Autenticación, contraseñas y firma digital

Marko Hölbl

Annotation

Este curso introduce el concepto de autenticación y contraseñas, así como los conceptos y tecnologías subyacentes. También se presenta el papel de la firma digital en la autenticación y sus conceptos y tecnologías de fondo.

Objectives

Este curso proporciona información básica sobre la autenticación, sus elementos y la autenticación basada en contraseña, y cómo proteger adecuadamente las contraseñas tanto del lado del usuario como del autenticador. Se analizan los conceptos de gestión de contraseñas, autenticación multi factor y autenticaciones sin contraseña.
Además, se presenta información sobre los antecedentes técnicos de la firma digital, incluidas las funciones hash, la criptografía de clave pública y la infraestructura de clave pública. Por último, se presenta la firma digital como medio de autenticación.

Keywords

contraseñas, autenticación, firma digital, infraestructura de clave pública, funciones hash

Date of Creation

06.01.2022

Duration

15 horas

Language

English

License

ISBN

Literature

[1] Batten, L. M. (2013). Public key cryptography: applications and attacks, John Wiley & Sons.

[2] Boonkrong, S. (2021). Authentication and Access Control: Practical Cryptography Methods and Tools, Springer.

[3] Buchmann, J., et al. (2013). Introduction to public key infrastructures, Springer.

[4] Burnett, M. (2006). Perfect password: Selection, protection, authentication, Elsevier.

[5] Grassi, P. A., et al. (2017). "NIST special publication 800-63b: digital identity guidelines." National Institute of Standards and Technology (NIST).

[6] Grimes, R. A. (2020). Hacking Multifactor Authentication, John Wiley & Sons.

CHAPTER 1

Introducción

DEFINITION

La autenticación es el proceso de verificar la identidad de alguien o algo.

Para ello se utiliza la información proporcionada por la entidad que debe ser comprobada. El proceso de autenticación en sistemas informáticos privados y públicos, como las redes de ordenadores, suele requerir que alguien, normalmente el usuario, utilice credenciales emitidas por el sistema para iniciar la sesión. El hecho de que un usuario tenga una contraseña supone que demuestra que es auténtico. El método de autenticación más común es la combinación de nombre de usuario y contraseña. Sin embargo, existen otras opciones de autenticación, como la biometría, las tarjetas inteligentes, los tokens de un solo uso, etc.

En la mayoría de los casos, la autenticación requiere la presentación de credenciales o de un objeto de valor para corroborar la afirmación de que uno es quien dice ser. Los objetos de valor o credenciales se basan en una serie de características distintivas que demuestran lo que uno sabe, tiene o es.

- **Algo que sabes:** Esto podría ser una posesión mental tuya, como una contraseña que tanto el usuario como el autenticador conocen. Aunque se trata de una solución administrativa rentable, es vulnerable a los fallos de memoria de las personas y a otros defectos, como el almacenamiento seguro de los archivos de contraseñas por parte de los administradores del sistema. El usuario puede utilizar la misma contraseña para todos los inicios de sesión del sistema. Las contraseñas, las frases de contraseña y los PIN (números de identificación personal) son ejemplos de este tipo de factores.

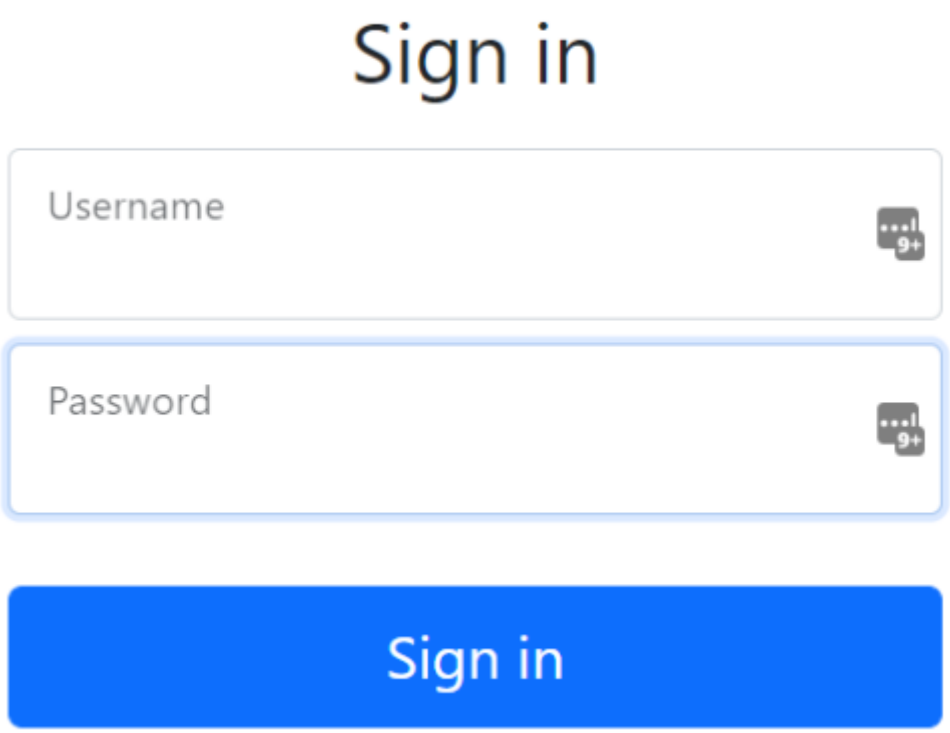


Fig. 1. Ejemplo de inicio de sesión de un usuario con nombre de usuario y contraseña.

- **Algo que tienes**: Puede ser cualquier tipo de *token* o etiqueta de autoidentificación emitida u obtenida, incluyendo tarjetas inteligentes, *tokens* de hardware, teléfonos móviles y una variedad de otros medios. Dado que las identificaciones físicas individuales son difíciles de abusar, esta forma es más segura que el primer enfoque (algo que sabes). Por ejemplo, perder una tarjeta inteligente es más difícil que recordar el número de la tarjeta.

Fig. 2. Ejemplos de *tokens* de hardware para el tipo de autenticación "Algo que tienes".

- **Algo que eres**: Se trata de un rasgo físico adquirido de forma natural, como una huella dactilar. Este tipo de autenticación se denomina biometría. Aunque la biometría es sencilla de utilizar, los costes de obtener lectores biométricos pueden suponer un problema. Las huellas dactilares, los patrones de retina, los patrones de ADN y el reconocimiento facial son ejemplos de este factor.

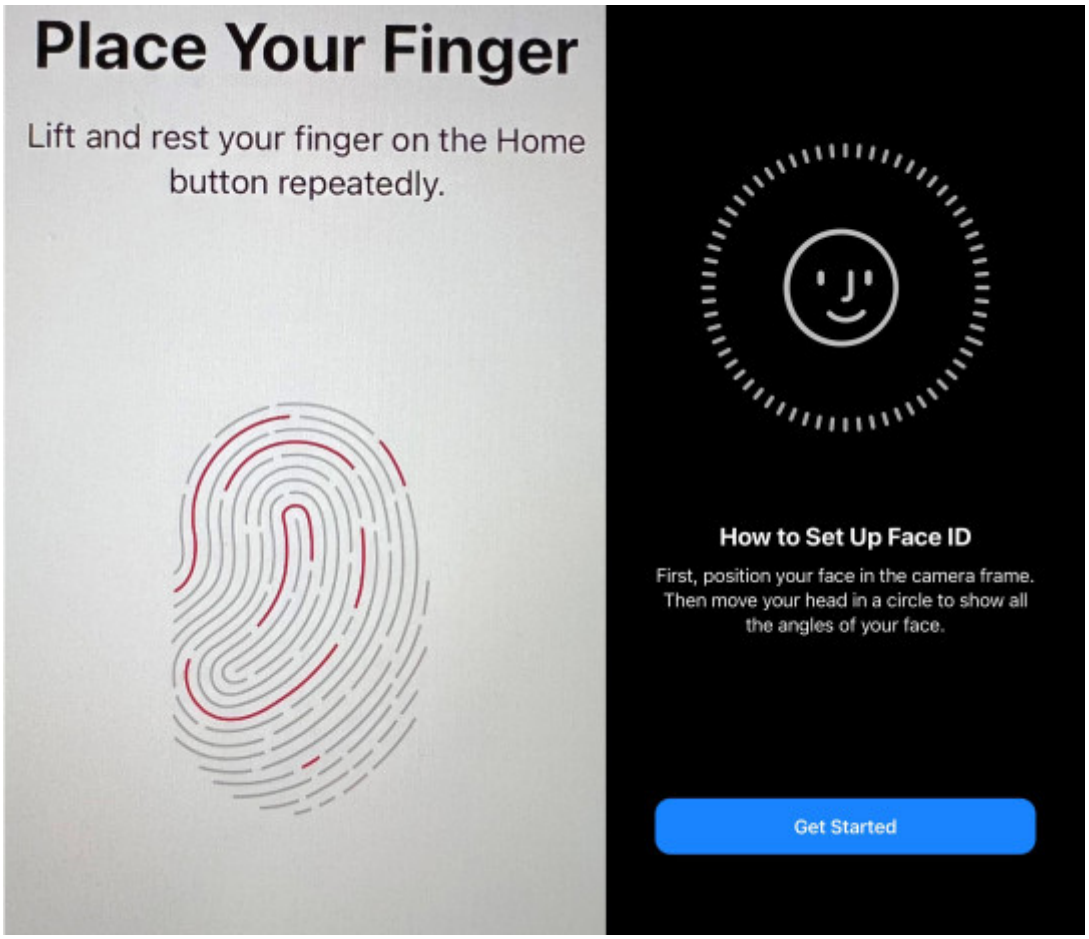


Fig. 3. Ejemplos de autenticación biométrica con un smartphone.

ADVANTAGE

Si un sistema pide sistemáticamente numerosos factores de autenticación, puede conseguir una seguridad sólida.

DISADVANTAGE

Sin embargo, una autentificación demasiado frecuente puede tener el efecto contrario, poniendo en riesgo la comodidad del usuario.

Interaktivní prvek

Otra forma de ver la autenticación es por los medios que se emplea y consecuentemente toma una de las tres formas:

- En la **autenticación básica** interviene un servidor. Por ejemplo, el servidor mantiene un archivo de usuario con contraseñas, nombres de usuario y algunos otros datos esenciales de autenticación. Este es el método más común de autenticación de usuarios. Sin embargo, tiene varios defectos, como el olvido y el extravío de la información de autenticación, como las contraseñas.

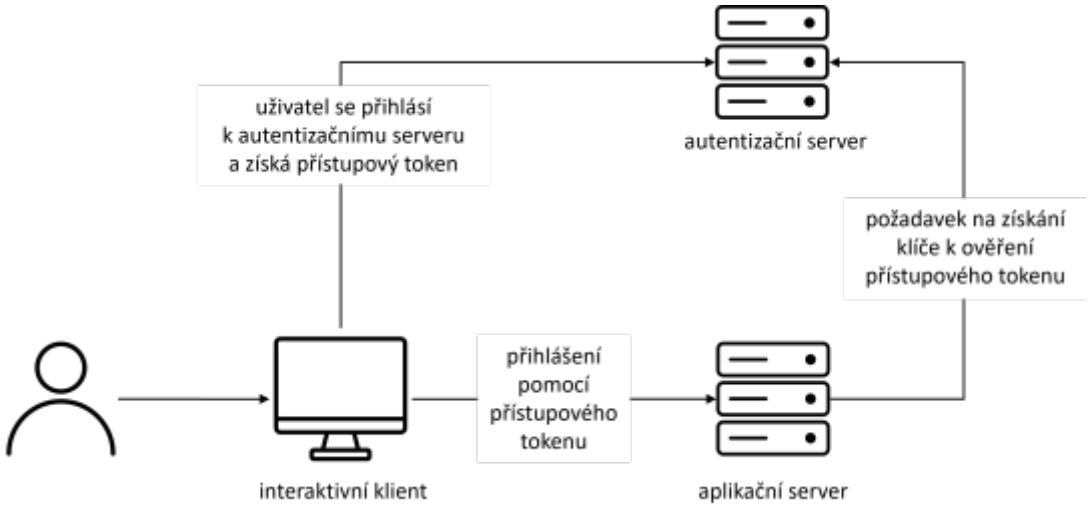


Fig. 4. Autenticación básica basada en un servidor.

- El **desafío-respuesta** es un método de autenticación en el que el servidor u otro sistema de autenticación propone un desafío al host que solicita la autenticación y espera una respuesta. Un ejemplo de ello es el uso de un *nonce*, un número arbitrario variable en el tiempo o una secuencia de bits que sólo se utiliza una vez para verificar que los datos sólo se utilizan una vez.
- La **autenticación centralizada** se refiere a un sistema en el que un servidor autentifica, autoriza y audita a los usuarios de la red. Estos tres procedimientos se llevan a cabo en respuesta a la

actividad del servidor. Un ejemplo de este tipo de autenticación es Kerberos.

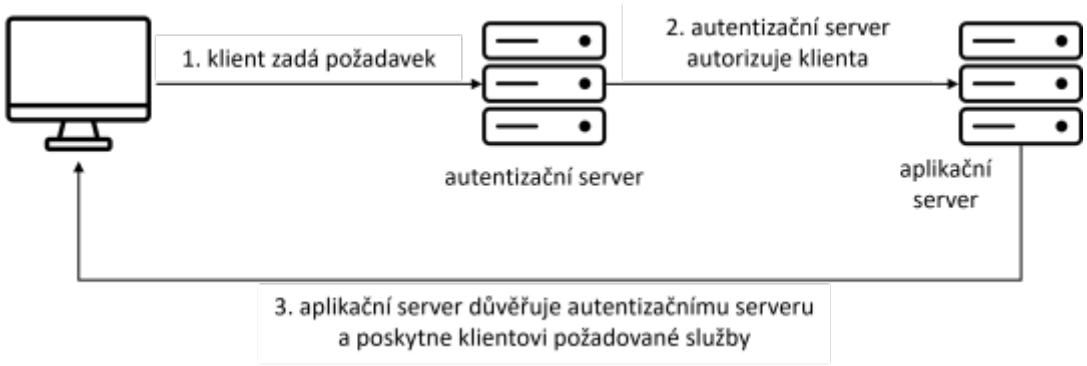


Fig. 5. Autenticación centralizada.

Interaktivní prvek

CHAPTER 2

Elementos y Proceso de Autenticación

La autenticación requiere un proceso de autenticación que se basa en:

- una entidad o grupo de entidades que buscan la autenticación;
- una característica distintiva de la entidad o entidades que se están autenticando;
- un autenticador (generalmente un servidor);
- un mecanismo de autenticación para verificar la veracidad de las características de autenticación;
- un mecanismo de control de acceso para aceptar o denegar la autenticación.

El primer elemento suele ser **personas, procesos o sistemas** que quieren acceder a un sistema. Si actúan individualmente, deben estar preparados para mostrarle al autenticador la prueba de que están autorizados para utilizar el recurso del sistema solicitado.

La **característica distintiva** del usuario es el segundo elemento de autenticación. Hablamos de esto antes, y se dividen en algo que sabes, algo que tienes y algo que eres. Algunos de estos elementos pueden no ser suficientes para autenticar la entidad por completo, y se puede utilizar una combinación de elementos de varios factores y confianza para mejorar la autenticación y brindar garantías más sólidas.

La función del **autenticador** es verificar positiva y automáticamente las credenciales de la entidad y determinar si esa entidad tiene permiso o no para acceder al recurso del sistema solicitado. Cuando se envía una solicitud de autenticación, el autenticador solicita las credenciales para completar el proceso de autenticación. Después de eso, el autenticador recopila los datos y los envía al mecanismo de autenticación. Un servidor designado por el usuario, una ***red privada virtual*** (**VPN**), un firewall, un servidor web, un servidor dedicado para toda la empresa, un servicio de autenticación independiente o algún otro tipo de servicio de identidad global puede funcionar como autenticador. Lo que sea que se use como autenticador debe llevar a cabo un procedimiento de autenticación que dé como resultado algún tipo de valor de resultado, como un *token* que se puede usar para obtener información sobre el usuario autorizado más adelante.

En la Figura 5 se muestra una descripción general de este proceso de autenticación.

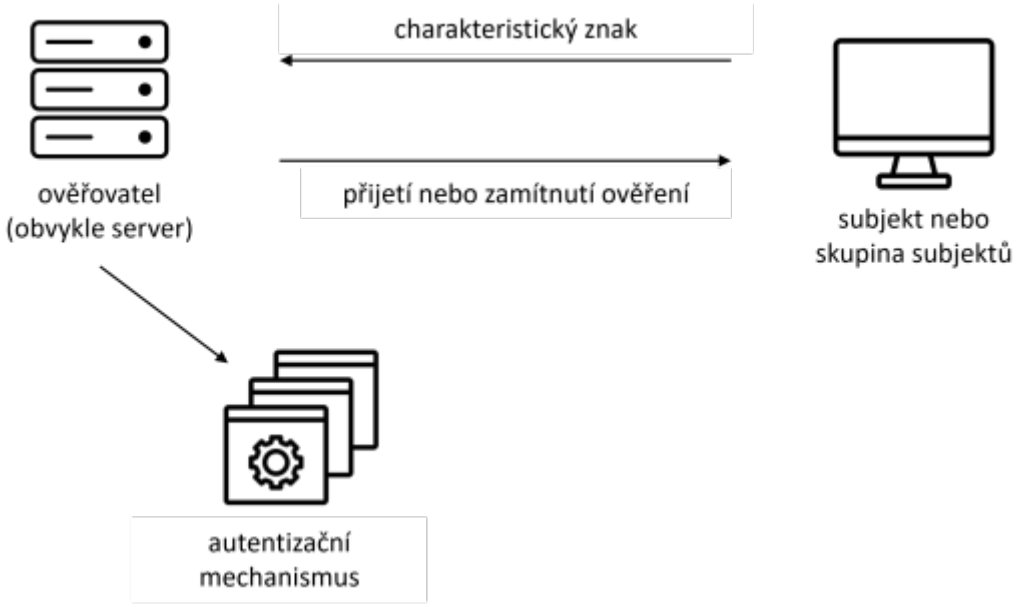


Fig. 6. El proceso básico de autenticación y sus elementos.

