Contenido

[**Introducción** 1](#_Toc177849531)

[Autenticación Multifactor (MFA) 1](#_Toc177849532)

[Contraseña de un solo uso (OTP) 1](#_Toc177849533)

[**Métodos de autentificación** 1](#_Toc177849534)

[Ataques 1](#_Toc177849535)

[SIM Swapping: 1](#_Toc177849536)

[Keylogging: 2](#_Toc177849537)

[Credential Stuffing: 2](#_Toc177849538)

[Ingeniería Social: 2](#_Toc177849539)

[Phishing: 2](#_Toc177849540)

[Machine-in-the-Middle Attacks 2](#_Toc177849541)

[**Herramientas y prácticas de seguridad:** 2](#_Toc177849542)

[Defensa 2](#_Toc177849543)

[Single Sign-On (SSO): 2](#_Toc177849544)

[Gestores de contraseñas: 2](#_Toc177849545)

[PassKeys: 3](#_Toc177849546)

# **Introducción**

## Autenticación Multifactor (MFA)

Categorías:

* Knowledge: Claves o contraseñas, conocimiento.
* Possession: Algo físico, como un dispositivo.
* Inherence: Algo único y personal, como datos biométricos.

Contraseña de un solo uso (OTP)  
Un OTP (One-Time Password) es una contraseña que se utiliza una única vez, como los códigos generados por tokens, que cambian cada poco segundo.

# **Métodos de autentificación**

## Ataques

SIM Swapping:  
Cada tarjeta SIM tiene un identificador único. El problema es que la SIM puede ser sustituida si alguien engaña al proveedor de servicios, redirigiendo las comunicaciones al teléfono del atacante. Por esta razón, es más seguro utilizar aplicaciones que no dependan de SMS, ya que esta amenaza es potencialmente peligrosa.

Keylogging:  
El keylogging es un malware que registra las pulsaciones del teclado, lo que permite robar credenciales como usuarios y contraseñas. Este tipo de ataque puede ocurrir al instalar software no confiable o infectado por virus. Una defensa clave es evitar usar dispositivos que no sean los personales, lo que reduce el riesgo de exposición.

Credential Stuffing:  
En este ataque, los hackers utilizan credenciales obtenidas de bases de datos filtradas en línea para intentar acceder a otras cuentas, aprovechando que muchas personas reutilizan contraseñas. Por ello, es fundamental no reutilizar contraseñas.

Ingeniería Social:  
Ataque basado en la manipulación de la confianza entre personas. Es crucial no compartir información personal sensible con terceros.

Phishing:  
Ataque técnico derivado de la ingeniería social, generalmente a través de correos electrónicos falsos que intentan hacer pasar sitios web fraudulentos por legítimos. Estos ataques suelen incluir preguntas que buscan obtener información personal, como colores o canciones favoritas. Es esencial verificar siempre las URL o escribirlas manualmente en el navegador. Es importante equilibrar la seguridad y la usabilidad.

### Machine-in-the-Middle Attacks

Ataques de Hombre en el Medio. A medida que la información viaja a través de internet mediante routers y servidores, existe el riesgo de que un intermediario malintencionado capture los datos. Si esto ocurre, la información puede verse comprometida.

# **Herramientas y prácticas de seguridad:**

## Defensa

Single Sign-On (SSO):  
El SSO permite iniciar sesión utilizando una cuenta de otro servicio, como Google o Facebook. Esto ahorra tiempo al no tener que crear y recordar nuevas contraseñas, y mejora la seguridad al gestionar los accesos mediante criptografía y cifrado.

Gestores de contraseñas:  
Estos permiten generar y almacenar contraseñas seguras automáticamente, además de recordar las URL de los sitios web. Algunos ejemplos integrados son Apple iCloud Keychain, Google Password Manager y Microsoft Credential Manager. Estos gestores requieren una contraseña maestra para acceder a todas las demás.

PassKeys:  
Las passkeys son claves generadas por los dispositivos, que ayudan a mitigar las vulnerabilidades de los gestores de contraseñas. Con ellas, los dispositivos pueden autenticarse de forma automática.

**Seguridad de los Datos y Gestión de Contraseñas**

El uso de contraseñas en formato usuario:contraseña es vulnerable a ataques. Para asegurar que las contraseñas no se almacenen de manera insegura, se utiliza la técnica del **hashing**. Esta convierte la contraseña en un valor hash mediante un algoritmo como **SHA-256** o **bcrypt**. El hash generado tiene una longitud fija, y este valor se almacena en la base de datos, en lugar de la contraseña en texto plano, protegiendo así los datos en caso de un ataque. El hash actúa como una "caja negra": incluso si un atacante obtiene acceso, no puede revertir el proceso de hashing fácilmente para obtener la contraseña original.

Al realizar el proceso de autenticación, el servidor aplica el mismo algoritmo de hash a la contraseña ingresada y compara el resultado con el hash almacenado. Si coinciden, la contraseña es correcta.

