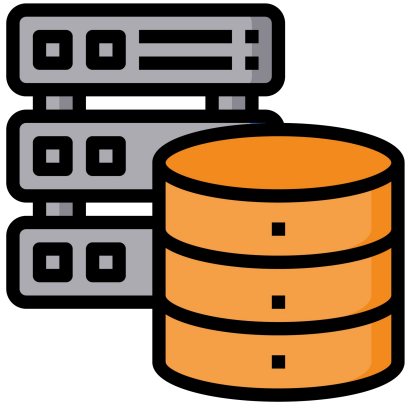
Validar la integridad

La idea de verificar la integridad de un dato usando la criptografía es sencilla. El dato para el cual queremos validar su integridad es sometido a un algoritmo (los más conocidos para este propósito son SHA y MD5), el resultado es una cadena de caracteres que se conoce como **hash**. De alterarse los datos, al volver a calcular el hash, estos producirían un resultado diferente. Al compararlo con el hash original, la diferencia sería evidente.



Integridad para “Data at Rest”

Para escenarios donde los datos no están en uso a en tránsito (por ejemplo, un repositorio de archivos), la forma más habitual de garantizar la integridad de los datos es calcular el hash, almacenarlo. En una instancia posterior cuando el dato deba ser consultado, se puede verificar el hash y así determinar que los contenidos del archivo no fueron alterados. Una ejemplo de esto es cuando vamos a descargar un instalador de Internet y la página nos provee con el hash para que una vez descargado el software podamos verificar su integridad y autenticidad.



¡A probarlo!

1) En tu computadora crear un archivo de texto plano e ingresar algún contenido, guardarlo y cerrarlo.

1) Abrir una consola de PowerShell y ejecutar el siguiente comando:

```
01  Get-FileHash -Algorithm MD5 -Path <ruta al archivo> | Select-Object -Property Hash
```

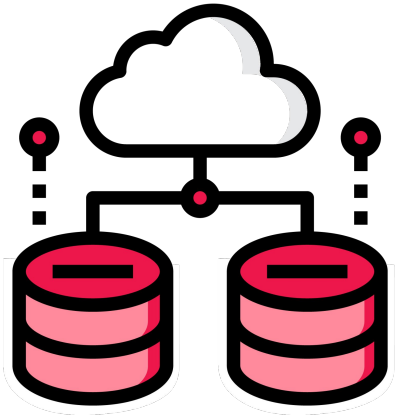
1) Ahora editar el archivo y modificar el contenido, guardarlo nuevamente y cerrarlo.

1) Volver a ejecutar el comando que vimos en el paso 2.

¿Qué pasó?

Integridad para “Data in transit”

Dado un escenario donde 2 computadoras van a intercambiar mensajes, por ejemplo, cuando una computadora se conecta por medio de una VPN a un servidor, mediante la criptografía también podemos verificar la integridad de cada uno de los paquetes que vamos a intercambiar. Por cada paquete que vamos a enviar mediante la red, podemos calcular el hash que le corresponda y agregar el resultado a uno de los encabezados del paquete. El servidor del otro lado, recibe el paquete, lo somete al mismo algoritmo y de producir el mismo hash, podemos determinar que el paquete es íntegro.



Problemáticas de “Data in transit”

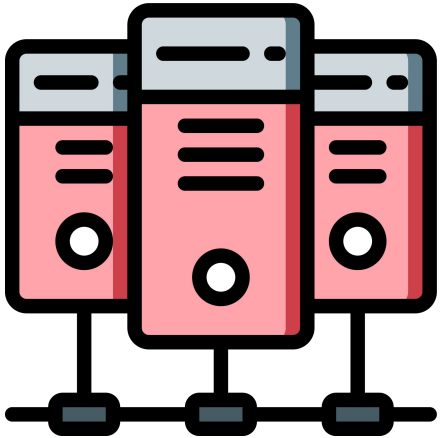
Hasta ahora vimos que los algoritmos que se utilizan para calcular el hash toman como entrada una única variable que son los datos a transmitir y, luego, adjuntan a esos datos el hash.

¿Qué sucede si un atacante altera los datos, calcula el hash para el nuevo contenido y reemplaza el hash original con el nuevo?
El servidor que recibe los datos verificará el hash y coincidirá con aquel que el atacante ha generado.

¿Cómo resolvemos este problema?
¡HMAC al rescate!

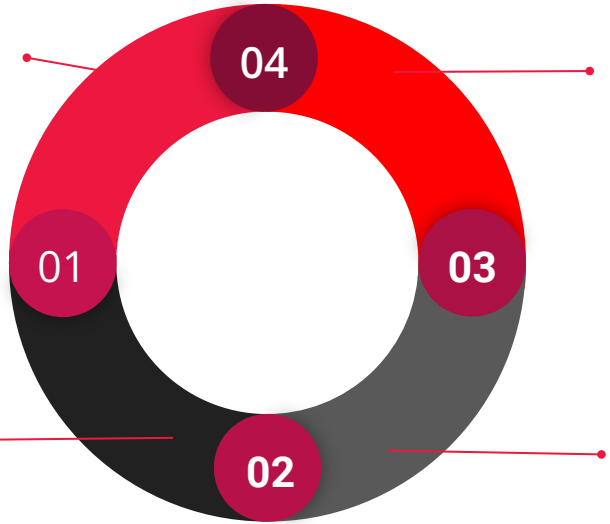
HMAC (*hash message authentication code*)

HMAC se usa en escenarios de datos en tránsito donde podemos usar una clave simétrica como entrada adicional para el algoritmo de hashing. Es decir que para producir el hash de ambos lados (emisor y receptor) ambos deben conocer la clave.



Paso 1: acordar una clave privada
El emisor y receptor de la comunicación se ponen de acuerdo en qué clave privada van a utilizar y en qué algoritmo criptográfico utilizaran para producir el hash. El primer dato es secreto mientras que los algoritmos son de dominio público y, por lo tanto, no son secretos.

Paso 2: calcular el hash del paquete
El emisor calcula el hash correspondiente a cada paquete enviado utilizando los datos del mismo y la clave acordada como datos de entrada del algoritmo que producirá el hash. A continuación, adjunta el resultado como un encabezado del paquete a enviar.



Paso 4: Recibir el paquete
El receptor toma los datos (que vienen en el paquete), toma la clave acordada en el paso 1, y utiliza ambos datos como elementos de entrada para el algoritmo que producirá el hash. Una vez obtenido, si el cálculo realizado por el receptor coincide con el hash incluido en el paquete, podemos decir que el mismo no ha sufrido alteraciones desde su envío.

Paso 3: transmitir el paquete
Los paquetes son transmitidos de un extremo al otro.

DigitalHouse>