El **método de autenticación** se compone de tres piezas que funcionan juntas para garantizar que los rasgos de autenticación del usuario estén presentes:

- la entrada,
- el sistema de transporte y
- el verificador.

Interaktivní prvek

Un componente de entrada sirve como interacción del usuario con el mecanismo de autenticación. Algunos ejemplos son un teclado de ordenador, un lector de tarjetas, una cámara de video, un teléfono o un dispositivo similar. Los elementos de identificación del usuario capturados se llevan a un lugar donde serán examinados, analizados y aceptados o rechazados. Sin embargo, deben ser transportados para que estos productos lleguen a este lugar. Como resultado, la sección de transporte del sistema se encarga de pasar los datos entre el componente de entrada y el elemento que puede verificar la identidad de una persona. Esta información se transfiere a través de una red en sistemas de autenticación modernos, donde los protocolos pueden protegerla. El componente final del sistema de autenticación es la verificación, que es el mecanismo de control de acceso.

Interaktivní prvek

2.1 Tipos de Autenticación

Identificamos tres factores que se utilizan en la autenticación de un usuario. También señalamos que, si bien estos factores son buenos, hay algunos que sufren de vulnerabilidades. La Tabla 1 muestra las deficiencias de cada uno de los factores.

Table 1. Categorías de autenticación

Factor	Ejemplos	Vulnerabilidades
lo que sabes	contraseña, PIN	se puede olvidar, adivinar, duplicar, fácil de obtener en caso de fraude (por ejemplo, *phishing*)
lo que tienes	*tokens*, tarjeta inteligente, contraseña de un solo uso enviada a su número de teléfono	se puede perder, robar, duplicar
lo que eres	Huella dactilar, cara, iris	no repudiable - no se puede cambiar en caso de abuso

DISADVANTAGE

Mencionamos que los dos primeros factores, "lo que sabes" y "lo que tienes", pueden generar problemas para el autenticador porque la información proporcionada puede ser inexacta. Puede ser poco confiable porque tales factores están sujetos a una serie de problemas bien conocidos, incluida la posibilidad de que los artículos se pierdan, falsifiquen o reproduzcan fácilmente. El conocimiento también se puede olvidar, y el conocimiento y las cosas se pueden compartir o robar.

Interaktivní prvek

2.2 Autenticación Multi-factor

DEFINITION

La ***autenticación multi-factor*** (**MFA**) utiliza al menos dos factores diferentes (lo que sabes, lo que tienes y lo que eres). La ***autenticación de dos factores*** (**2FA**) es la misma, pero se utilizan exactamente dos factores.

Hoy en día, si se usa MFA, casi siempre es 2FA. Por lo general, el primer factor es una contraseña o un PIN (algo que sabes), y el segundo suele ser una tarjeta bancaria, un SMS o un código generado por una aplicación (lo que tienes, es decir, su dispositivo móvil). Usar huellas dactilares, escaneos de retina, etc. (lo que eres) es una opción, pero se usa con menos frecuencia porque se requiere hardware adicional (costo más alto).

La autenticación de múltiples factores es una buena manera de mitigar el riesgo y reducir las posibilidades de credenciales comprometidas. Por ejemplo, veamos una combinación de contraseña y código de aplicación. Incluso si el sitio en sí se ve comprometido o se obtiene una contraseña de otro lugar, el atacante no puede iniciar sesión porque, si bien puede proporcionar el nombre de usuario y la contraseña adecuados, no puede proporcionar el código generado en el dispositivo móvil. La contraseña robada se vuelve inútil (a menos que el atacante también robe el dispositivo móvil, pero ese no es un ataque escalable y, por lo tanto, no es una amenaza grave para la mayoría de las personas). Mientras tanto, los administradores del sistema aún pueden detectar intentos fallidos de iniciar sesión y pedirle a un usuario específico que cambie su contraseña o a todos sus usuarios si su sistema se vio comprometido y todas las contraseñas se filtraron.

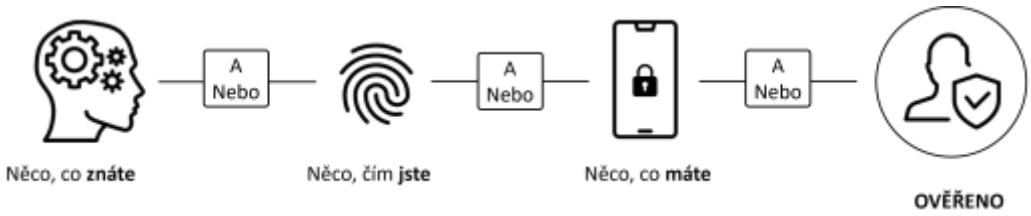


Fig. 7. Autenticación multi-factor.

CHAPTER 3

Autenticación basada en contraseña

La técnica de autenticación de contraseña es la más común y fácil de usar. Muchos sistemas suelen estar configurados así de forma predeterminada. Las contraseñas reutilizables, las ***contraseñas de un solo uso*** (**OTP**), las contraseñas de desafío y respuesta y las contraseñas de enfoque combinado son ejemplos de autenticación de contraseña.

Contraseñas reutilizables

Hay dos formas de autenticación en la autenticación de contraseña reutilizable: autenticación de usuario y autenticación de cliente.

- **Autenticación de usuario.** Es el tipo de autenticación más frecuente y es probable que la mayoría de los usuarios estén familiarizados con él. Siempre lo inicia el usuario, que envía una solicitud al servidor de autenticación y autorización para acceder a un recurso del sistema en particular. Cuando el servidor recibe la solicitud, solicita al usuario un nombre de usuario y una contraseña. El servidor los compara con copias en su base de datos cuando se envían. La autorización se proporciona en función del partido.
- **Autenticación de cliente.** Por lo general, un usuario busca la autenticación y, posteriormente, la autorización para acceder a un sistema o un conjunto de recursos del sistema desde el servidor. La autenticación de los usuarios no les otorga acceso a ningún recurso del sistema que deseen. La autorización del usuario para utilizar los recursos deseados en la cantidad solicitada y no más debe establecerse mediante autenticación. La autenticación del cliente es el nombre de este tipo de autenticación. Establece las identidades de los usuarios y les permite acceder a los recursos del sistema de forma controlada.

Debido a que estos métodos de autenticación son los más utilizados, también son los más explotados.

DISADVANTAGE

Además, tampoco son confiables porque las personas las olvidan, las escriben, las comparten y, lo que es más importante, son fáciles de adivinar, ya que las personas eligen contraseñas simples. Y también son vulnerables a la vigilancia y el cracking. Además, las contraseñas débiles (por ejemplo, cortas o de estructura simple) son vulnerables a las computadoras súper fuertes de hoy en día, que pueden descifrarlas por fuerza bruta mediante una búsqueda exhaustiva.

Contraseñas de un solo uso

La autenticación de sesión es otro nombre para la autenticación de contraseña de un solo uso. A diferencia de las contraseñas reutilizables, que se pueden usar repetidamente, las contraseñas de un

solo uso solo se usan una vez y luego se descartan.

ADVANTAGE

Usando fuertes generadores de números aleatorios, se generan al azar. Esto disminuye la probabilidad de que sean adivinados. En muchas circunstancias, se cifran antes de enviarse en claro para limitar las posibilidades de que sean interceptados.

Las contraseñas de un solo uso vienen en una variedad de formas. Los ejemplos incluyen S/Key y contraseñas de *token*. S/Key es un sistema de creación de contraseñas de un solo uso definido en RFC 1760.

Otro ejemplo es el llamado número TAN utilizado en Alemania en el pasado (Figura 8). Las contraseñas de un solo uso, aunque suelen ser más seguras, tienen una serie de inconvenientes, incluidos los problemas de sincronización causados por el tiempo transcurrido entre la marca de tiempo de la contraseña y el reloj del sistema. La contraseña no se puede utilizar una vez que estos dos tiempos están desfasados.

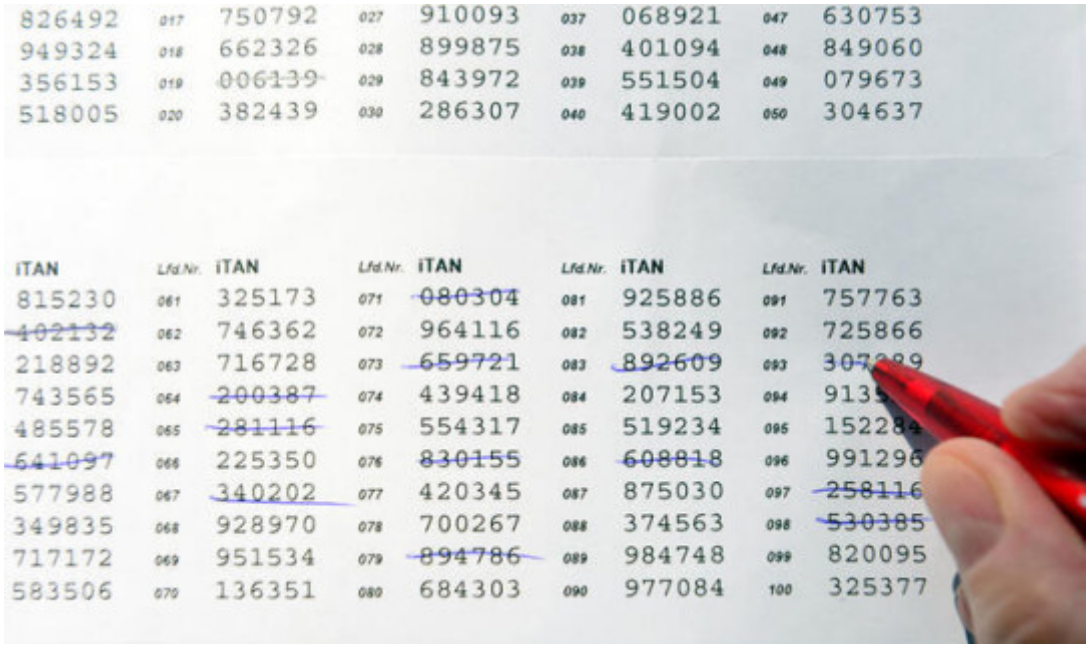


Fig. 8. Los números TAN son un ejemplo de contraseñas de un solo uso (OTP).

Algunos tipos de contraseñas de un solo uso (p. ej., códigos SMS y códigos generados por aplicaciones) generalmente se usan como un segundo factor cuando se emplea 2FA.

Contraseñas de desafío-respuesta

DEFINITION

Como método de autenticación basado en contraseña, el desafío-respuesta es un proceso de autenticación de protocolo de enlace en el que el autenticador desafía al usuario que busca la autenticación. Para ser autenticado, el usuario debe ofrecer la respuesta correcta.

Dependiendo del sistema, el desafío puede tomar muchas formas diferentes. Puede ser una solicitud básica de una contraseña, un número, un resumen o un *nonce*. El individuo que quiere ser autenticado debe responder al desafío del sistema. Hoy en día, las respuestas se envían a través de una función unidireccional y un token de contraseña, que se conocen como *tokens* asíncronos. Cuando el servidor recibe la respuesta del usuario, vuelve a verificar la contraseña.

La aplicación más común de autenticación de desafío-respuesta se encuentra en sistemas distribuidos. A pesar de su popularidad, la autenticación de desafío-respuesta se enfrenta a dificultades debido a fallas como la participación del usuario y los ataques de prueba y error. El problema con la participación del usuario es la capacidad del usuario para ubicar el desafío en pantallas típicamente ruidosas. A continuación, el usuario debe escribir una respuesta rápidamente.

Según el nivel de seguridad requerido, se le puede pedir al usuario que recuerde la respuesta larga o se le puede obligar a escribirla, después de lo cual el usuario debe transcribirla e ingresarla. Esto tiene el potencial de ser propenso a errores.

INTERESTING

Algunos fabricantes han intentado aliviar la carga del usuario de recordar y teclear largas cadenas automatizando la mayor parte del procedimiento, ya sea cortando y pegando el desafío y las respuestas o mediante un proceso automatizado de bajo nivel que limita la respuesta del usuario a respuestas sí/no.

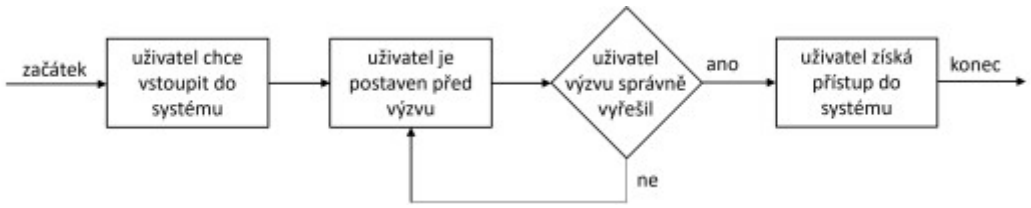


Fig. 9. El proceso de autentificación en un sistema desafío-respuesta.

También vale la pena señalar que las respuestas de desafío que requieren contraseñas podrían ser mal utilizadas en su forma más básica porque las contraseñas son relativamente fáciles de obtener. Las contraseñas pueden ser potencialmente interceptadas si se envían en texto sin formato. Sin embargo, dado que una contraseña no se comunica en texto sin formato a través de las redes, esto no representa una consideración de seguridad.

Interaktivní prvek

3.1 Problemas de Seguridad de las Contraseñas

En los ciberataques, la violación de datos es uno de los tipos de ataques más frecuentes y de los objetivos de un atacante. Por ello, las contraseñas como mecanismo de autenticación son cada vez más problemáticas.

La singularidad de una contraseña es una de sus características más importantes. Sin embargo, muchas contraseñas son todo menos esto. En la Tabla 2 se enumeran las contraseñas y frases más populares utilizadas por la gente en todo el mundo.

Table 2. Las 10 contraseñas más comunes, fuente: cybernews.com.

Contraseñas
123456
123456789
qwerty
password
12345
qwerty123
1q2w3e
12345678
111111
1234567890

DISADVANTAGE

Además, existen otros problemas relacionados con las contraseñas. Muchas personas optan por vincularlas a algo que puedan recordar fácilmente para generar combinaciones sencillas y memorables. Sin embargo, esto no hace que la contraseña sea única, sino todo lo contrario.

Cybernews examinó aproximadamente 15.000 millones de entradas y las clasificó en varias categorías y frases. Sus resultados muestran que ciertas características de las contraseñas son problemáticas: datos relacionados con el usuario. Además, investigaron la longitud de las contraseñas en función del número de caracteres que incluían. Por desgracia, la mayoría de las contraseñas utilizadas tenían 8 caracteres o menos.

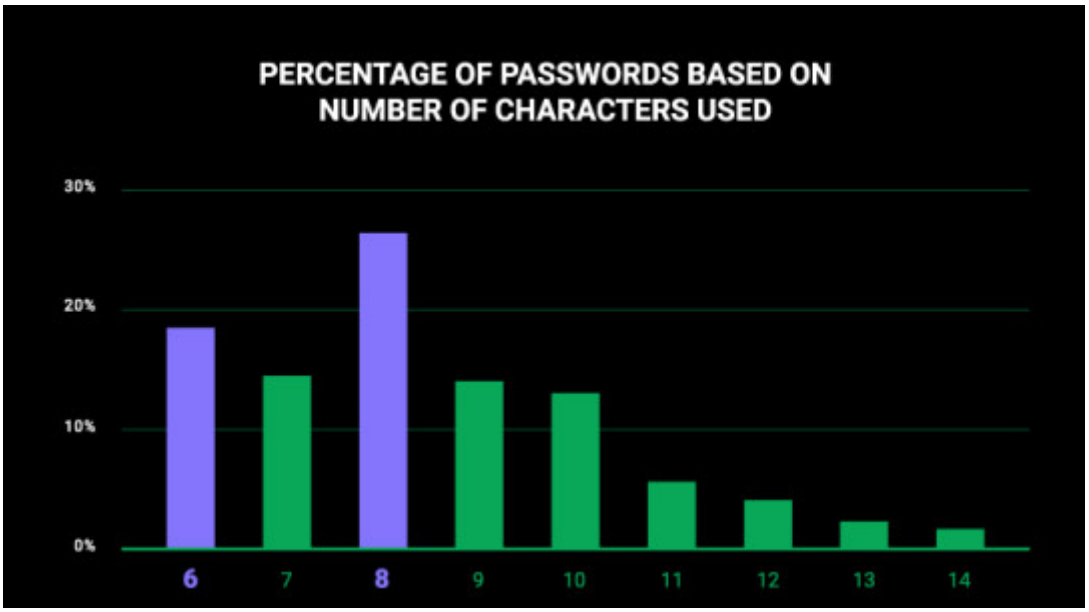


Fig. 10. Estadísticas sobre la longitud de las contraseñas, fuente: cybernews.com.

Hay métodos más eficaces para crear una contraseña fuerte. Utilizando "heat" como elemento de la contraseña, por ejemplo, una contraseña sencilla puede ser "letsgoheat" (10 caracteres), mientras que una contraseña más difícil podría ser "heatromearsenalhjamesp" (una frase de contraseña de 22 caracteres). La gente también genera contraseñas seguras con dispositivos mnemotécnicos, que son preferibles porque suelen ser largas y contienen palabras aleatorias sin relación lógica entre ellas, lo que las hace más fáciles de recordar para una persona, pero más difíciles de descifrar para un algoritmo.

3.2 Ataques a Contraseñas

Las contraseñas pueden ser atacadas de varias maneras, y en general, los ataques se pueden clasificar de la siguiente manera:

- Los ataques no electrónicos no requieren conocimientos técnicos para descifrar la contraseña. Ejemplos de estos ataques son el mirar por encima del hombro, la ingeniería social y el rebuscar en la papelera.
- Los ataques de tipo electrónico requieren conocimientos técnicos. Ejemplos de estos ataques son los ataques de diccionario y de fuerza bruta, y los ataques de tabla arco iris.