**Tipos de Ataques:**

* **Ataques de diccionario (Dictionary Attacks):** Consisten en probar una lista de contraseñas comunes y calcular su hash hasta encontrar una coincidencia.
* **Ataques de fuerza bruta (Brute-Force Attacks):** Este método prueba todas las combinaciones posibles de contraseñas, lo que aumenta el tiempo y los recursos necesarios para adivinar una contraseña.
* **Rainbow Tables:** Son tablas precalculadas de hashes y sus contraseñas correspondientes, lo que permite comparar hashes sin necesidad de calcularlos desde cero. Aunque eficientes, las tablas arcoíris pueden ser mitigadas con salting.

**Coincidencia de Contraseñas y Salting**

Otro problema importante es la colisión de contraseñas, donde usuarios diferentes pueden tener el mismo hash si sus contraseñas son idénticas. Para prevenir esto, se usa el salting, que agrega un valor aleatorio (llamado sal) a cada contraseña antes de generar el hash. Esto asegura que contraseñas iguales produzcan hashes diferentes. La sal generalmente se almacena junto con el hash, y su presencia entre símbolos de dólar ($) indica el tipo de algoritmo de hash utilizado.

Cuando un sitio permite recuperar contraseñas enviándolas al correo electrónico, esto es una señal de un mal manejo de seguridad, ya que implica que las contraseñas se almacenan en texto claro en el servidor, algo inaceptable debido a la irreversibilidad esperada de los hashes.

**Funciones Hash Criptográficas**

Las one-way hash functions (funciones hash unidireccionales) son esenciales en la seguridad de los datos. Toman como entrada una cadena de texto y devuelven un hash de longitud fija, lo que garantiza la irreversibilidad y la seguridad del almacenamiento de contraseñas.

**Códigos y Cifradores**

En criptografía, es importante distinguir entre códigos (codes) y cifrados (ciphers):

* Los **codes** son asignaciones de palabras o frases a claves específicas, pero no son directamente cifrados. Un ejemplo sería un libro de códigos donde cada palabra tiene una clave asociada.
* Los **ciphers** son algoritmos que convierten texto claro en texto cifrado. El proceso de cifrado se llama encipher y su reverso es decipher, que convierte el texto cifrado nuevamente en texto claro.

**Cifrado Simétrico y Asimétrico**

* **Cifrado simétrico (Symmetric-key encryption):** Ambas partes (A y B) comparten una misma clave secreta para cifrar y descifrar la información. Ejemplo: AES (Advanced Encryption Standard) y Triple DES.
* **Cifrado de clave pública (Public-key cryptography):** Utiliza dos claves distintas: una pública para cifrar y una privada para descifrar. Este método es más seguro, ya que solo el destinatario puede descifrar el mensaje. Un algoritmo común es RSA, que se basa en la factorización de números primos muy grandes.

**Intercambio de Claves (Key Exchange)**

El intercambio seguro de claves, como el protocolo Diffie-Hellman, permite que las partes generen una clave secreta compartida sin transmitirla directamente. Esto previene que un atacante intercepte la clave durante el intercambio.

**Firmas Digitales**

Las firmas digitales son un componente crucial en la criptografía de clave pública. Se utilizan para autenticar documentos mediante un algoritmo que emplea una clave privada y un valor hash. Esto garantiza la autenticidad e integridad del documento, asegurando que no ha sido alterado desde su firma.

Ejemplos de algoritmos utilizados para firmas digitales incluyen DSA (Digital Signature Algorithm) y RSA.

**Encriptación en Tránsito vs. Extremo a Extremo**

El cifrado en tránsito protege los datos mientras viajan a través de redes, como en los servicios de correo electrónico o plataformas como Zoom. Sin embargo, para garantizar que solo el remitente y el destinatario puedan acceder al contenido, se debe usar cifrado de extremo a extremo (end-to-end encryption).

**Eliminación Segura de Archivos**

Cuando se borra un archivo en un dispositivo, no se elimina realmente. El sistema solo olvida la ubicación del archivo, pero los datos permanecen hasta que sean sobrescritos. Para eliminar de manera segura, es necesario sobrescribir el archivo con ceros o utilizar herramientas específicas de eliminación segura que aseguran que los datos no puedan ser recuperados.

**Encriptación Completa de Disco**

La encriptación de disco completo (full disk encryption) asegura que los datos en reposo sean inaccesibles sin la autenticación adecuada. En caso de robo del dispositivo, los datos aparecerán como información aleatoria sin la contraseña. Sin embargo, también existen riesgos como el ransomware, donde los atacantes cifran los datos del usuario y exigen un rescate.

**Criptografía y Computación Cuántica**

La computación cuántica representa una amenaza significativa para los actuales sistemas de seguridad, ya que los cubits permiten realizar cálculos exponencialmente más rápidos que los métodos tradicionales. Esto podría comprometer la seguridad de los algoritmos criptográficos actuales, como RSA, lo que impulsa la investigación en criptografía post-cuántica, que pretende resistir los avances en computación cuántica.