3.2.1 Ataques No Electrónicos

DEFINITION

La ingeniería social es un tipo de ataque en el que el atacante trata de aprovechar la tendencia natural de la gente a confiar en cualquiera. Aprovechando esa confianza, el atacante obtiene rápidamente las credenciales de la víctima y las utiliza para acceder posteriormente a su cuenta.

Phishing, *Pharming*, y *Whaling* son solo algunos ejemplos. Tenga en cuenta que algunos de ellos requieren ciertos conocimientos técnicos (por ejemplo, el *phishing*).

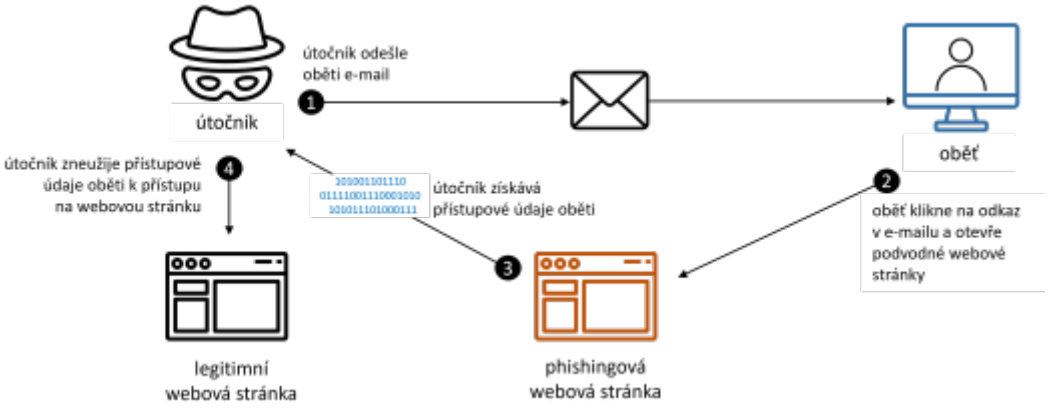


Fig. 11. Proceso de un ataque de phishing.

En un ataque de mirar por encima del hombro (*shoulder surfing*), el atacante se sitúa detrás del usuario, observando cómo teclea sus credenciales, que posteriormente utilizará para acceder a su cuenta.

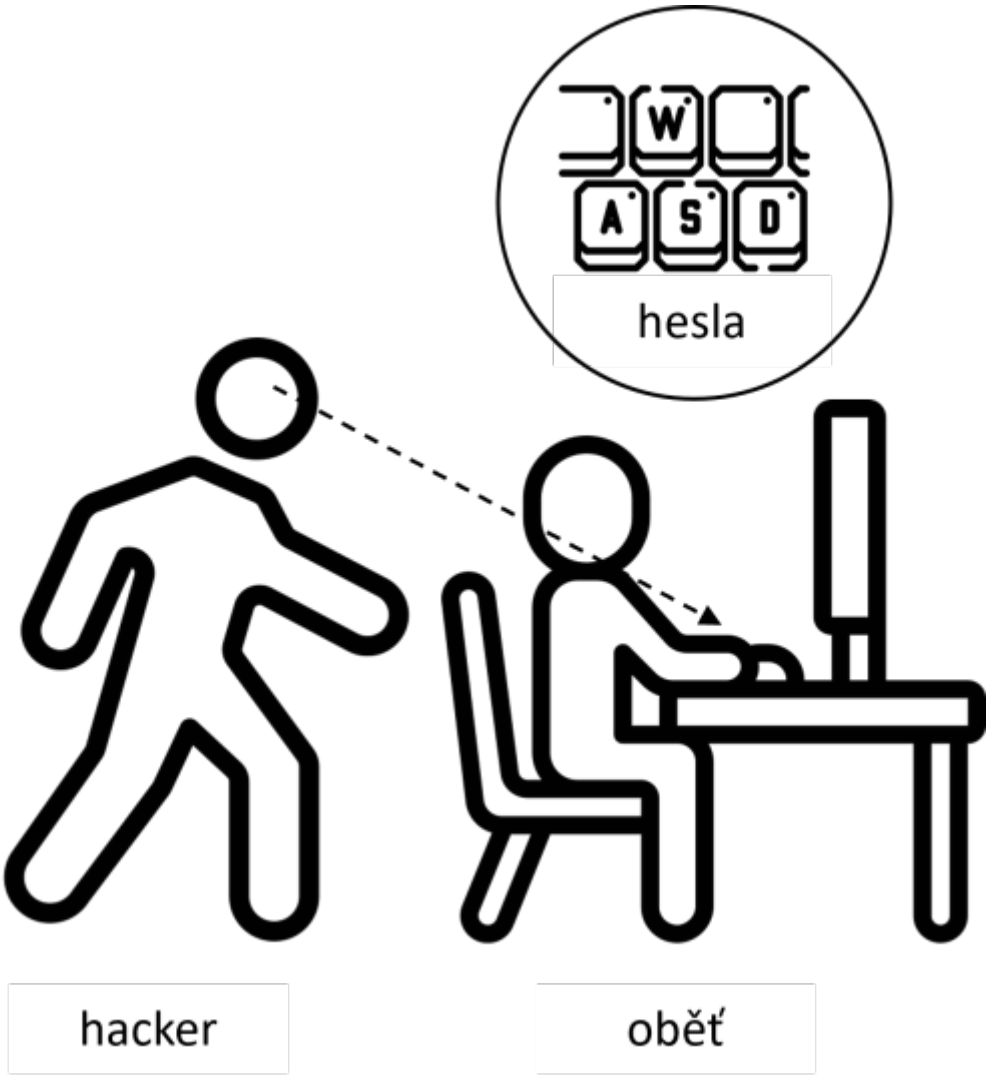


Fig. 12. Ejemplo de *shoulder surfing*.

El atacante que rebusca en la papelera (*Dumpster Diving*) descubre algo valioso en su basura, como su contraseña o el pin de su tarjeta de crédito.

3.2.2 Ataques Electrónicos

DEFINITION

Un **ataque de diccionario** es un ataque en el que un atacante intenta entrar en un sistema protegido por contraseña utilizando cada palabra de un diccionario como un tipo de contraseña para ese sistema.

Consiste en probar todas las cadenas de una lista que ha sido preestablecida. Históricamente, en este tipo de ataques se utilizaban palabras de diccionario (de ahí la expresión "ataque de diccionario").

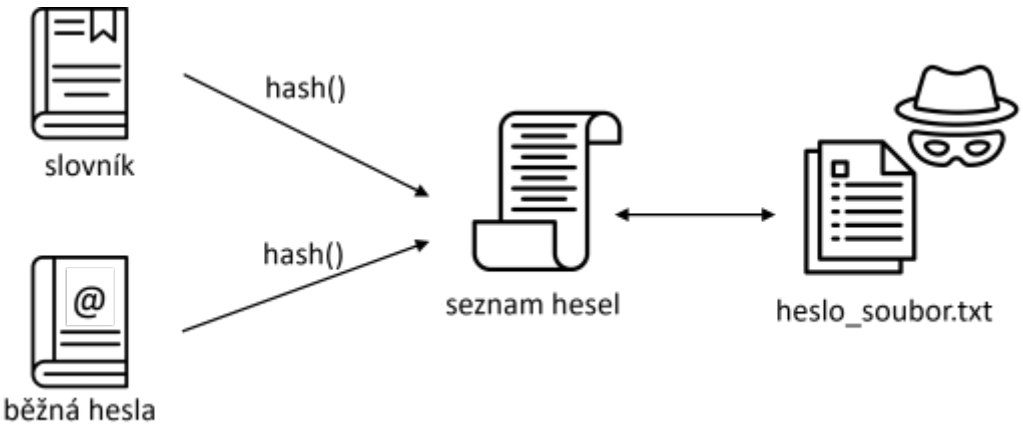


Fig. 13. Cómo funciona un ataque de diccionario.

DEFINITION

Un ataque de diccionario sólo intenta las opciones que se cree que tienen más probabilidades de éxito.

Muchas personas tienen la tendencia a elegir contraseñas cortas que son palabras ordinarias o contraseñas comunes, o variantes obtenidas, por ejemplo, añadiendo un dígito o un carácter de puntuación. Los ataques de diccionario suelen tener éxito porque muchas personas tienen la tendencia a elegir contraseñas cortas que son palabras ordinarias o contraseñas comunes o variantes obtenidas, por ejemplo, añadiendo un dígito o un carácter de puntuación.

Dado que las listas disponibles cubren la mayoría de las estrategias típicas de formación de contraseñas, los ataques de diccionario son difíciles de superar con la generación de patrones utilizando software de crackeo. Una forma más segura es utilizar una herramienta de gestión de contraseñas o un método manual para construir una contraseña grande (15 letras o más) o una contraseña de varias palabras al azar.

Ataques de Fuerza Bruta

DEFINITION

Descrito de forma sencilla, la fuerza bruta es un método de descifrado de contraseñas en el que el atacante prueba tantas combinaciones potenciales de contraseñas como sea posible utilizando un conjunto de parámetros.

Un sitio web puede, por ejemplo, establecer un parámetro que exija que la contraseña tenga entre 8 y 16 caracteres. El descifrador de contraseñas podría empezar con 00000000 en la variante más básica. Luego podría probar con 00000001, 00000010, 00000100, y así sucesivamente hasta que haya agotado todas las combinaciones de caracteres concebibles.

Para una contraseña de longitud 8 – Cada campo puedes ser:

- una letra del alfabeto en minúsculas (26 posibilidades)

- una letra del alfabeto en mayúsculas (26 posibilidades)
- un número (10 posibilidades del 0 al 9)
- signos de puntuación u otros caracteres especiales (33 posibilidades)

Teniendo en cuenta todo esto, se puede calcular el número final de contraseñas posibles para una contraseña de 8 caracteres: 3.025.989.069.143.040, es decir, unos 3 cuatrillones, y cada uno es un intento distinto.

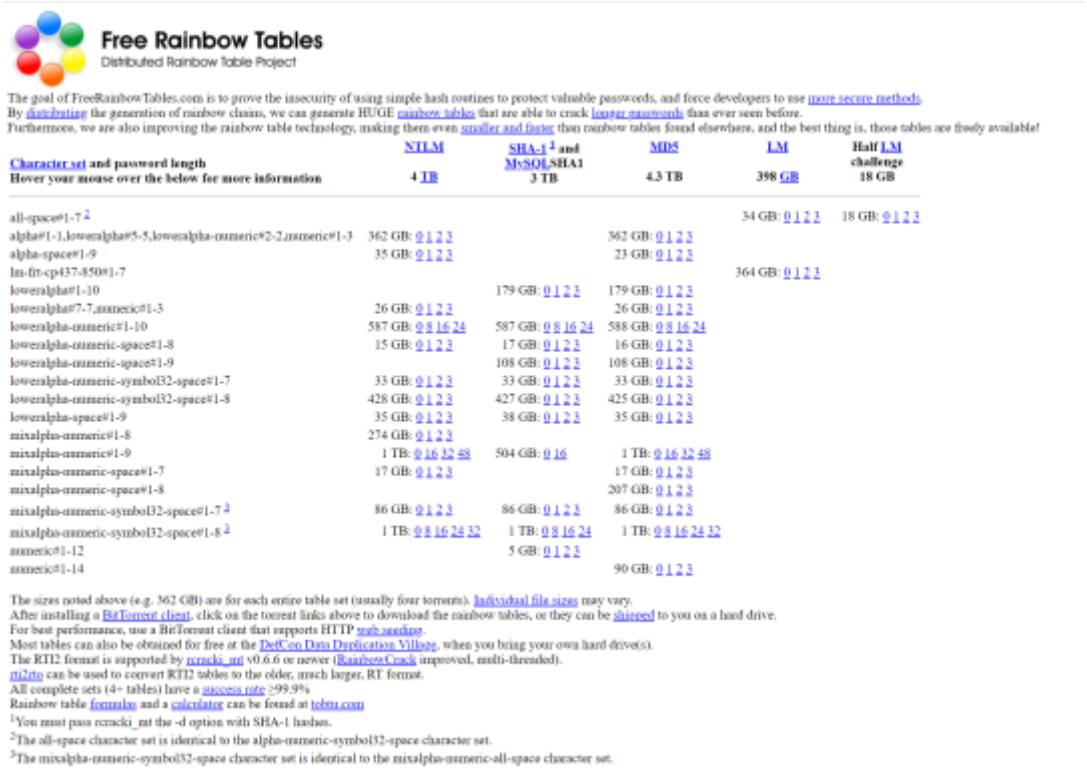
Se podría pensar que alguien construye un programa que va a un sitio web, teclea su nombre de usuario y contraseña, pulsa el botón de inicio de sesión e intenta adivinar su contraseña. Luego repite el proceso tres cuatrillones de veces más. Sin embargo, no es así. Si un sitio web tarda 2 segundos en cargar una página, son 2 segundos de espera por cada intento de obtener una página de "contraseña incorrecta". En otras palabras, si el sitio web no bloquea el inicio de sesión después de una cantidad específica de intentos sospechosos, podría tardar hasta 9 cuatrillones de segundos, o 287,9 millones de años. En realidad, un ataque de este tipo se lleva a cabo utilizando nombres de usuario y contraseñas filtrados. Éstos se filtran como resultado de una violación de datos (que ocurre más a menudo de lo que se cree). Por lo tanto, una contraseña puede ser revelada de dos maneras:

- Su contraseña no está cifrada y se almacena en texto plano en un entorno extremadamente inseguro. El lector sólo tendría que copiar y pegar su contraseña. Si su contraseña es password1, por ejemplo, cualquiera que lea el contenido de la brecha de datos vería password1. En este caso, la fuerza bruta es innecesaria porque el sitio web ya ha entregado su información en bandeja de plata.
- Su contraseña se convierte en hash en lugar de almacenarse en texto plano en un entorno más seguro. Por ejemplo, si el sitio web ha cifrado su contraseña utilizando la función hash SHA-256, la contraseña1 aparecería como 0b14d501a594442a01c6859541bcb3e8164d183d32937b851835442f69d5c94e.

Fig. 14. Ataque de fuerza bruta a las contraseñas.

Las tablas arcoíris (o tablas *rainbow*) son un tipo especial de ataque de fuerza bruta a las contraseñas. Su objetivo es descifrar contraseñas almacenadas con hash. Debido a que las tablas *rainbow* son una lista precalculada de hashes de términos de diccionario o contraseñas previamente violadas y almacenadas en una base de datos usando el hash como clave, existe un compromiso de tiempo y memoria. Generar una tabla *rainbow* puede llevar mucho tiempo, pero sólo hay que hacerlo una vez. Se pueden buscar los

hashes de las contraseñas y obtener la contraseña adecuada rápidamente cuando se hace. Para poner en perspectiva la enormidad de estas bases de datos, algunas tablas *rainbow* pueden tener un tamaño de 7 a 9 TB.



Free Rainbow Tables
Distributed Rainbow Table Project

The goal of FreeRainbowTables.com is to prove the insecurity of using simple hash routines to protect valuable passwords, and force developers to use more secure methods.
By distributing the generation of rainbow chains, we can generate HUGE rainbow tables that are able to crack longer passwords than ever seen before.
Furthermore, we are also improving the rainbow table technology, making them even smaller and faster than rainbow tables found elsewhere, and the best thing is, those tables are freely available!

Character set and password length Hover your mouse over the below for more information	NTLM 4 TB	SHA-1 [1] and MySQL SHA1 3 TB	MD5 4.3 TB	LM 398 GB	Half LM challenge 18 GB
all-space#1-7 [2]				34 GB: 0 1 2 3	18 GB: 0 1 2 3
alpha#1-1,loweralpha#5-5,loweralpha-numeric#2-2,numeric#1-3	362 GB: 0 1 2 3		362 GB: 0 1 2 3		
alpha-space#1-9	35 GB: 0 1 2 3		23 GB: 0 1 2 3		
lm-frt-cp437-850#1-7				364 GB: 0 1 2 3	
loweralpha#1-10		179 GB: 0 1 2 3	179 GB: 0 1 2 3		
loweralpha#7-7,numeric#1-3	26 GB: 0 1 2 3		26 GB: 0 1 2 3		
loweralpha-numeric#1-10	587 GB: 0 8 16 24	587 GB: 0 8 16 24	588 GB: 0 8 16 24		
loweralpha-numeric-space#1-8	15 GB: 0 1 2 3	17 GB: 0 1 2 3	16 GB: 0 1 2 3		
loweralpha-numeric-space#1-9		108 GB: 0 1 2 3	108 GB: 0 1 2 3		
loweralpha-numeric-symbol32-space#1-7	33 GB: 0 1 2 3	33 GB: 0 1 2 3	33 GB: 0 1 2 3		
loweralpha-numeric-symbol32-space#1-8	428 GB: 0 1 2 3	427 GB: 0 1 2 3	425 GB: 0 1 2 3		
loweralpha-space#1-9	35 GB: 0 1 2 3	38 GB: 0 1 2 3	35 GB: 0 1 2 3		
mixalpha-numeric#1-8	274 GB: 0 1 2 3				
mixalpha-numeric#1-9	1 TB: 0 16 32 48	504 GB: 0 16	1 TB: 0 16 32 48		
mixalpha-numeric-space#1-7	17 GB: 0 1 2 3		17 GB: 0 1 2 3		
mixalpha-numeric-space#1-8			207 GB: 0 1 2 3		
mixalpha-numeric-symbol32-space#1-7 [3]	86 GB: 0 1 2 3	86 GB: 0 1 2 3	86 GB: 0 1 2 3		
mixalpha-numeric-symbol32-space#1-8 [3]	1 TB: 0 8 16 24 32	1 TB: 0 8 16 24	1 TB: 0 8 16 24 32		
numeric#1-12		5 GB: 0 1 2 3			
numeric#1-14			90 GB: 0 1 2 3		

The sizes noted above (e.g. 362 GB) are for each entire table set (usually four torrents). Individual file sizes may vary.
After installing a BitTorrent client, click on the torrent links above to download the rainbow tables, or they can be shipped to you on a hard drive.
For best performance, use a BitTorrent client that supports HTTP web seeding.
Most tables can also be obtained for free at the DefCon Data Duplication Village, when you bring your own hard drive(s).
The RTI2 format is supported by rcracki_mt v0.6.6 or newer (RainbowCrack improved, multi-threaded).
rti2rto can be used to convert RTI2 tables to the older, much larger, RT format.
All complete sets (4+ tables) have a success rate ≥99.9%
Rainbow table formulas and a calculator can be found at tobtu.com
[1]You must pass rcracki_mt the -d option with SHA-1 hashes.
[2]The all-space character set is identical to the alpha-numeric-symbol32-space character set.
[3]The mixalpha-numeric-symbol32-space character set is identical to the mixalpha-numeric-all-space character set.

Fig. 15. Tamaños de las tablas *rainbow*, fuente: freerainbowtables.com.

Interaktivní prvek

Interaktivní prvek

3.2.3 Herramientas de Ataque de Contraseñas

Las herramientas para hackear contraseñas se están volviendo cada vez más populares en estos días, así que repasaremos algunas de ellas. En su mayor parte, las herramientas de descifrado de contraseñas se utilizan para probar la seguridad de una contraseña o lanzar un ataque hostil. Existen numerosas herramientas en línea y fuera de línea dedicadas únicamente a descifrar contraseñas. Las interfaces de inicio de sesión remotas, como los servicios SSH y RDP, son objeto de ataques en línea. Por otro lado, los ataques fuera de línea ocurren después de que los archivos hayan sido exfiltrados. Después de eso, las contraseñas son atacadas inmediatamente.

Algunas de las herramientas disponibles son Hashcat, John the Ripper o THC Hydra.

Hashcat es un programa de recuperación de contraseñas multiplataforma que funciona tanto con GPU como con CPU. Hashcat fue creado en 2009 por el MIT y es reconocido por admitir una amplia gama de

algoritmos hash como LM Hash, NT Hash, MD4, MDS y muchos otros. Cuando se desarrolló por primera vez, este programa admitía cuatro tipos diferentes de ataques.

- Ataques de diccionario: más de 14 millones de contraseñas, comenzando por las más populares y terminando por las menos comunes. Adivinará una contraseña, la cifrará y comparará el hash con la contraseña que está intentando descifrar.
- Ataques combinadores: similares a los ataques de diccionario, pero en lugar de usar listas de dos palabras como "diccionarios", crea una nueva lista de palabras con cada palabra unida a las demás.
- Ataques de máscaras: por ejemplo, si se sabe que la contraseña de una cuenta tiene 9 caracteres y termina con un dígito, sabe que necesitará una combinación de $52 \cdot 10^9$ para adivinar la contraseña, lo que llevará aproximadamente 4 años. Sin embargo, si se sabe que la contraseña comienza con una letra mayúscula y termina con un número, el tiempo se reducirá a la mitad.
- Ataque basado en reglas: Hashcat puede especificar qué tipo de contraseña probar en función de cómo su víctima crea una contraseña.
- Ataque de fuerza bruta: lo intentará todo hasta encontrar algo "que normalmente llevará mucho tiempo porque probará todas las combinaciones posibles".

La Licencia Pública General de GNU lanzó **John the Ripper** en 1996. Se trata de una herramienta de seguridad, auditoría y recuperación de contraseñas de código abierto que soporta cientos de tipos de hash y cifrado. Está disponible en una variedad de plataformas, permitiéndole usar el mismo cracker en cualquiera de ellas. Como se dijo anteriormente, esta herramienta soporta una variedad de tipos de hash.

Cuando se ejecuta en diferentes plataformas, los tipos de hash pueden diferir. Esta herramienta tiene muchos modos de cracking, incluyendo:

- Modo de lista de palabras: En este modo, se especificará un "archivo de texto" de la lista de palabras que idealmente debería ser ordenado, y la contraseña será comparada con la contraseña que está tratando de romper.
- Crack simple: Este es el primer modo con el que se empieza a crackear. De hecho, la contraseña exitosa se compara con todas las contraseñas cargadas para verificar si algún usuario tiene la misma contraseña, lo que acelera el proceso.
- Modo incremental: es el modo de cracking más potente, que probará todas las combinaciones imaginables, pero no se detendrá debido al gran número de combinaciones posibles.
- Modo externo: se trata de funciones escritas en C que son construidas por la herramienta al inicio y que contienen un programa que empleará para generar una contraseña.

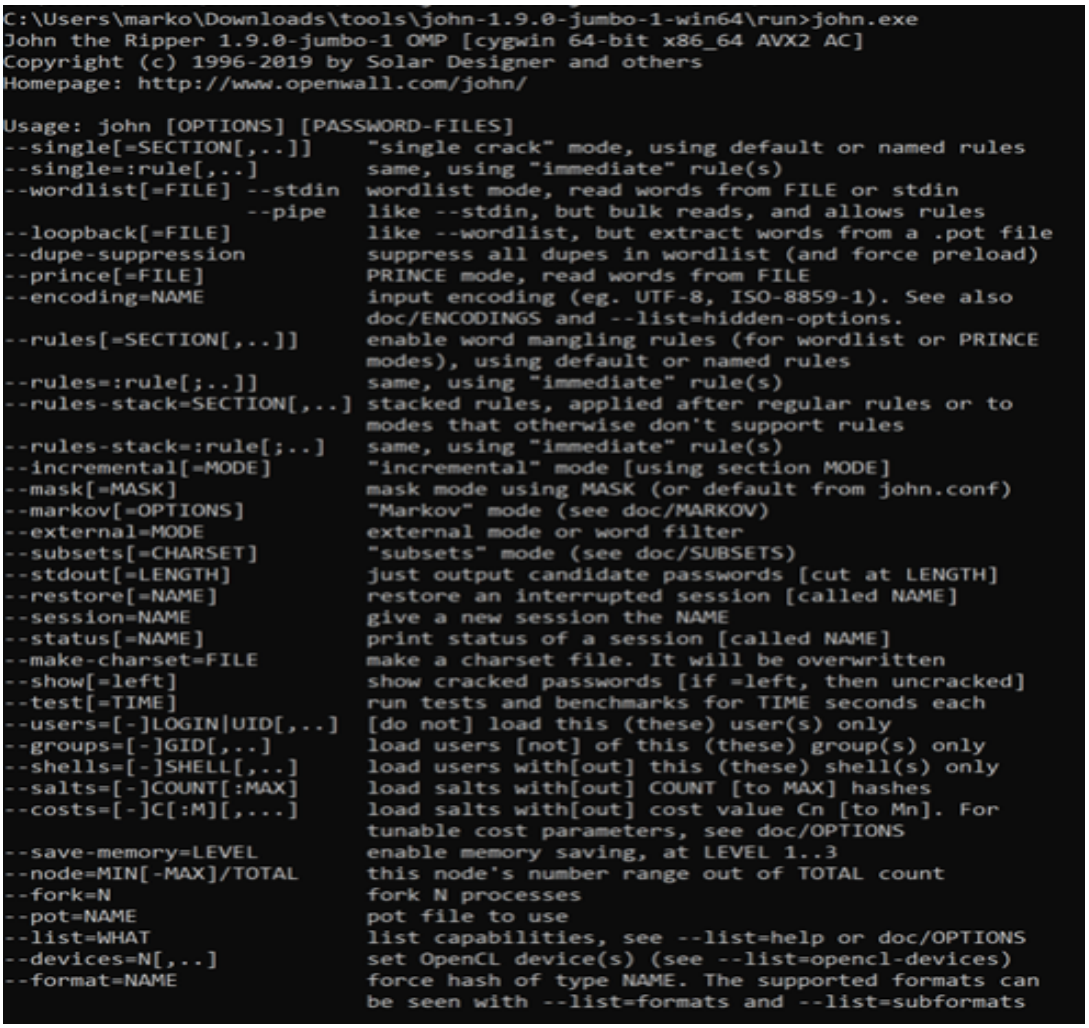


Fig. 16. John the Ripper en acción.

THC hydra, diseñado por Van Hauser en 2001, es un programa de cracking online que demuestra lo sencillo que es obtener acceso no autorizado a una máquina remota. Esta utilidad es compatible con una variedad de protocolos, incluyendo "FTP, HTTP, HTTPS, MySQL, Postgress..." y una variedad de plataformas, incluyendo UNIX, MacOS, Windows y dispositivos móviles. Esta herramienta puede ejecutar un ataque de diccionario paralelo, un ataque de fuerza bruta o híbrido, un ataque paralelo a muchos servidores, y más. THC Hydra es reconocida por ser rápida y efectiva, pero esto depende del protocolo.

La principal diferencia entre esta herramienta y John the Ripper es que ésta es una herramienta de descifrado de contraseñas online, mientras que John the Ripper es una herramienta offline.

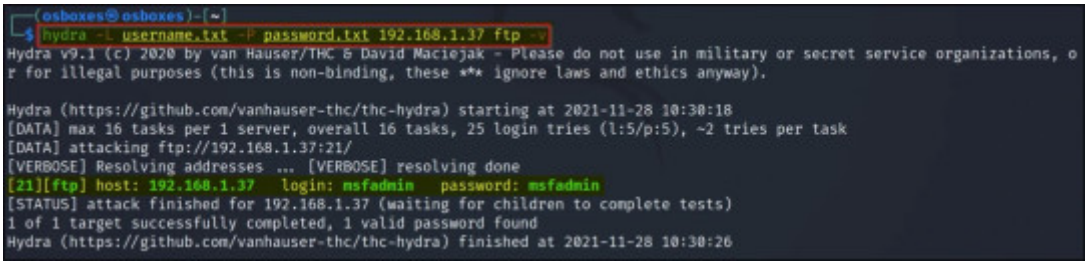


Figura 17. THC Hydra en acción.

Interaktivní prvek

CHAPTER 4

Diferentes Aspectos de la Seguridad de las Contraseñas

En este capítulo, trataremos diferentes aspectos de la seguridad de las contraseñas, que pueden dividirse en:

- Centrada en el usuario
- Centrada en el servidor.

Estos aspectos incluyen las directrices para las contraseñas de seguridad, la autenticación de 2 factores, el almacenamiento adecuado de las contraseñas en el lado del servidor, etc.

4.1 Directrices y Buenas Prácticas en Materia de Contraseñas Seguras

Las contraseñas son el método de autenticación predominante porque son las más fáciles de implementar para los desarrolladores y las más fáciles de entender y usar para los usuarios. Sin embargo, algunas debilidades conceptuales están asociadas con el uso de contraseñas (por ejemplo, mal seleccionadas, fáciles de adivinar, etc.). El ***Instituto Nacional de Estándares y Tecnología*** (**NIST**) de EE. UU. actualiza regularmente sus recomendaciones para crear y administrar contraseñas. En uno de los cambios recientes en el paradigma de la seguridad de las contraseñas, han sugerido que los usuarios se centren en la longitud de la contraseña por encima de la complejidad (combinaciones de caracteres especiales, números, letras minúsculas o mayúsculas) porque las contraseñas complejas son difíciles de recordar. En consecuencia, los usuarios tienden a lograr la complejidad de formas predecibles (por ejemplo, agregando un número 1 al final de una contraseña). Una forma de lograr la longitud es con frases de contraseña sin sentido donde las palabras están en una secuencia que no tiene significado. Por la misma razón, NIST ya no recomienda reglas estrictas de composición de caracteres al crear una contraseña. Sin embargo, recomienda comparar regularmente las contraseñas (o al menos cualquier contraseña nueva) con una lista de contraseñas comprometidas para identificar contraseñas ya reveladas y débiles. No obstante, la longitud mínima recomendada de una contraseña es de 12 caracteres. Si bien en el pasado se recomendaban cambios regulares de contraseñas, éste ya no es el caso, ya que esto hace que sea menos probable que los usuarios recuerden sus contraseñas después de los cambios y, en cambio, comiencen a usar las mismas contraseñas con pequeñas modificaciones.

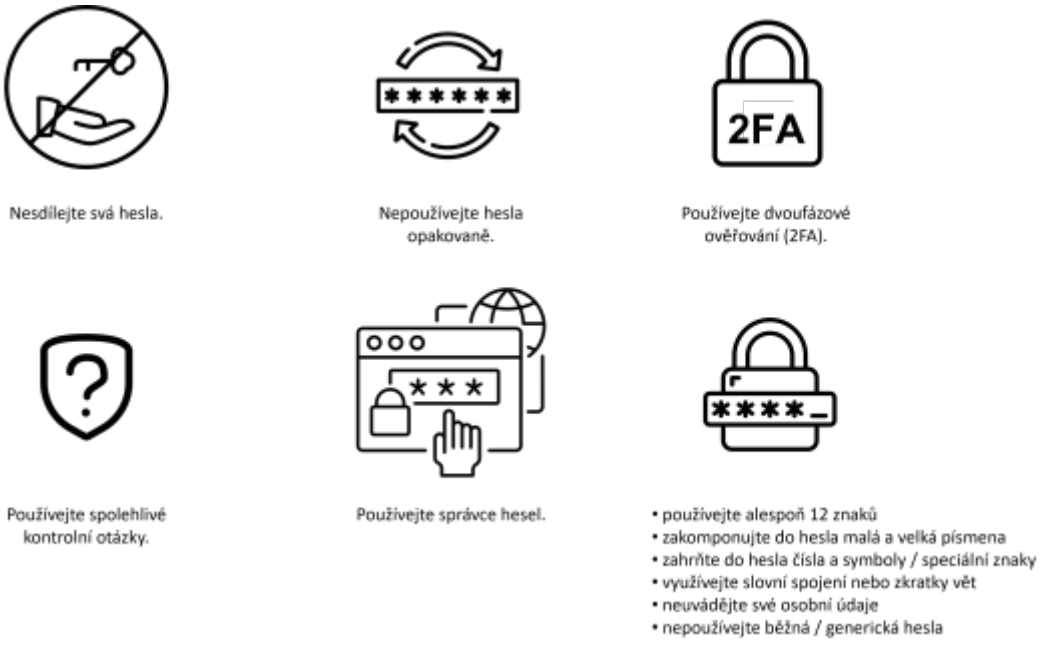


Fig. 18. Mejores prácticas para el uso de contraseñas.

Para una contraseña segura y de creación propia, se deben tener en cuenta los siguientes requisitos:

- utilizar al menos 12 caracteres
- incluya letras minúsculas y mayúsculas

- incluya números y símbolos / caracteres especiales
- utilice frases o abreviaturas de frases
- no incluya información personal
- no utilice contraseñas comunes/genéricas.

Tenga en cuenta que una contraseña NO debe incluir información personal como la fecha de nacimiento, el nombre de una mascota, su nombre o su dirección de correo electrónico.

Aunque puedas utilizar medidas técnicas para asegurarte de que los usuarios eligen contraseñas fuertes, es imposible controlar lo que los usuarios hacen con esas contraseñas. Pueden escribirla en un papel junto a su ordenador, compartirla con otras personas o utilizarla para otras cuentas. Esto último es especialmente peligroso, ya que el uso de la misma contraseña en varias cuentas comprometerá a todas ellas si alguno de los servicios que utilizan las mismas contraseñas es vulnerado. Esto significa que, si utilizas la misma contraseña para acceder a tu biblioteca y a tu cuenta de correo electrónico, y alguien vulnera la seguridad de la biblioteca (lo que debería ser mucho más fácil que los servidores de un gran proveedor de servicios de correo electrónico, por ejemplo, Google) y roba la contraseña, puede utilizar esa información para acceder a tu cuenta de correo electrónico. Educar a los usuarios es la única solución real para evitar estas malas prácticas. Por lo tanto, hay que educar a los usuarios (sobre cómo) para que creen contraseñas seguras, no las escriban en ningún lugar accesible y nunca reutilicen la misma contraseña.

Existen otras buenas prácticas a la hora de proteger las contraseñas por parte del usuario:

- Utilice diferentes contraseñas para diferentes escenarios
- Utilizar una frase de contraseña
- Utilizar un gestor de contraseñas
- Utilice la autenticación de dos factores (2FA)

Aunque pueda parecer inocente, utilizar la misma contraseña en muchos sitios web es un asunto peligroso. Las filtraciones de información personal en los sitios web de los consumidores están cada vez más extendidas. Si le roban su información en una red social y usted utiliza la misma contraseña en su aplicación de banca electrónica y en algunos sitios de compra en línea, el atacante tendrá vía libre para acceder a todos esos sitios. Hay aplicaciones y software que pueden avisarte de que tus contraseñas han formado parte de una filtración de datos. Si su información se ha filtrado, el gestor de contraseñas de Google puede, por ejemplo, avisarle.

Interaktivní prvek

La siguiente buena práctica es utilizar más de una palabra. Una frase de contraseña es una secuencia de palabras que parece una frase, pero que no debe tener ningún significado. La frase de contraseña no debe incluir información personal de fácil acceso, al igual que la contraseña. Incluso puedes utilizar

generadores para crear una cadena de palabras aleatoria para el usuario.

Interaktivní prvek

Por último, se pueden utilizar diferentes servicios en línea para comprobar si su contraseña ha sido violada. Uno de los servicios más conocidos de este tipo es el sitio web haveibeenpwned.com. Sin embargo, existen otros servicios web como éste.