**Cifrado de Clave Pública**

En un sistema de cifrado de clave pública, se utilizan dos claves diferentes: una pública y una privada. La clave pública se distribuye libremente y puede ser utilizada por cualquier persona para cifrar mensajes dirigidos al dueño de la clave privada. Sin embargo, solo el propietario de esta clave privada puede descifrar dichos mensajes, garantizando así la confidencialidad de la comunicación. Esto resuelve el problema del intercambio de claves secretas, ya que no es necesario que las dos partes compartan una clave común de manera privada.

El algoritmo RSA es uno de los más populares para implementar cifrado de clave pública. RSA se basa en la dificultad computacional de factorizar números grandes. Cuando un mensaje se cifra utilizando la clave pública de un receptor, solo la clave privada del receptor puede descifrar ese mensaje. En aplicaciones prácticas, a menudo se utiliza en combinación con cifrados de clave simétrica para aprovechar la eficiencia de ambos métodos.

**Firmas Digitales**

Una firma digital es una técnica criptográfica que permite garantizar la autenticidad e integridad de un mensaje o documento. Utiliza una clave privada para generar una firma, y una clave pública para verificarla. El proceso de firma digital implica crear un resumen del documento (a través de un hash criptográfico), el cual se cifra utilizando la clave privada del firmante. El receptor puede descifrar este resumen utilizando la clave pública del remitente y comparar este valor con el hash calculado directamente sobre el documento recibido. Si ambos coinciden, se confirma que el documento no ha sido alterado y que proviene del titular de la clave privada.

Este mecanismo es esencial para sistemas como el correo electrónico seguro, contratos electrónicos y la autenticación en línea. La seguridad de las firmas digitales se basa en la imposibilidad práctica de falsificar la clave privada del firmante.

**Ataques de Diccionario**

Los ataques de diccionario son un tipo de ataque contra contraseñas o claves criptográficas. En lugar de probar todas las combinaciones posibles, como en un ataque de fuerza bruta, el atacante utiliza una lista predefinida de palabras comunes o combinaciones probables (el "diccionario") para intentar adivinar la clave o contraseña. Este tipo de ataque se aprovecha de la tendencia de las personas a usar contraseñas sencillas o comunes, como “123456” o “password”, lo que facilita el éxito del atacante. Para mitigar este tipo de ataques, es fundamental utilizar contraseñas complejas que combinen letras, números y caracteres especiales, así como implementar prácticas de seguridad como el "salting", que añade datos adicionales antes de cifrar la contraseña.

**Ataques de Fuerza Bruta**

Un ataque de fuerza bruta consiste en probar sistemáticamente todas las combinaciones posibles de una contraseña o clave hasta encontrar la correcta. Aunque este método garantiza eventualmente romper la seguridad, es extremadamente ineficiente cuando las claves son largas y complejas. La clave para resistir este tipo de ataques es aumentar el tamaño de las contraseñas o claves. Una contraseña más larga y compleja incrementa exponencialmente el número de posibles combinaciones, haciendo que el ataque requiera un tiempo prohibitivo para tener éxito.

**Tablas Arcoíris**

Las tablas arcoíris son una herramienta utilizada para atacar contraseñas cifradas. Funcionan almacenando hashes precomputados de muchas combinaciones posibles de contraseñas, lo que permite a un atacante comparar rápidamente el hash de una contraseña capturada con los valores en la tabla para descubrir la contraseña original. Para mitigar los ataques basados en tablas arcoíris, se utiliza una técnica llamada "salting", que consiste en agregar un valor único y aleatorio a cada contraseña antes de cifrarla. Esto hace que incluso las contraseñas idénticas tengan hashes diferentes, dificultando el uso de tablas arcoíris.

**Salting**

El "salting" es una técnica utilizada para proteger contraseñas contra ataques de diccionario y tablas arcoíris. Implica agregar un valor aleatorio, llamado "sal", a una contraseña antes de aplicar la función hash. Esto asegura que incluso si dos usuarios tienen la misma contraseña, sus hashes serán diferentes, lo que complica significativamente los ataques. Además, el sal evita que un atacante pueda usar tablas arcoíris precomputadas para descifrar las contraseñas, ya que el hash de la contraseña dependerá tanto del sal como de la contraseña original.

**Funciones Hash Criptográficas**

Las funciones hash criptográficas son fundamentales para muchas aplicaciones de seguridad. Son funciones matemáticas que transforman cualquier tipo de datos en una cadena de longitud fija (el resumen o hash). Estas funciones tienen varias propiedades críticas: son deterministas (el mismo mensaje siempre produce el mismo hash), son de una sola dirección (no es posible recuperar el mensaje original a partir del hash), y son resistentes a colisiones (es extremadamente difícil encontrar dos mensajes diferentes que generen el mismo hash).

Se utilizan comúnmente para verificar la integridad de los datos y como base para firmas digitales y contraseñas seguras. Ejemplos de funciones hash criptográficas incluyen MD5 (ya no recomendado debido a vulnerabilidades), SHA-256 y SHA-3.