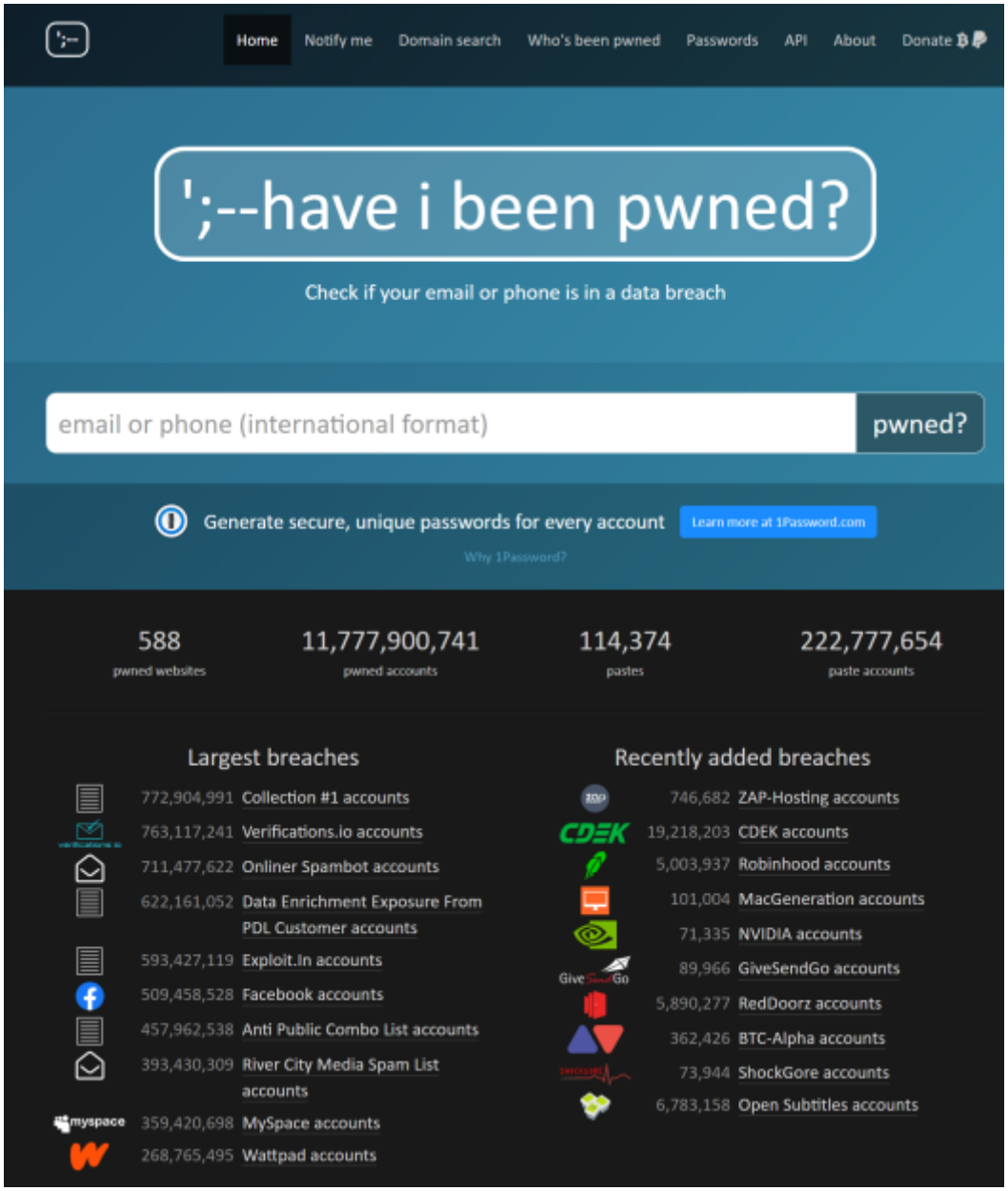


Fig. 19. El servicio web "'have i been pwned?".

4.2 Administradores de Contraseñas

Además de la autenticación de 2 factores, usar un administrador de contraseñas es una buena manera de mantener las contraseñas seguras. Esta opción funciona casi universalmente, mejora la seguridad de sus contraseñas y hace que el procedimiento de inicio de sesión sea más conveniente que menos.

Los buenos administradores de contraseñas, por otro lado, cifran sus datos a fondo. Incluso si un atacante obtiene acceso al archivo de datos, primero debe decodificarlo antes de extraer cualquier información útil. En la mayoría de las circunstancias, esto es extremadamente difícil y no vale la pena el esfuerzo. En comparación con la alternativa, el uso regular de un administrador de contraseñas garantiza que siempre tenga una lista de cuentas que se pueda actualizar según sea necesario.

Los atacantes son disuadidos por el almacenamiento de datos localmente en lugar de en la nube. Obtener los datos de una sola persona o familia requiere un esfuerzo significativo. Puede elegir dónde se almacenan los datos de su contraseña con buenos administradores de contraseñas.

Muchas personas tienen sistemas y tienden a compartir cosas en común, al igual que las contraseñas comunes. Estas tácticas son bien conocidas entre los atacantes. Ese conocimiento se ha integrado en algoritmos de craqueo y, como resultado, los sistemas pueden ser más dañinos que beneficiosos. Supongamos que su sistema es excelente por el bien del argumento. Aun así, todavía hay un riesgo. A medida que las infracciones adicionales exponen su contraseña, aumenta la probabilidad de que su sistema sea pirateado. Una vez que su sistema ha sido violado, el atacante puede usarlo para adivinar sus credenciales de inicio de sesión en otros sitios web. Finalmente, utilizar su sistema en lugar de un administrador de contraseñas es en realidad más lento.

Las frases de contraseña largas suelen ser preferibles a las contraseñas. Sin embargo, muchos sitios web no las admiten (restricciones de caracteres y longitud). Además, aunque son más fáciles de recordar que las contraseñas, no resuelven el problema de las personas que utilizan la misma contraseña en varios sitios o que dependen de un sistema.

Sin embargo, como contraseña maestra de su gestor de contraseñas, que es la llave que abre todos los datos de las contraseñas, se recomienda utilizar una frase de contraseña. Esto le permite elegir y memorizar una contraseña maestra extremadamente larga con facilidad.

Si considera almacenar sus contraseñas en un navegador, hay una serie de razones por las que esta solución es mala. Los navegadores no se toman la seguridad de las contraseñas tan en serio como deberían porque no requieren una contraseña maestra para acceder. Lo único que hay que hacer es iniciar la sesión en el ordenador. Cuando se utilizan ordenadores distintos del propio, este método es un inconveniente. Las contraseñas de los navegadores sólo son efectivas dentro del navegador. Esta no es una opción viable hoy en día, ya que la contraseña se requiere con frecuencia tanto para la web como para las aplicaciones móviles.

Los administradores de contraseñas brindan la capacidad de almacenar más que simples contraseñas, lo cual es bastante útil. Puede usar un administrador de contraseñas para guardar y completar credenciales, como tarjetas de crédito, lo que también se puede hacer con navegadores. También puede mantener otra

información confidencial en el administrador de contraseñas, como licencias, identidades, números de cuentas bancarias y cualquier otra información.

Por lo tanto, considere su administrador de contraseñas como una caja fuerte digital que puede llevar consigo.

ADVANTAGE

Además, los gestores de contraseñas también ofrecen otras ventajas relacionadas con la usabilidad:

- Está integrado con las áreas donde necesita usar contraseñas, y es rápido y simple crear, actualizar y completar contraseñas.
- Es compatible con muchas plataformas y las contraseñas están sincronizadas.
- Se comporta de una forma admirable en una amplia gama de circunstancias y diseños y, por lo general, está bien mantenido.
- Se toma en serio la seguridad y se esfuerza (cifrado sólido de extremo a extremo) para garantizar que los datos estén seguros incluso si se produce una filtración.

Para que un administrador de contraseñas sea eficaz, hay que recordar algunos puntos clave:

- Debe utilizarse en todos sus sitios web. En todas partes. Las excepciones solo añaden vulnerabilidad y complejidad al sistema (es menos probable que tenga éxito).
- Para cada sitio, cree una contraseña única. Si tiene la oportunidad, hágala lo más largas posible. Se debe utilizar un mínimo de 20 a 30 caracteres. Cuanto más difícil sea descifrar una contraseña, más larga es (en realidad, exponencialmente más difícil). Debido a que el administrador de contraseñas las completará por usted la mayor parte del tiempo, no tendrá que teclearlas.
- No todos los sitios web son totalmente compatibles con los administradores de contraseñas y, en ocasiones, las contraseñas deben copiarse en el portapapeles antes de pegarse en un formulario de inicio de sesión. Esto no es muy seguro porque podría exponer su contraseña de varias maneras. Después de un breve período de tiempo, la mayoría de los administradores de contraseñas eliminarán automáticamente los datos de contraseña del portapapeles.
- Algunos servicios siguen imponiendo requisitos de contraseña absurdos, como que las contraseñas tengan al menos 12 caracteres. Aquí los administradores de contraseñas son útiles ya que generan tales contraseñas al azar, que es lo más seguro que puedo obtener dadas las restricciones.
- Haga que su contraseña maestra sea lo más larga posible y use una contraseña que sea difícil de adivinar. También es una buena idea cambiar su contraseña maestra regularmente para reducir el peligro de que un registrador de claves la filtre o la intercepte.
- Si desea intercambiar contraseñas, haga que la otra persona cree su propia caja fuerte digital y use la función de compartir del administrador de contraseñas para compartir contraseñas

individuales con ellos.

Algunos de los administradores de contraseñas populares incluyen:

- LastPass
- Dashlane
- LogMeOnce
- 1Password
- Keeper
- KeePass

Algunos de ellos vienen gratis; tienes que pagar por algunos otros. Y algunos son gratuitos, pero se debe pagar por funciones avanzadas.

Interaktivní prvek

4.3 Autenticación de dos factores (2FA)

DEFINITION

2FA es una capa adicional de protección que verifica que cualquiera que intente acceder a una cuenta en línea es quien dice ser.

El usuario debe proporcionar previamente su nombre de usuario y contraseña. Luego se les pedirá que envíen otra información antes de que puedan recibir acceso.

Retirar dinero de un cajero automático es un buen ejemplo de autenticación de dos factores. La transacción solo se puede completar con la combinación correcta de una tarjeta bancaria (algo que tienes) y un PIN (número de identificación personal, algo que sabes).

La mayoría de los sitios web ofrecen la posibilidad de verificación basada en SMS. Sin embargo, los dispositivos móviles están ganando terreno a la 2FA.

ADVANTAGE

Las ventajas son evidentes:

- Debido a que emplea dispositivos móviles que (generalmente) se llevan todo el tiempo, no se requieren tokens adicionales.
- Los códigos de acceso generados dinámicamente son más seguros de usar que la información de inicio de sesión fija (estática) ya que cambian continuamente.

DISADVANTAGE

Sin embargo, tiene también desventajas:

- Inconvenientes: siempre que se requiera autenticación, los usuarios deben tener un teléfono móvil cargado y dentro del alcance de una red celular. El acceso a menudo es imposible sin planes de soporte si el teléfono no puede mostrar mensajes, o si está dañado o se apaga para una actualización o debido a temperaturas extremas (por ejemplo, exposición invernal). Es posible que los mensajes de texto no lleguen de inmediato, causando demoras adicionales en el procedimiento de autenticación debido a copiar y pegar o teclear manualmente.

Tenga en cuenta que los SMS no son tan seguros como cabría esperar. Las transmisiones de texto SMS a teléfonos móviles son inseguras y vulnerables a la intercepción. Como resultado, terceros pueden robar y usar el token. La 2FA del teléfono móvil a menudo se omite durante la recuperación de la cuenta. Los teléfonos inteligentes modernos se utilizan para consultar el correo electrónico y recibir mensajes de texto. En la mayoría de los casos, el correo electrónico siempre está conectado. Debido a que el teléfono puede recibir el segundo factor, si el teléfono se pierde o es robado, todas las cuentas para las que el correo electrónico es la clave pueden ser pirateadas. Como resultado, los teléfonos inteligentes fusionan

los dos criterios en uno. Si se roba el teléfono de un usuario, el delincuente puede obtener acceso a las cuentas del usuario. Los piratas informáticos pueden obtener acceso a las redes de telefonía móvil mediante la clonación de tarjetas SIM. Si un dispositivo no admite SMS, la autenticación de dos factores a través de una llamada de voz es una opción viable para prácticamente todos los demás.

La App de móvil Authy

Supongamos que tiene un smartphone u otro dispositivo móvil. En ese caso, puede adquirir su código de autenticación de dos factores sin usar SMS o llamadas de voz descargando e instalando una de las muchas aplicaciones populares de autenticación de dos factores directamente en su dispositivo. Esta es una forma mucho más segura de iniciar sesión con la autenticación de dos factores. Aplicaciones como Authy y Google Authenticator producen un **TOTP** (***Time-based One-Time Passcode***) justo dentro de la aplicación.

ADVANTAGE

Incluso si un atacante consiguiera convencer a su proveedor de servicios móviles para que hiciera un cambio de SIM, seguiría sin poder acceder a sus códigos de autenticación. La información necesaria para producir esos códigos se almacena en el propio dispositivo y no en la tarjeta SIM.

Querrá configurar sus primeras cuentas 2FA ahora que ha instalado Authy en su teléfono. Esto se hace escaneando un código QR (dado por el sitio donde desea asegurar una cuenta) usando la aplicación. Probablemente comenzará a proteger otras cuentas una vez que haya capturado su código inicial y protegido su primera cuenta.

Fig. 20. Escanear un código QR code en la App de móvil Authy.

Ahora debe elegir entre mantener todos sus tokens 2FA en un solo dispositivo y hacer una copia de seguridad en la nube.

DISADVANTAGE

Si elige la primera opción, luego pierde, actualiza o le roban su dispositivo, tendrá que convencer a todos los servicios en los que ha activado 2FA para que lo apaguen. Después de eso, cuando cambie su teléfono, deberá volver a ingresar a su cuenta y volver a habilitar manualmente 2FA en cada servicio.

ADVANTAGE

Es por eso que Authy le permite hacer una copia de seguridad de sus tokens 2FA en su almacenamiento seguro en la nube, al que solo puede acceder usted, para que siempre pueda restaurar sus cuentas si pierde, le roban o reemplaza un dispositivo obsoleto.

Le piden que establezca una contraseña de respaldo cuando realiza una copia de seguridad de los tokens 2FA en la nube, y utilizan esta contraseña para cifrar sus datos y luego sincronizarlos con su servicio en la nube. Sus datos están extremadamente seguros en su plataforma en la nube porque nunca almacenan físicamente su contraseña, pero es fundamental que la recuerde.

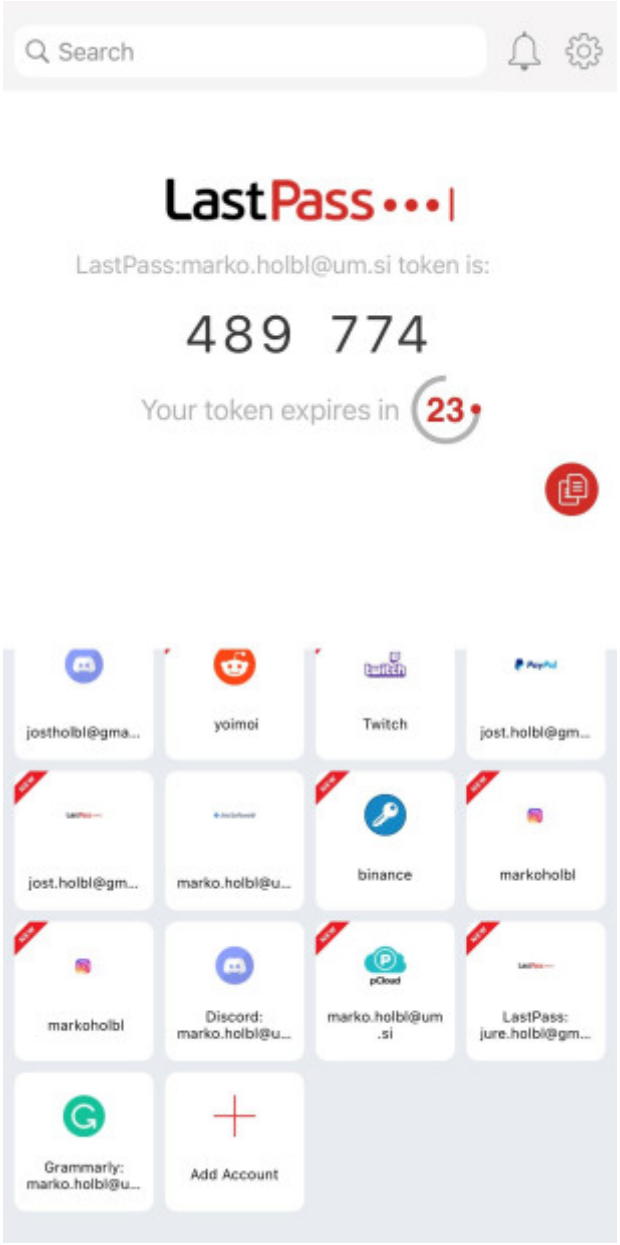


Fig. 21. Utilizando la App de móvil Authy.

Después de eso, se recomienda encarecidamente que instale Authy en otro dispositivo. La aplicación sincronizará automáticamente los tokens con cada dispositivo en el que tenga Authy instalado si los ha sincronizado con la nube de Authy. También puede descargar la aplicación Authy Desktop que es

independiente del navegador si solo tiene un dispositivo móvil.

Interaktivní prvek

4.4 Aspectos del Almacenamiento Seguro de Contraseñas (lado del servidor)

Las contraseñas deben protegerse adecuadamente del lado del autenticador (generalmente el servidor). Diariamente se informa de muchas filtraciones de datos y, por lo tanto, no se puede confiar en la seguridad de los sistemas de autenticadores. Por lo tanto, las contraseñas no se pueden almacenar en texto sin formato y deben almacenarse de manera protegida. Sin embargo, primero presentaremos brevemente algunos conceptos necesarios para comprender el almacenamiento seguro de contraseñas.

4.4.1 Almacenamiento de contraseñas hash

DEFINITION

Una función hash criptográfica toma una entrada (o mensaje) y devuelve una cadena alfanumérica de longitud fija.

La cadena alfanumérica se conoce como valor hash, compendio del mensaje, huella digital, resumen o suma de comprobación.

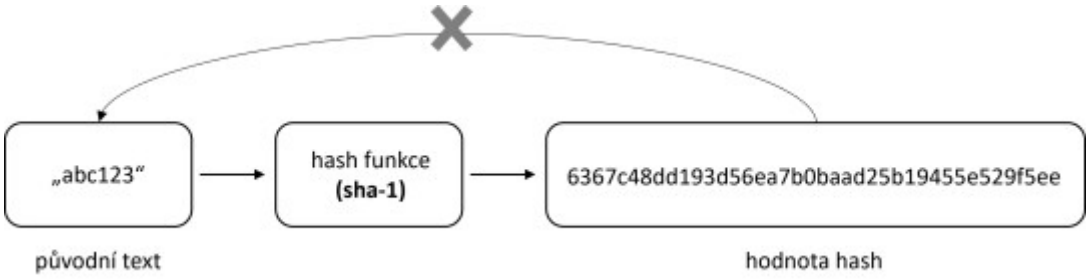


Fig. 22. Cómo funciona el hash.

La Figura 22 muestra el proceso de *hashing*. Comenzamos con la palabra "abc123" y usamos la función hash SHA-1 para obtener una salida alfanumérica de tamaño fijo, que llamamos valor hash. No podremos recuperar nuestro texto de entrada original usando este valor hash. No podemos invertir un valor hash para encontrar el contenido original, ya que las funciones hash son unidireccionales y, por lo tanto, irreversibles. Si el mismo material se pasa a través de la misma función hash, el resultado de salida/hash debería ser el mismo. Entonces, en lugar de guardar la contraseña en texto sin formato, podemos codificarla con la función hash y guardar el valor hash.

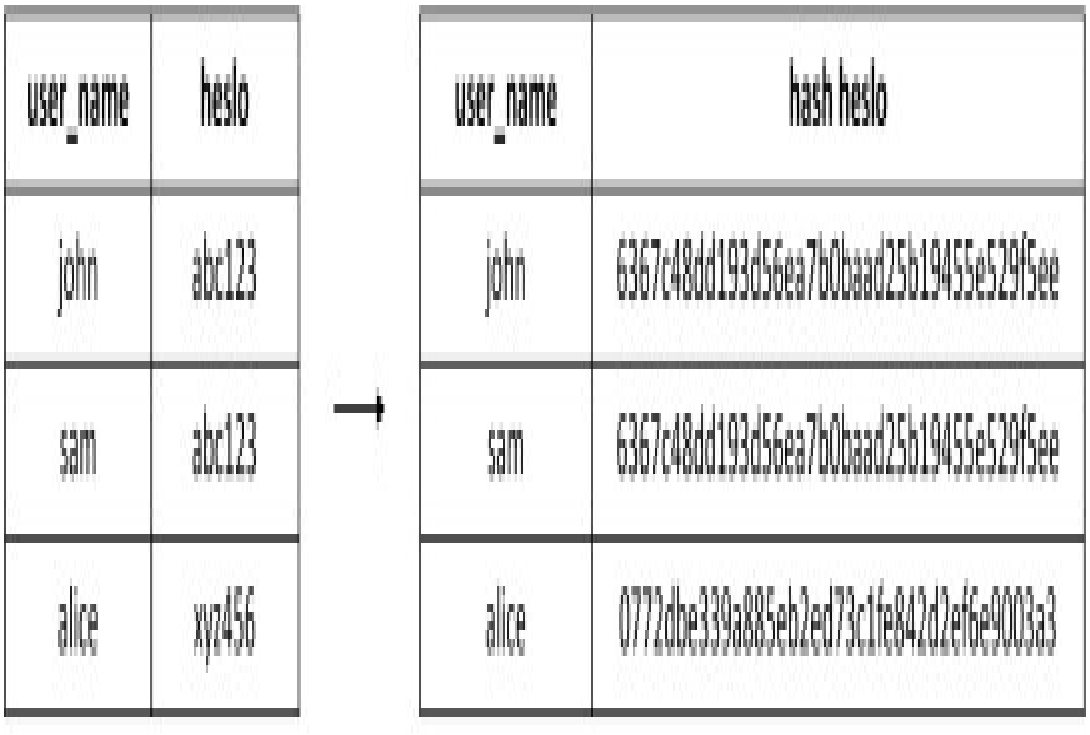


Fig. 23. Protección de contraseñas almacenadas mediante función hash.

Cuando un usuario intenta iniciar sesión en el sistema, la función hash se usa para codificar la contraseña del usuario y compararla con el valor hash almacenado en la tabla. Podemos permitir que el usuario inicie sesión en el sistema si ambos valores hash son iguales. En la Figura 23, tanto John como Sam tienen la misma contraseña, "abc123", y sus valores hash son los mismos después de aplicar el algoritmo hash. Considere el caso en el que John tiene acceso a la base de datos y puede ver la contraseña hash. Entonces John puede ver que el valor hash de su contraseña es el mismo que el valor hash de la contraseña de Sam. Como resultado, John podrá usar las credenciales de Sam para iniciar sesión en el sistema. Para evitar esto, podemos usar una técnica llamada *salting*.

4.4.2 *Salted* Hash

Nuestro objetivo con *salted* hash es conseguir que el valor hash de la contraseña sea único. Así, el sistema genera una colección aleatoria de caracteres llamada sal. Cuando el usuario escribe una contraseña de texto sin formato, se añade el conjunto de caracteres aleatorios producidos. A continuación, se utiliza la función hash para extraer el valor hash del texto insertado (*salted* hash). En este caso, debe guardarse el valor de sal de cada usuario.

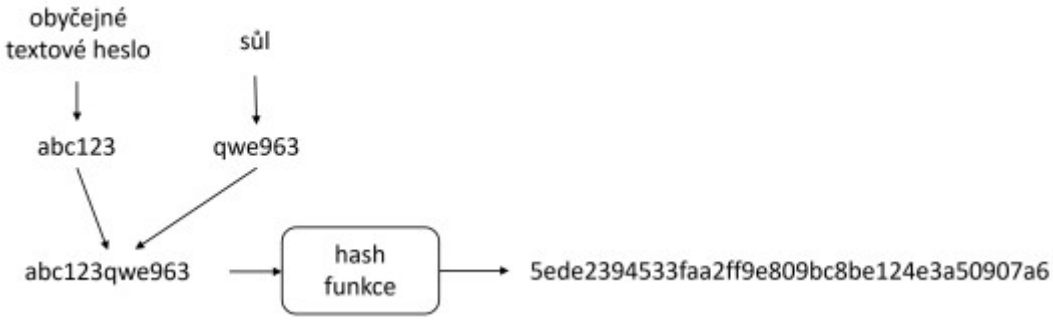


Fig. 24. El proceso de *salted* hash.

Aunque john y sam tienen la misma contraseña, sus valores hash son diferentes (ver Figura 25).

Interaktivní prvek

Durante el proceso de inicio de sesión, el sistema recupera el valor apropiado de sal del usuario de la base de datos, lo agrega a la contraseña de entrada, lo pasa a través de la función hash y compara el valor hash resultante con el valor hash registrado en la tabla. El usuario se autentica correctamente si ambos valores hash coinciden.

user_name	hodnota soli	solené hash heslo
john	qwe963	5ede2394533faa2ff9e809bc8be124e3a50907a6
sam	hjk521	6367c48dd193d56ea7b0baad25b19455e529f5ee
alice	asd753	0772dbe339a885eb2ed73c1fe842d2ef6e9003a3

Fig. 25. Ejemplo de una tabla que almacena contraseñas con sal y hash.

La protección mínima de las contraseñas almacenadas debe incluir la función hash y el uso de la sal. Además, NIST sugiere bloquear a un usuario fuera del sistema si usa una contraseña incorrecta demasiadas veces (por ejemplo, después de tres intentos fallidos, el usuario no puede volver a intentarlo durante un minuto), permitir emojis, ASCII y caracteres Unicode en las contraseñas y permitir funciones de "copiar y pegar" en los campos de contraseña para que el uso de administradores de contraseñas y la autenticación multifactor sea más conveniente.

Interaktivní prvek

CHAPTER 5

Autenticación sin Contraseña

Dados todos los problemas y debilidades de las contraseñas, la idea de la autenticación sin contraseña no es nueva. Como su nombre lo indica, la autenticación sin contraseña permite que un usuario inicie sesión o adquiera acceso sin teclear una contraseña ni responder a preguntas de seguridad. La autenticación sin contraseña reduce la necesidad de contraseñas peligrosas y su administración, al mismo tiempo que mejora la seguridad de las cuentas de usuario al reducir su vulnerabilidad a los ataques. Existen diferentes mecanismos de autenticación sin contraseña, como tarjetas de proximidad, tokens físicos, dispositivos/llaves USB, enlaces mágicos, reconocimiento biométrico, aplicaciones móviles, etc. La mayoría de estas técnicas se utilizan comúnmente en la autenticación multifactor para aumentar la seguridad. Sin embargo, algunas de estas soluciones se pueden emplear como un sistema de autenticación de primer factor.

Estos elementos se dividen típicamente en dos categorías:

- Ejemplos de elementos de propiedad son teléfonos inteligentes, tokens OTP, tarjetas inteligentes o tokens de hardware ("algo que el usuario tiene").
- Las huellas dactilares, los escáneres de retina, el reconocimiento facial o de voz y otros identificadores biométricos son ejemplos de factores inherentes ("lo que es el usuario").

La autenticación sin contraseña a menudo se confunde con la ***autenticación multifactor*** (**MFA**) porque ambas usan una variedad de factores de autenticación. Sin embargo, mientras que MFA se usa como una capa adicional de seguridad además de la autenticación basada en contraseña, la autenticación sin contraseña no requiere un secreto memorizado y, por lo general, usa solo un factor de alta seguridad para autenticar la identidad, lo que lo hace más rápido y fácil para los usuarios.

Las contraseñas son difíciles de recordar y los requisitos son cada vez más complejos. Diferentes sitios pueden tener diferentes políticas de contraseñas, por lo que una contraseña generada para un sitio podría no funcionar en otro. Recordar una contraseña generada para una política de contraseña mejorada moderna suele ser un desafío.

Al igual que ***Fast Identity Online*** (**FIDO**), aquí entran en juego diferentes estándares. Aunque la autenticación sin contraseña y la tecnología FIDO existen desde hace algún tiempo, los servicios en línea y los proveedores de identidad aún no los utilizan a gran escala. La autenticación sin contraseña se convertirá en el futuro de la autenticación, gracias a la incorporación de capacidades biométricas en la mayoría de los dispositivos móviles y portátiles modernos.

ADVANTAGE

La autenticación sin contraseña mejora la experiencia del usuario final al eliminar la fatiga de la contraseña. El usuario ya no necesita crear una contraseña más larga y segura y puede recibir acceso unificado a todos los programas simplemente conectándose a un dispositivo USB o escaneando su huella digital.

5.1.1 Identidad rápida en línea

La identidad rápida en línea o ***Fast Identity Online*** (**FIDO**) es un conjunto de protocolos de autenticación de código abierto establecidos por FIDO Alliance para eliminar las contraseñas. Para implementar una autenticación segura, los protocolos FIDO utilizan algoritmos básicos de criptografía de clave pública. Las claves privadas nunca saldrán del dispositivo de seguridad y todas las conversaciones estarán encriptadas.

Fig. 26. Ejemplo de autenticación basada en FIDO.

FIDO Alliance ha emitido tres conjuntos de estándares:

- **UAF** (***Universal Authentication Framework***): La opción de autenticación sin contraseña está incluida en el protocolo FIDO UAF. Los usuarios que utilizan este protocolo deben firmar un desafío proporcionado por el servidor FIDO utilizando uno o más factores de seguridad disponibles en su dispositivo de seguridad/digital.
- **U2F** (***Universal Second Factor***): La opción de autenticación con un segundo factor la proporciona el protocolo FIDO U2F. Para establecer su identidad, los usuarios deben proporcionar dos pruebas. Este ha sido renombrado a CTAP1 con el lanzamiento del protocolo FIDO2.
- FIDO2: Se trata del conjunto de especificaciones más reciente de FIDO Alliance.

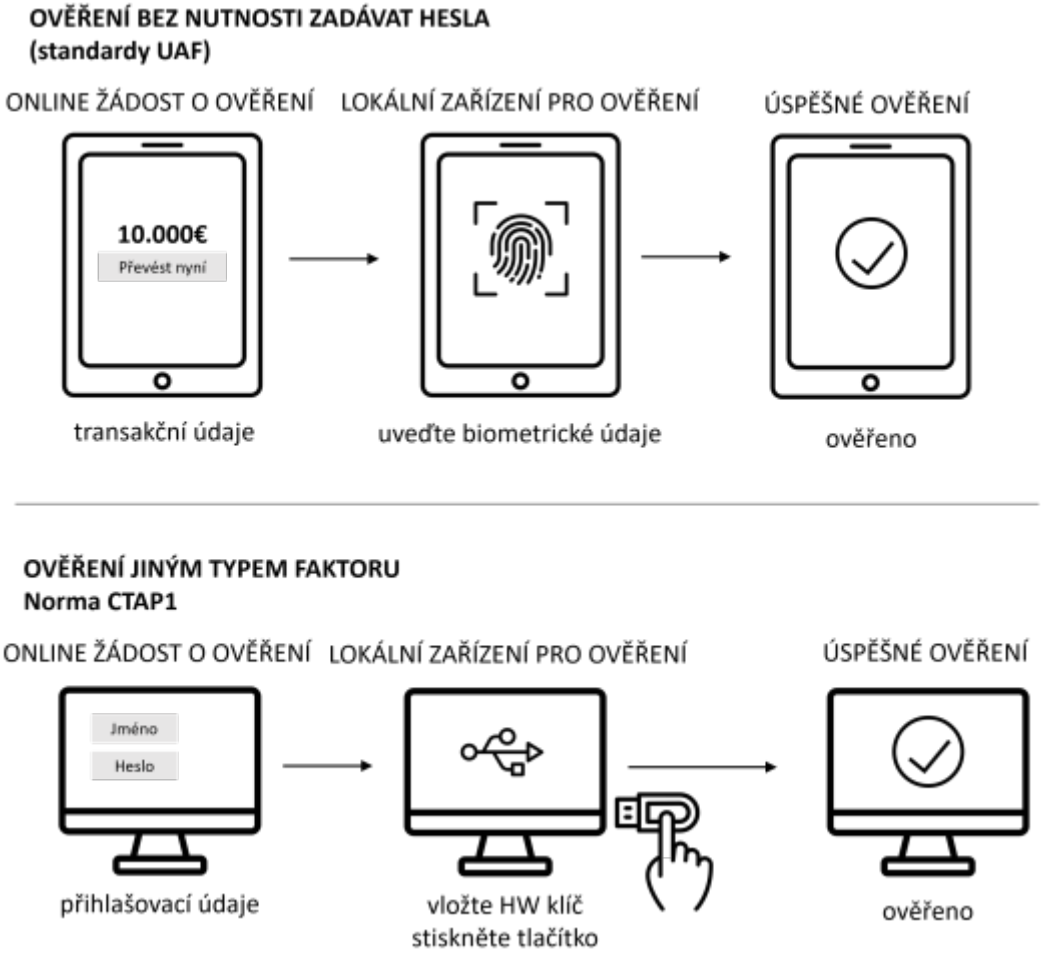


Fig. 27. Estándares UAF y U2F (CTAP1) para la autenticación sin contraseña.

5.1.2 FIDO2 y Webauthn

La especificación FIDO2 consta de:

- Estándar de autenticación web W3C (WebAuthn) y
- el Cliente FIDO para el Protocolo de Autenticación 2 (CTAP2).

FIDO2 permite a los usuarios usar dispositivos ordinarios para autenticarse sin esfuerzo en los servicios de Internet tanto en contextos móviles como de escritorio. WebAuthn es una API estándar en línea para la autenticación FIDO que se incorpora a plataformas y navegadores. CTAP2 es una versión de CTAP que permite a los usuarios usar autenticadores externos e integrados ofrecer autenticación sin contraseña, de dos factores o de múltiples factores. La API de WebAuthn es una herramienta para crear y administrar credenciales de clave pública. En la Figura 28 se muestra una descripción general del método de autenticación FIDO2.

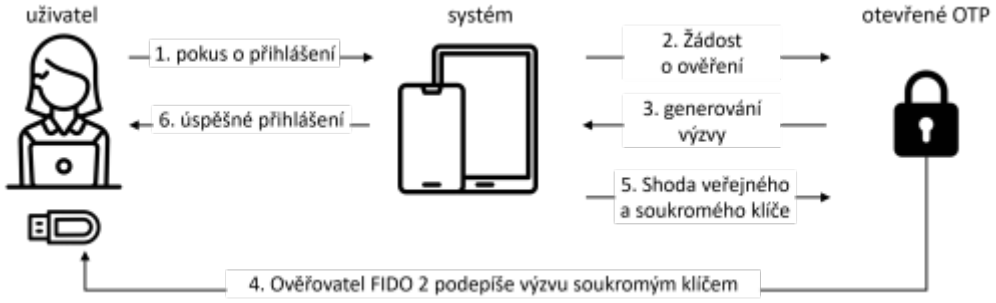


Fig. 28. Método de autenticación FIDO2.

Interaktivní prvek

CHAPTER 6

Introducción a la Firma Digital

DEFINITION

Se conoce como firma digital a un sistema matemático para verificar la validez de mensajes o documentos digitales.

Una firma digital genuina le da al destinatario una buena razón para creer que el mensaje fue creado por un remitente conocido (autenticidad) y que no fue alterado en tránsito si se cumplen los requisitos previos (integridad).

Los principales objetivos que una firma digital se esfuerza por alcanzar son:

- **Autenticación**: las firmas digitales están vinculadas a un usuario específico a través de su clave privada. Como resultado, pueden determinar quién posee la clave privada utilizada para firmar los datos/mensajes originales (por ejemplo, documento, correo electrónico o archivo). Consulte más abajo información sobre las claves públicas y privadas.
- **Integridad**: se utiliza una técnica de hash en las firmas digitales para garantizar que no se altere un mensaje. Consulte también más abajo información sobre el hash.

Entonces, las firmas digitales son una de las formas de autenticar una entidad, pero primero debemos aclarar algunos conceptos antes de que se pueda mostrar cómo se puede usar una firma digital en la autenticación.

6.1 Criptografía de Clave Pública

Para comprender las firmas digitales, primero debemos explicar la criptografía asimétrica, a menudo denominada criptografía de clave pública. En contraste con el cifrado clásico (simétrico), que usa solo una clave para el cifrado, el cifrado asimétrico emplea un par de claves. El cifrado es el proceso de codificación de información, como se muestra en la Figura 29.

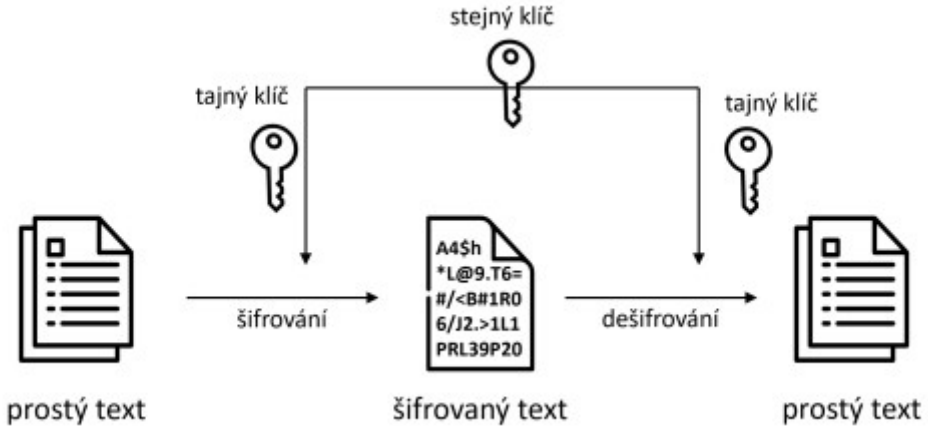


Fig. 29. Cifrado simétrico

Imagina que deseas enviar a alguien un mensaje encriptado usando el encriptado clásico. En este escenario, ambas partes deben acordar una clave única. No pueden transmitirlo entre sí porque entonces alguien podría verlo y podrá ver todos sus mensajes.

DEFINITION

Por otro lado, el cifrado asimétrico emplea un par de claves, una clave pública y una clave privada que están relacionadas matemáticamente. Solo la clave privada vinculada puede descifrar lo que está cifrado con la clave pública.

Ahora, si alguien quiere que otros le envíen mensajes encriptados, simplemente ha de publicar su clave pública para que todos la vean. Luego, sólo debe utilizar su clave privada para descifrar los mensajes cifrados con su clave pública que le lleguen, ya que los mensajes cifrados con su clave pública solo se pueden descifrar con su clave privada. Esto es útil porque así no tenemos que preocuparnos por compartir la clave pública de forma segura.

En pocas palabras, para que dos partes se comuniquen de forma segura mediante el cifrado asimétrico, el proceso es el siguiente:

- Las claves públicas se intercambian entre las dos partes.
- La persona 1 encripta el mensaje que desea enviar utilizando la clave pública de la persona 2 y se lo envía a la persona 2.
- La persona 2 descifra el mensaje con su clave privada.

Este proceso se representa en la Figura 30.

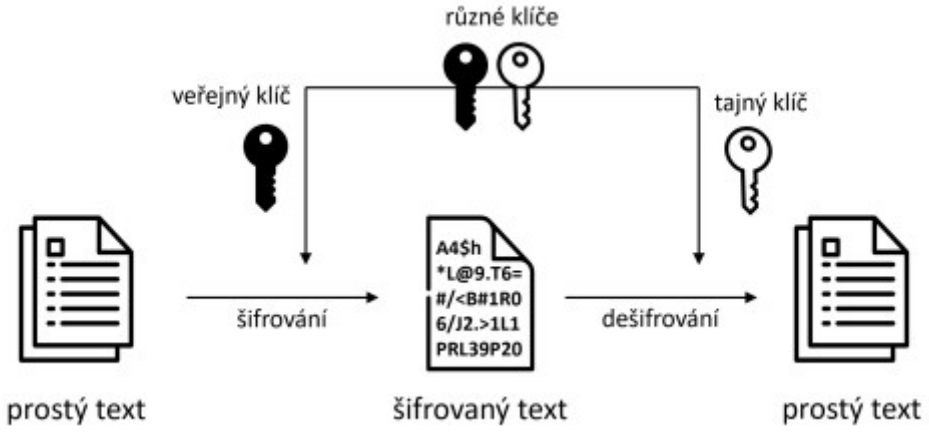


Fig. 30. Proceso de cifrado asimétrico (clave pública).

DEFINITION

Las firmas digitales funcionan firmando (cifrando) cualquier cosa con una clave privada, que luego es validada por la clave pública que está conectada con ella. En este caso, el par de claves se utiliza de manera opuesta.

Esto se debe a que el firmante es la única persona con acceso a la clave privada utilizada para realizar la firma. Por lo tanto, se puede estar seguro de que fue esa persona quien firmó. Cualquiera puede usar la clave pública para verificar (es decir, descifrar con éxito el mensaje) que el propietario de la clave pública creó el mensaje.

Interaktivní prvek

6.2 El Proceso de Firma Digital

Como ya se mencionó en la firma digital, se utiliza un par de claves, una clave pública y una clave privada. Este par de claves criptográficas se utiliza para cifrar (bloquear) y descifrar (desbloquear) datos de origen de la misma manera que las claves físicas se utilizan para bloquear y desbloquear. Las claves privadas se mantienen seguras y confidenciales porque si alguien conoce la clave privada de otra persona, puede firmar los datos de origen como esa persona. Por otro lado, se supone que las claves públicas se comparten con todos. Los datos cifrados por la clave privada solo se pueden descifrar con la clave pública, revelando los datos originales.

DEFINITION

El proceso de firma digital incluye criptografía asimétrica y funciones hash.

Estos dos bloques de construcción se combinan para formar el proceso de firma real de la siguiente manera:

1. Usando una función hash, el remitente calcula el hash del contenido de origen que desea enviar.
2. El remitente cifra el hash calculado con su clave privada para crear una firma digital.
3. Luego, el contenido y la firma digital se pueden enviar al destinatario.
4. Después de que el destinatario reciba los mensajes, el receptor utiliza la clave pública disponible públicamente del remitente para descifrar la firma digital cifrada del remitente. Si tiene éxito, se confirma la identidad del remitente como propietario de la clave privada utilizada para cifrar el archivo.
5. El receptor luego obtiene el contenido original del mensaje recibido y genera un hash de este contenido.
6. El contenido se valida como idéntico a lo que proporcionó el remitente si el hash calculado del receptor coincide con el del remitente. Si los valores hash no coinciden, el contenido se ha manipulado y, por lo tanto, la firma no es válida.

Una representación gráfica del proceso se muestra en la Figura 31.

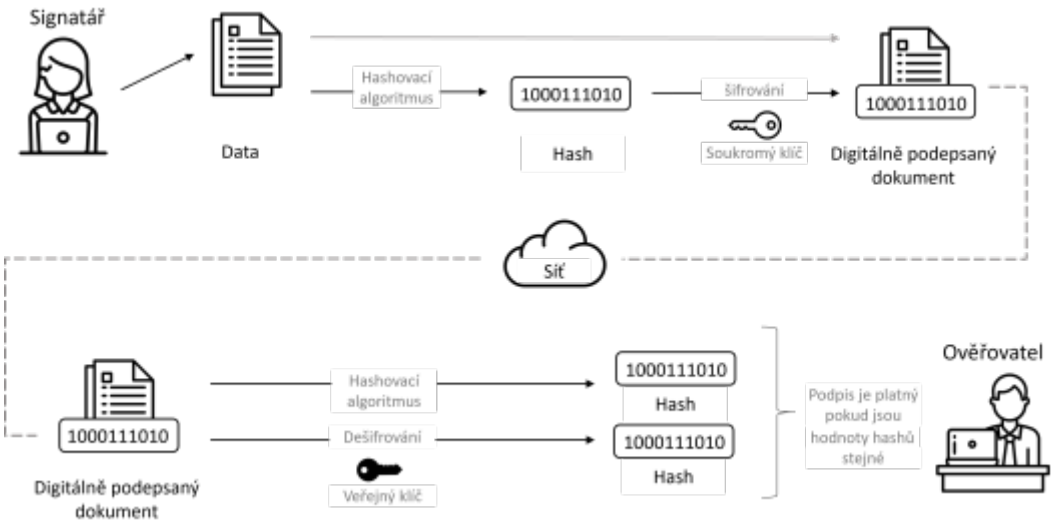


Fig. 31. Representación esquemática de la creación y verificación de una firma digital.

Debido a que la clave pública del remitente está disponible públicamente, cualquiera puede descifrar el contenido cifrado que envía el remitente. Como resultado, este método de cifrado solo verifica la integridad, no la confidencialidad.

Uno se preguntaría por qué generamos un valor hash de los datos antes de firmarlos. Simplemente porque hace que la firma sea mucho más pequeña y que el proceso de creación y verificación de la firma digital sea más rápido, ya que solo se comparan los valores hash con los datos/documentos completos. Tenga en cuenta que funciona porque los algoritmos hash siempre producen un valor de una longitud determinada.

ADVANTAGE

Como habrá adivinado, las firmas digitales brindan varias ventajas, incluidas las siguientes:

- aumentan la seguridad y la confianza porque no se pueden modificar ni falsificar;
- proporcionan no repudio al cifrador de datos de origen;
- garantizan la integridad de los datos transmitidos

DISADVANTAGE

Sin embargo, también existen inconvenientes con las firmas digitales, como:

- el hecho de que no hay forma de revocar firmas (fideicomiso de datos de origen) después de que se hayan distribuido, lo que hace que los juicios sean irreversibles;
- el uso de claves públicas imposibilita el secreto. Cualquiera que tenga la clave pública puede verificar la firma.

Interaktivní prvek

Interaktivní prvek

CHAPTER 7

La Infraestructura de Clave Pública

Ya hemos aprendido sobre los conceptos de firma digital y criptografía de clave pública. Resulta que necesitamos algo más para que todo el concepto funcione correctamente en la vida real. Necesitamos algo llamado ***Infraestructura de Clave Pública*** (**PKI**).

DEFINITION

PKI es un conjunto de tecnologías, procesos y entidades que permite la comunicación segura a través de redes públicas inseguras.

Por ejemplo, PKI es lo que agrega la S a HTTPS, y si está viendo este contenido en un navegador web, presumiblemente está usando uno para asegurarse de que proviene de una fuente confiable. Las PKI permiten el acceso regulado a los sistemas y recursos, la protección de datos y la responsabilidad de las transacciones al establecer la identidad de las personas, los dispositivos y los servicios.

PKI se utiliza en una variedad de aplicaciones, incluida la seguridad de las comunicaciones en Internet de las cosas (IoT) y la firma de documentos digitales. PKI, que se basa en criptografía asimétrica, se usa comúnmente para configurar comunicaciones electrónicas seguras, como compras en línea, operaciones bancarias y correos electrónicos, y comunicaciones entre los usuarios y los sitios web a los que se conectan mediante HTTPS. PKI permite una fuerte autenticación, cifrado de datos y firmas digitales para personas, servicios y objetos al proporcionar identidades digitales. Estos métodos de seguridad brindan acceso seguro a recursos físicos y digitales, comunicación segura entre personas, servicios y cosas, y la firma digital de documentos, transacciones u otros datos.

7.1 Componentes de la Infraestructura de Clave Pública

PKI consta de los siguientes componentes:

- una ***autoridad de certificación*** (**CA**)
- una ***autoridad de registro*** (**RA**)
- una ***autoridad de validación*** (**VA**)
- certificados digitales

Y, por supuesto, ***criptografía de clave pública*** (**PKC**).

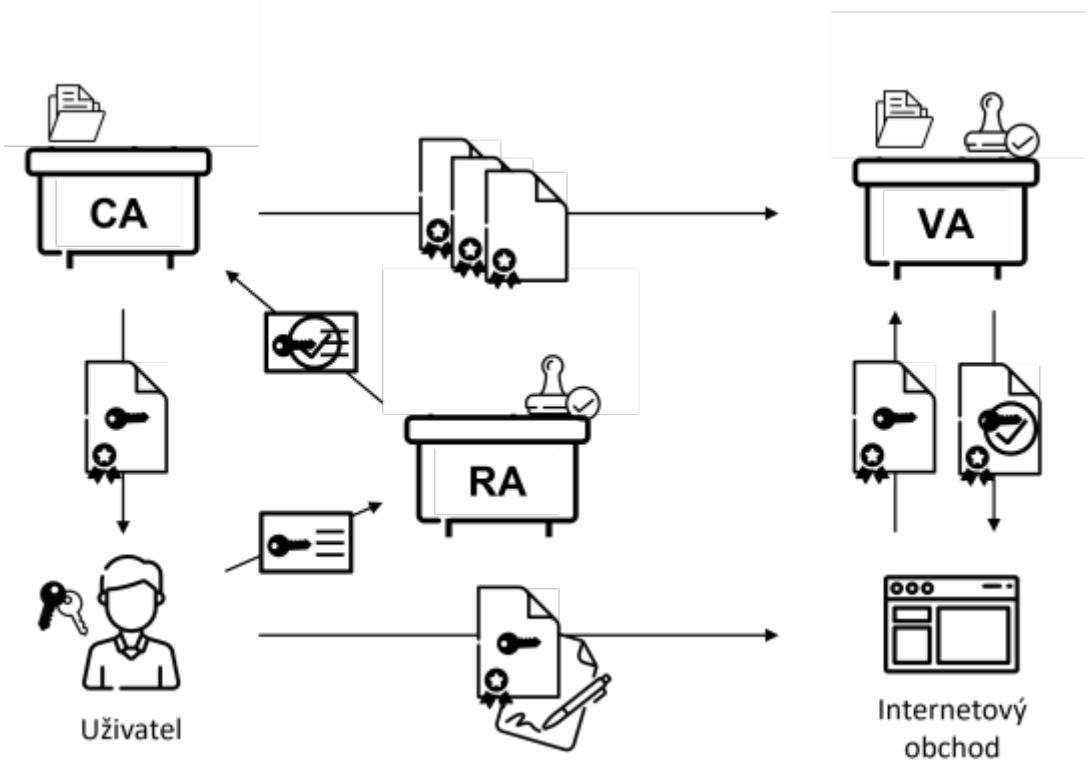


Fig. 32. Componentes de la infraestructura de clave pública (PKI).

Autoridad de Certificación

DEFINITION

Una autoridad de certificados, a menudo conocida como autoridad de certificación (CA), es una empresa que crea y distribuye certificados digitales.

Un certificado digital verifica que el sujeto designado del certificado posee una clave pública. Otros (entidades de confianza) pueden confiar en las firmas y afirmaciones sobre la clave privada que coincide con la clave pública certificada. Una CA actúa como un tercero de confianza, en el que confían tanto el sujeto del certificado (propietario) como la parte que confía en el certificado.

La firma de certificados utilizados en HTTPS, el protocolo de navegación segura, es uno de los usos más frecuentes de las autoridades de certificación. Otra aplicación popular es que los gobiernos nacionales emitan tarjetas de identidad que puedan utilizarse para la firma digital o la administración electrónica.

Autoridad de Registro

DEFINITION

En las infraestructuras de clave pública, una Autoridad de Registro (RA) es una función para el registro de certificados. Se encarga de recibir las solicitudes de firma de certificados de individuos, servidores, cosas y otras aplicaciones, ya sea para la inscripción inicial o para las renovaciones. Estas solicitudes son verificadas por la Autoridad de Registro y remitidas a una Autoridad de Certificación (CA).

Una Autoridad de Registro también se encarga de la gestión del ciclo de vida de los certificados. Consideremos el caso de la revocación. La Autoridad de Registro incluye la lógica de negocio para aceptar solicitudes, incluyendo métodos para verificar el origen del solicitante y la parte que debería tener el certificado.

Por cuestiones de accesibilidad y seguridad, una Autoridad de Registro suele estar separada de una Autoridad de Certificación. Se puede acceder a la AR a través de una interfaz gráfica de usuario (GUI) fácil de usar o mediante API y protocolos estándar que son fáciles de integrar.

Autoridad de Validación

Los certificados PKI son validados por una Autoridad de Validación (VA). El acceso a las ***Listas de Revocación de Certificados*** (**CRL**), al ***Protocolo de Estado de Certificados en Línea*** (**OCSP**) y a las descargas de certificados de la cadena de CA son ejemplos de servicios de validación de certificados. Dado que los certificados pueden ser emitidos y revocados, es vital verificar la autenticidad de un certificado antes de confiar en él. La Autoridad de Validación es la encargada de resolver esta cuestión.

La Autoridad de Certificación emisora es responsable de proporcionar actualizaciones del estado del certificado a la Autoridad de Validación según la política establecida. Se confía en que cada Autoridad de Certificación conectada emita una lista de certificados digitales revocados cuando se utilizan las CRL (CA).

Certificado Digital

Un certificado digital es un tipo de identificación electrónica para entidades individuales u organizaciones, similar a un DNI. Incluye información como la identificación, un número de serie y las fechas de caducidad. También podemos ver la firma digital de la autoridad certificadora, que garantiza la autenticidad del certificado y la clave pública del titular del mismo dentro de la información. Por ejemplo, la PKI permite conexiones autenticadas y, si se combina con otros enfoques criptográficos, también asegura las conexiones entre dos máquinas que se comunican, ya que las identidades de las dos partes pueden confirmarse mediante certificados digitales. Casi todos los certificados emitidos hoy en día se ajustan a la norma X.509.

Hay muchos tipos de certificados:

- **Certificados de firma de código**: El código ha sido validado como procedente de los desarrolladores y no ha sido modificado, lo que hace que el software sea digno de confianza. Se utiliza para firmar publicaciones de software y validar el software del vendedor o desarrollador para confirmar que es legítimo.
- **Certificados de correo electrónico**: El protocolo S/MIME puede utilizarse para salvaguardar y validar los correos electrónicos, permitiendo al remitente establecer la autoría y evitar la manipulación.
- **Certificados de firma de documentos**: Los certificados de firma de documentos de Adobe, Microsoft y otros programas de software deben utilizarse para garantizar que los documentos no han sido alterados y son dignos de confianza. Este certificado se utiliza casi siempre cuando se ve una firma digital en un documento.
- **Certificados TLS (HTTPS)**: Se utilizan para las conexiones seguras HTTPS.

Fig. 33. Ejemplo de certificado digital en Microsoft Windows.

Interaktivní prvek

7.2 La Estructura Jerárquica de la Infraestructura de Clave Pública

Una jerarquía de CAs que firman y emiten certificados digitales o credenciales es común en PKI. Las sub-CAs reciben la facultad de firmar certificados digitales para los dispositivos por parte de cada CA. Los dispositivos finales comunican los certificados digitales en la parte inferior, que son permitidos por la sub-CA que está por encima de ellos, que los generó y firmó. A veces se denominan certificaciones de dispositivo. Las sub-CAs que crean certificados de dispositivo tienen su propio certificado, que es autorizado por la firma digital de la CA que está por encima de ellas, y así sucesivamente. La PKI llega finalmente a la raíz, que sirve de base para este dominio particular del ecosistema PKI.

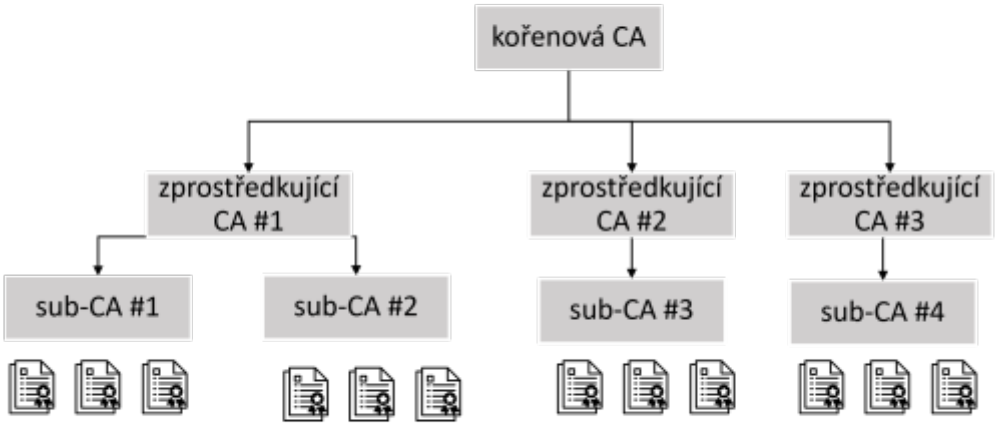


Fig. 34. Un ejemplo de jerarquía PKI.

Podemos permitir niveles específicos de revocación o denegación de acceso en caso de fuga o compromiso de una clave privada en el ecosistema colocando los ecosistemas PKI en jerarquías.

Cualquiera puede revocar el certificado de cualquier elemento de la PKI, desde el dispositivo hasta una CA de alto nivel, dependiendo de la naturaleza de la violación de la seguridad. Sin embargo, esa revocación del certificado también revoca todo lo que esté por debajo del elemento en la jerarquía.

Además, esto demuestra por qué las implementaciones de PKI se organizan en jerarquías en forma de árbol. Este diseño permite al propietario del ecosistema llevar a cabo un control de daños selectivo en caso de infracción. Por ello, la emisión de certificados de dispositivo desde la CA raíz no es una buena idea porque limita la flexibilidad. Si algo sale mal en este caso, podríamos tener que invalidar y revocar

toda la PKI y todos los dispositivos desplegados en el campo. En consecuencia, los certificados de dispositivo son emitidos prácticamente siempre por sub-CAs que están por debajo de la autoridad de certificación raíz.

7.3 Ciclo de Vida del Certificado Digital

El ciclo de vida de un certificado digital comienza con su creación y se puede explicar brevemente de la siguiente manera:

- **Inscripción del certificado**: la Autoridad de Certificación (CA) recibe una solicitud de certificado de una entidad. Una persona, un dispositivo o incluso unas pocas líneas de código pueden considerarse una entidad.
- **Emisión del certificado**: la Autoridad de Registro (RA) debe verificar la identidad del solicitante, lo que generalmente se hace a través de credenciales o basándose en la identidad de otra RA que ya verificó al solicitante.
- **Validación del certificado**: el servidor verifica con la CA cada vez que se usa el certificado para autenticar para asegurarse de que aún es válido y no ha caducado o ha sido revocado.
- **Revocación del certificado**: cuando los certificados se emiten por primera vez, tienen una fecha de caducidad que se indica. Cuando pasa esa fecha, la CA coloca el certificado en la Lista de Revocación de Certificados (CRL), una forma de lista negra que indica al servidor que no confíe en determinados certificados.
- **Renovación del certificado**: las CAs se pueden configurar para renovar automáticamente los certificados cuando alcanzan su fecha de caducidad, aunque generalmente requieren una nueva verificación de identidad.

Interaktivní prvek

Autoridades de certificación y cadena de confianza

El término "cadena de confianza" se refiere a la relación entre un certificado digital y una CA de confianza. Para que sea de confianza, un certificado debe poder rastrearse hasta la raíz de confianza que lo emitió, lo que significa que todos los certificados de la cadena -servidor, intermedio y raíz- deben ser de confianza.

En la Figura 35, podemos ver que para el servidor google.com GTS-CA 1C3 es una CA de nivel inferior. GTS-Root R1 es una CA de nivel medio. R1 es la CA raíz superior de GlobalSign. Con este método se puede establecer una cadena de confianza.

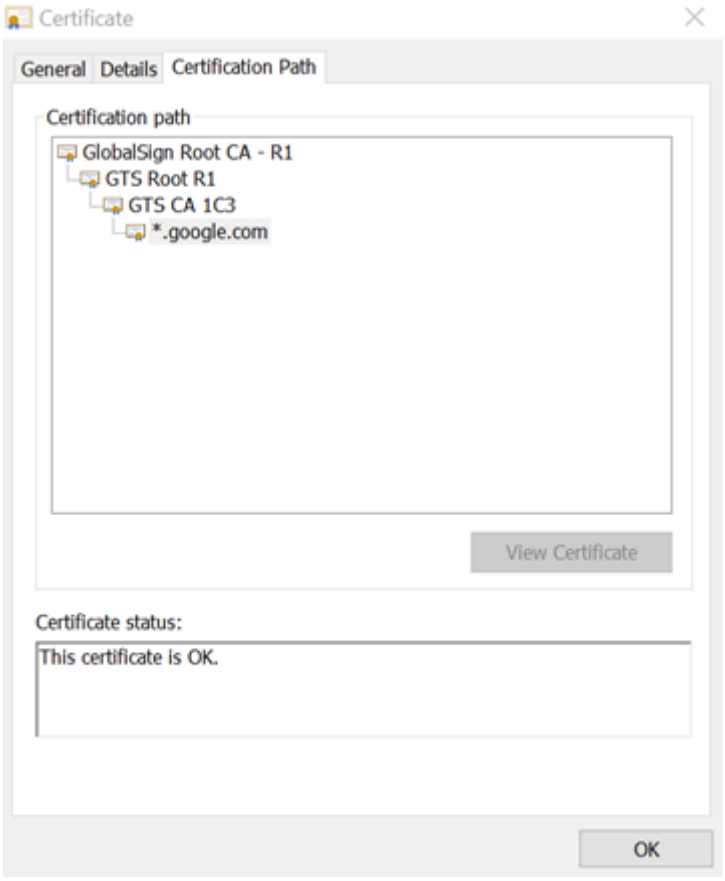


Fig. 35. Ejemplo de cadena de confianza.

La cadena de confianza consta de tres partes:

- Un **certificado raíz** es un certificado digital que pertenece a la Autoridad de Certificación que lo emitió. Por ejemplo, la mayoría de los navegadores lo traen preinstalado y se guarda en un "almacén de confianza". Las Autoridades de Certificación vigilan de cerca los certificados raíz. Por ejemplo, GlobalSign Root CA- R1 es una CA raíz.
- Los **certificados intermedios** son como las ramas de un árbol, y el certificado raíz es como el tronco del árbol. Sirven de enlace entre los certificados raíz protegidos y los certificados de servidor emitidos públicamente. Siempre habrá al menos un certificado intermedio en una cadena, pero puede haber más.
- El **certificado de servidor** es el que se ha concedido a un determinado dominio (en este caso, www.google.com).

7.4 Autenticación utilizando PKI y PKC

En el mundo digital, la infraestructura de clave pública (PKI) es un sistema para autenticar personas y dispositivos. Una o más entidades de confianza firman digitalmente documentos verificando que una clave criptográfica específica pertenece a un usuario o dispositivo específico. Luego, la clave se puede usar como la identidad del usuario en las redes digitales.

Los certificados digitales también se pueden usar en 2FA o en la autenticación sin contraseña.

Cuando un usuario intenta autenticar su identidad en un servidor, el servidor crea datos aleatorios y los envía al usuario. Luego, el usuario cifra los datos con su clave privada y los devuelve al servidor. El servidor descifra los datos utilizando la clave pública del certificado digital del usuario y, si los datos descifrados coinciden con los datos recibidos, el servidor sabe que el usuario es quien dice ser. Este es el proceso básico de uso de PKI+PKC con fines de autenticación.

CHAPTER 8

Test

La forma más segura de almacenar contraseñas es

- ◯ cifrado
- ◯ hashing
- ◯ texto sin formato
- ◯ salted hash

¿Cuál es el objetivo de la autenticación?

- ☐ comprobar la identidad de uno
- ☐ identificar a alguien
- ☐ comprobar a qué puede acceder alguien
- ☐ es una parte esencial de la ciberseguridad

Añadir sal a las contraseñas hace que sea más difícil para un atacante atacar, ya que hace que el ataque de diccionario sea específico para cada:

- ◯ usuario
- ◯ atacante
- ◯ dispositivo
- ◯ contraseña

¿Cuáles son los métodos de autenticación más comunes?

- ☐ nombre de usuario y contraseña

- ☐ escaneo facial
- ☐ email and password
- ☐ RSA SecureID

¿Cuál es la desventaja más destacada de la característica "lo que eres"?

- ◯ No es repudiable
- ◯ se puede perder
- ◯ se puede olvidar
- ◯ puede ser robada

¿Cuáles son las principales categorías de autenticación?

- ☐ Cómo te ves
- ☐ lo que sabes
- ☐ lo que eres
- ☐ lo que tienes

¿Qué es la autenticación multifactor?

- ◯ La autenticación que utiliza al menos dos factores diferentes para la autenticación
- ◯ La autenticación que utiliza exactamente dos factores diferentes para la autenticación
- ◯ La autenticación que utiliza exactamente un factor diferente para la autenticación
- ◯ Es lo mismo que la autenticación de 2 factores (2FA)

Los ejemplos de autenticación que utilizan el principio "lo que tienes" incluyen:

- ☐ un teléfono inteligente
- ☐ una contraseña
- ☐ una huella digital

☐ una tarjeta inteligente

En un método de autenticación desafío-respuesta, ¿a quién se le presenta el desafío?

◯ al usuario

◯ al servidor

◯ al programa

◯ al autenticador

El proceso de autenticación básico incluye:

☐ un servidor

☐ datos de autenticación

☐ un usuario

☐ una palabra clave

Las partes de un método de autenticación son:

☐ una entrada

☐ un verificador

☐ un ordenador

☐ un sistema de transporte

¿Qué longitud deben tener las contraseñas según las directrices actuales y las mejores prácticas?

◯ 6 caracteres

◯ 4 caracteres

◯ 7 caracteres

◯ al menos 12 caracteres

¿Qué hace una función hash?

- ◯ hace un hashtag
- ◯ calcula un identificador de datos único
- ◯ crea contraseñas
- ◯ impide la autentitación

¿Qué elementos se pueden utilizar para la autenticación de dos factores?

- ☐ un mensaje SMS
- ☐ un token de identificación
- ☐ una App de teléfono inteligente
- ☐ un nombre de usuario

¿Cuál es el estándar para la autenticación sin contraseña?

- ◯ FIDO2
- ◯ FIBA
- ◯ UFI
- ◯ UPA

Los estándares actuales de la industria para el almacenamiento seguro de contraseñas incluyen:

- ☐ cifrado
- ☐ hash
- ☐ texto sin formato
- ☐ salted hash

¿En qué secuencia se utilizan las claves en el proceso de firma digital?

- ◯ la clave privada del remitente y la clave pública del destinatario
- ◯ la clave privada del remitente y la clave pública del remitente
- ◯ la clave privada del destinatario y la clave pública del destinatario
- ◯ la clave pública del remitente y la clave privada del destinatario

En la técnica de salting de contraseñas, ¿qué tipo de ataque es más difícil?

- ☐ ataques de diccionario
- ☐ ataques de fuerza bruta
- ☐ ataques al servidor
- ☐ ataques al dispositivo del usuario

¿Cuáles son las vulnerabilidades de la técnica de autenticación "lo que sabes"?

- ☐ se puede olvidar
- ☐ se puede perder
- ☐ se puede duplicar
- ☐ se puede negar

Si se utiliza la criptografía de clave pública para el cifrado, ¿qué clave se utiliza para cifrar los datos?

- ◯ la clave pública del destinatario
- ◯ la clave pública del remitente
- ◯ la clave privada del destinatario
- ◯ la clave privada del remitente

¿Cuáles son las vulnerabilidades de la técnica de autenticación "lo que tienes"?

- ☐ se puede olvidar

- ☐ se puede perder
- ☐ se puede duplicar
- ☐ se puede negar

¿Cuál es la secuencia de claves utilizada en el cifrado asimétrico?

- ◯ la clave privada del remitente y la clave pública del destinatario
- ◯ la clave privada del remitente y la clave pública del remitente
- ◯ la clave privada del destinatario y la clave pública del destinatario
- ◯ la clave pública del remitente y la clave privada del destinatario

¿Cuál es la secuencia correcta de pasos en el proceso de firma digital?

- ◯ hash, firmar y enviar
- ◯ firmar, hash y enviar
- ◯ encriptar, hash y enviar
- ◯ hash, codificar y enviar

¿Cuáles son los ejemplos de autentificación por contraseña?

- ☐ contraseñas de un solo uso
- ☐ contraseñas reutilizables
- ☐ contraseñas estructuradas
- ☐ credenciales

¿Cuáles son los componentes de la PKI (infraestructura de clave pública)?

- ◯ CA, MA, LA, firma digital
- ◯ CA, RA, PA, firma digital
- ◯ CA, RA, PKC, certificado digital

○ CA, RA, PKC, firma digital

¿Qué son los ataques no electrónicos a las contraseñas?

☐ mirar por encima del hombro

☐ ingeniería social

☐ ataque de diccionario

☐ ataque de fuerza bruta

¿Cuál es la principal tarea de una CA?

○ emitir certificados

○ emitir firmas digitales

○ comprobar firmas digitales

○ verificar la identidad de un individuo

¿Qué son los ataques electrónicos a las contraseñas?

☐ phishing

☐ ingeniería social

☐ ataque de diccionario

☐ ataque de fuerza bruta

¿Cómo se implementa la PKI?

○ como una estructura de árbol

○ como un sistema cliente-servidor

○ en hardware

○ secuencialmente

¿Cuáles de las siguientes son herramientas para descifrar contraseñas?

- ☐ John the cracker
- ☐ John the ripper
- ☐ Hydra
- ☐ Hybrid

¿Qué no deben incluir las contraseñas?

- ☐ caracteres de diferente tipo
- ☐ palabras asociadas a uno mismo
- ☐ fechas de cumpleaños
- ☐ caracteres especiales

¿Qué funcionalidades suelen incluir los administradores de contraseñas?

- ☐ autocompletar
- ☐ generación de contraseñas
- ☐ evaluación de contraseñas
- ☐ obsolescencia de la contraseña

¿Cuáles son las desventajas de la 2FA?

- ☐ inconveniencia
- ☐ mayor seguridad
- ☐ Preocupaciones sobre la privacidad
- ☐ autenticación más fuerte

¿Cuáles son los principales objetivos de la firma digital?

- ☐ asegurar la autenticación
- ☐ asegurar la integridad
- ☐ asegurar la confidencialidad
- ☐ asegurar la autorización

En el cifrado asimétrico, ¿qué tipo de claves se utilizan?

- ☐ una clave pública
- ☐ una clave secreta
- ☐ una clave de privacidad
- ☐ una clave privada

¿Qué elementos se incluyen en el proceso de firma digital?

- ☐ funciones hash
- ☐ cifrado simétrico
- ☐ algoritmos de intercambio de claves
- ☐ algoritmos de cifrado asimétrico

¿Cuál es la secuencia correcta de pasos en el proceso de verificación de una firma digital?

- ☐ obtener hash de la firma, obtener hash de los datos, comparar
- ☐ obtener hash de los datos, obtener hash de la firma, comparar
- ☐ comparar, obtener hash de los datos, obtener hash de la firma
- ☐ comparar, obtener hash de la firma, obtener hash de los datos

¿Cuáles son los componentes de la PKI (infraestructura de clave pública)?

- ☐ CA
- ☐ PA

- ☐ firma digital
- ☐ certificado digital

¿Cuáles son los datos típicos de los certificados digitales?

- ☐ fecha de caducidad
- ☐ emisor
- ☐ longitud
- ☐ firma digital

¿Cuáles son las principales partes de la cadena de confianza en PKI?

- ☐ un certificado raíz
- ☐ un certificado intermedio
- ☐ una firma digital
- ☐ un certificado de administración

¿Cómo se pueden utilizar los certificados digitales para la autenticación?

- ☐ como factor principal de autenticación
- ☐ como un segundo factor de autenticación
- ☐ no se puede
- ☐ para firmar documentos